

Université Mouloud Mammeri de Tizi Ouzou.
Faculté de Génie Electrique et D'informatique.
Département D'informatique.

MEMOIRE DE FIN D'ETUDE

En vue d'obtention du diplôme de Master 2 en
informatique.

Option : Conduite de projets informatique

Thème :

Etat de l'art sur le Web Sémantique

Réalisé par :
Mr HADDADI Lyes

Encadré par :
Si Mohamed Malik

Juillet 2011

Remerciements

*Je tien à remercier mon promoteur M^R
Si Mohamed Malík, et lui exprimer ma profonde gratitude
pour son aide et conseils.*

*Que les membres du jury trouvent ici mes plus vifs
remerciements pour avoir accepté d'honorer par leur
jugement mon travail.*

*Tout mon respect à toute personne qui de près ou de loin a
contribué à la réalisation de ce modeste travail.*

∞ Dédicaces ∞

Je dédie ce modeste travail à :

- *Mes très chers parents auxquels je dois ma réussite et auxquels je ne rendrai jamais assez. Je leur souhaite une longue vie.*
- *Mes chers frères et soeurs en particulier ma petite sœur FATMA (FAFOUCHE).*
- *Tous mes amis en particulier YACINE.*
- *Tous mes camarades du département informatique.*
- *A ceux qui m'ont poussé à aller toujours de l'avant.*

Sommaire

Table des matières

| | |
|--|----|
| I. Introduction | 01 |
| II. Le World Wide Web..... | 03 |
| 1. Introduction | 03 |
| 2. Historique | 03 |
| 2.1. Les éléments précurseurs | 03 |
| 2.2. Premières briques | 04 |
| 2.3. La naissance d'Internet | 04 |
| 2.4. La naissance du World Wide Web (www) | 05 |
| 3. Définition du Web..... | 06 |
| 4. Fonctionnement du Web..... | 06 |
| 5. Principaux termes du Web..... | 07 |
| 6. Principales caractéristiques du Web..... | 09 |
| 7. Sites web | 10 |
| 7.1. Définition | 10 |
| 7.2. Intérêt d'un site Web | 11 |
| 7.3. Classification des sites Web..... | 11 |
| 7.4. Les technologies de programmation Web | 13 |
| 8. Limitez du Web..... | 15 |
| 9. Évolution du Web | 16 |
| 9.1. Web 1.0 | 16 |
| 9.2. Web 1.5 | 16 |
| 9.3. Web 2.0 | 16 |
| 9.4. Web 3.0 | 18 |
| 9.4.1. Le Web mobile | 18 |
| 9.4.2. Les applications Web | 19 |
| 9.4.3. Le Web sémantique | 19 |
| 10. Conclusion | 20 |
| III. Le Web sémantique | 21 |
| 1. Introduction | 21 |
| 2. Vers le Web Sémantique (WS) | 21 |
| 3.1 Problématique initiant au web sémantique | 21 |
| 3.2 Qu'est ce que le web sémantique? | 23 |
| 3.2.1. Définition | 23 |
| 3.2.2. Propriétés | 24 |

Table des matières

| | |
|--|----|
| 3.3 Historique | 25 |
| 3.4 Fondements | 29 |
| 3.4.1. Les logiques de description | 29 |
| 3.4.2. Autres influences sur le Web sémantique | 37 |
| 3.5 Objectifs | 37 |
| 3. Composants principale du Web Sémantique | 38 |
| 3.1. Les ontologies | 38 |
| 3.1.1. Introduction | 38 |
| 3.1.2 Origine et définitions | 38 |
| 3.1.3. Les composants d'une ontologie | 40 |
| 3.1.4. Cycle de vie d'une ontologie | 42 |
| 3.1.5. Rôles de l'ontologie | 42 |
| 3.1.6. Classification des ontologies | 43 |
| 3.1.7. Langages de définition et de manipulation d'ontologies | 45 |
| 3.1.8. Techniques d'Intégration d'ontologie | 46 |
| 3.1.8.1 Le mapping d'ontologies | 46 |
| 3.1.8.2 La fusion d'ontologies | 47 |
| 3.1.8.3 L'alignement des ontologies | 47 |
| 3.1.9. Outils et environnements pour la construction d'ontologie | 47 |
| 3.1.10. Systèmes de raisonnement sur les ontologies | 50 |
| 3.1.11. La place des Ontologies dans le Web sémantique | 50 |
| 3.1.12. Limites de l'approche ontologique pour le WS | 51 |
| 3.1.13. Conclusion et perspectives | 51 |
| 3.2. Les métadonnées et annotations sémantique | 53 |
| 3.2.1. Introduction | 53 |
| 3.2.2. Généralités | 53 |
| 3.2.2.1. Qu'est-ce qu'une métadonnée ? | 53 |
| 3.2.2.2. Qu'est-ce qu'une annotation? | 53 |
| 3.2.2.3. Intérêts des métadonnées/annotations | 55 |
| 3.2.3 Processus d'annotation sémantique | 56 |
| 3.2.4. Familles d'annotations et de métadonnées | 57 |
| 3.2.5. Outils d'annotation sémantique | 58 |
| 3.2.6. L'annotation sémantique/métadonnée et le WS | 59 |
| 3.2.7. Travaux sur les métadonnées/annotations pour le WS | 60 |
| 3.2.8. Conclusion et perspective | 60 |
| 3.3 Les agents | 61 |
| 4. Architecture et langages du web sémantique | 62 |
| 4.1. Le World Wide Web Consortium (W3C) | 62 |
| 4.2. Architecture | 62 |
| 4.3. Langages | 64 |
| 4.3.1. Les précurseurs | 64 |
| 4.3.2. Langage XML | 65 |
| 4.3.3. Langage RDF | 69 |

Table des matières

| | |
|---|-----|
| 4.3.4. Langage OWL | 71 |
| 4.3.5. Langage SWRL | 75 |
| 4.3.6. Langage SPARQL | 76 |
| 4.4. Conclusion | 78 |
| 5. Le web services sémantiques (WSS) | 79 |
| 5.1. Introduction | 79 |
| 5.2. Problématique initiant au web services sémantiques | 79 |
| 5.3. Présentation et objectifs | 80 |
| 5.4. Architecture Étendue | 81 |
| 5.5. Approches pour la réalisation des services Web sémantiques | 82 |
| 5.6. Conclusion et perspective | 89 |
| 6. Quelques applications du Web Sémantique | 91 |
| 6.1. Introduction | 91 |
| 6.2. Mémoires d'entreprise | 91 |
| 6.3. Applications médicales | 93 |
| 6.4. E-learning | 94 |
| 6.5. Recherche d'information | 95 |
| 6.6. Les wikis sémantiques | 96 |
| 6.7. Conclusion | 102 |
| VI. Conclusion et perspectives | 103 |
| Bibliographie | 104 |
| Annexe A | 118 |

Liste des figures

| | |
|---|----|
| Figure 01 : Traitement des pages web statiques..... | 12 |
| Figure 02 : Traitement des pages dynamiques | 13 |
| Figure 03 : Évolution du Web | 18 |
| Figure 04 : Proposition d'architecture distribuée qui conduira au World Wide Web | 22 |
| Figure 05 : Evolution du Web actuel vers le Web sémantique | 24 |
| Figure 06 : Hiérarchie des concepts à base de la relation Est-Un | 29 |
| Figure 07 : Une hiérarchie en DL | 30 |
| Figure 08 : Hiérarchie des logiques de la famille ALC | 33 |
| Figure 09 : Cycle de vue d'une ontologie | 42 |
| Figure 10 : Typologie et classification des ontologies | 45 |
| Figure 11 : vision de l'utilisation des métadonnées sur le Web sémantique | 55 |
| Figure 12 : les couches du Web Sémantique | 63 |
| Figure 13: Exemple d'une annotation sémantique en HTML-A | 64 |
| Figure 14 : Exemple d'une annotation sémantique en SHOE | 65 |
| Figure 15 : Exemple de représentation arborescente d'un document XML | 66 |
| Figure 16 : Exemple de fichier XML simple | 67 |
| Figure 17 : Triplet RDF | 69 |
| Figure 18 : Origine du langage OWL | 72 |
| Figure 19 : la relation entre les trois versions de l'OWL | 73 |
| Figure 20 : Pile des SW | 82 |
| Figure 21 : Opération WSDL augmentée de sémantique | 83 |
| Figure 22 : Organisation de Module d'OWL-S | 85 |
| Figure 23 : Relations entre OWL-S et WSDL | 86 |
| Figure 24 : Architecture d'IRS-II | 87 |

Liste des figures

| | |
|---|-----|
| Figure 25 : Exemple du wikitext de MediaWiki (à gauche) et son affichage (à droite) | 97 |
| Figure 26 : Différences de représentation entre un wiki (à gauche) et un wiki sémantique (à droite) | 98 |
| Figure 27 : Wikitext dans Semantic Mediawiki (à gauche) et son résultat (à droite) | 99 |
| Figure 28 : utilisation des technologies du WS dans les moteurs wiki sémantique | 102 |
| Figure 29: Application de voyage de business qui interagit avec les SW | 119 |
| Figure 30 : Architecture des web services | 122 |
| Figure 31 : Architecture en Pile des services Web..... | 123 |
| Figure 32 : Exemple de Requête SOAP | 124 |
| Figure 33 : Exemple de Réponse SOAP | 124 |
| Figure 34 : Exemple de Type | 126 |
| Figure 35 : Exemple de Message..... | 126 |
| Figure 36 : Exemple de Port Type..... | 126 |
| Figure 37 : Exemple de Binding..... | 126 |
| Figure 38 : Exemple de Service..... | 127 |
| Figure 39 : Serveurs de Registre des Affaires | 127 |



Introduction
Générale

I. Introduction:

La vision du Web sémantique est celle d'un Web dans lequel les ressources sont accessibles non seulement aux humains, mais également aux processus automatisés comme celui d'un agent automatisé errant le Web, accomplissant des tâches utiles telles que la recherche améliorée (en termes de précision), la découverte de ressource le filtrage d'information, ...etc., ce qui transformera le web en un vaste espace d'échanges de ressources entre machines, permettant l'exploitation de grands volumes d'informations et de services variés. L'automatisation des tâches nécessite d'élever le statut du Web lisible par machine à quelque chose que nous pouvons appeler compréhensible par machine.

L'idée principale est d'avoir des données sur le Web définies et liées de telle manière que leur signification soit interprétable par des humains et des machines.

Pour réaliser cette vision, il sera nécessaire d'annoter des ressources Web avec les métadonnées (c.-à-d., les données décrivant leur contenu/fonctionnalité). En particulier, nous pouvons souhaiter annoter des ressources Web avec les métadonnées sémantiques qui fournissent une certaine indication de la teneur d'une ressource. C'est plus qu'une annotation textuelle simple, car l'intention dans le contexte du Web sémantique est que cette information soit accessible non seulement aux humains mais également aux agents logiciels. Afin de faire ceci, nous avons besoin de langages qui soutiennent la représentation des métadonnées sémantiques. Les propositions de langages de métadonnées sont issues de recommandations du World Wide Web Consortium (W3C), c'est le cas particulier de Resource Description Framework (RDF), de RDF Schéma (RDF(S)) et de XML.

Cependant, de telles annotations auront une valeur limitée au processus automatisé à moins qu'elles ne partagent une compréhension commune quant à leur signification. Les ontologies peuvent aider à répondre à cette exigence en fournissant une "représentation d'une conceptualisation partagée d'un domaine particulier" et un vocabulaire partagé, contrôlé et qui peut être communiqué à des personnes et à des applications.

Les recherches réalisées pour atteindre les objectifs de cette nouvelle vision du web s'appuient sur un existant riche venant, d'abord, des recherches en représentation ou en ingénierie des connaissances, mais aussi de bien d'autres domaines comme les bases de données, la démonstration automatique de théorèmes, l'apprentissage automatique, l'hypermédia, l'ingénierie linguistique ou bien encore les interactions homme machine. Mais l'utilisation et l'acceptation de ces recherches à l'échelle du (ou d'une partie du) Web posent de nouveaux problèmes et défis : changement d'échelle dû au contexte de déploiement : le Web sur Internet et ses dérivés (intranet, extranet), la nécessité d'un niveau élevé d'interopérabilité, d'ouverture, de standardisation, diversités

Introduction générale

des usages, distribution bien sûr et aussi impossibilité d'assurer une cohérence globale.

- Dans le cadre de notre mémoire dont l'objectif est de mettre en évidence :
 - Les points importants pour la réalisation de la vision du Web,
 - Les problèmes rencontrés lors de sa réalisation,
 - Les perspectives futures.

- Et pour mener à bien notre travail nous avons opté pour le plan suivant :
 - ✓ Après une introduction générale, nous consacrerons le premier chapitre au Web dans lequel nous retracerons brièvement son historique, ces concepts de base et son évolution pour introduire le concept de notre mémoire.
 - ✓ le deuxième chapitre sera consacré au web sémantique dans lequel nous :
 - Définirons et mettrons en évidence son historique, ces fondements et ces objectifs,
 - Présenterons les composants principale (Les ontologies, Les métadonnées/ annotations sémantique),
 - Nous présenterons également son architecture et ces différents langages,
 - Et pour terminer ce chapitre nous présenterons quelques applications qui ont largement bénéficié des technologies du web sémantique.

 - ✓ Pour conclure, après cet état de l'art nous présenterons les différentes perspectives de recherche.



II

Le World Wide Web

II. Le World Wide Web:

1. Introduction :

L'Internet est constitué d'un vaste réseau de serveurs interconnectés entre eux à l'échelle mondiale et donnant accès à une masse considérable d'informations. Parmi les nombreuses applications de l'Internet, il en est une qui a pris le pas sur toutes les autres, au tournant des années 90 : le World Wide Web.

Les applications possibles du World Wide Web, ou Web, sont multiples. Le Web est devenu pratiquement indispensable dans notre quotidien. Il relie plus de 600 millions d'ordinateurs (312.693.296 sites en avril 2011) et 1,8 milliard de personnes y ont recours tous les jours. [http://news.netcraft.com/archives/category/web-server-survey]

2. Historique :

1. Les éléments précurseurs :

En 1957, l'URSS est la première des deux superpuissances à envoyer un satellite artificiel dans l'espace : c'est le fameux Spoutnik. Ce qui a poussé les États-Unis à créer un groupe appelé ARPA ("*Advanced Research Projects Agency*"), constitué de scientifiques, chargé de concevoir des innovations technologiques appliquées à l'armée. L'objectif étant de constituer un réseau garantissant la transmission de données en cas de conflit nucléaire au sol.

En 1962, l'US Air Force demande à un groupe de chercheurs de RAND (de "*Research ANd Development*", association non lucrative visant à développer les sciences et l'éducation aux États-Unis) de concevoir un réseau capable résister à une frappe nucléaire massive, afin de pouvoir riposter à son tour.

La solution est un système décentralisé, qui permet au réseau de continuer à fonctionner même si une ou plusieurs machines est touchée. L'idée de décentralisation est due à Paul Baran. Plus précisément, c'est lui qui pensa à un système où chaque machine, maillon d'un réseau en toile d'araignée, chercherait, à l'aide de paquets de données dynamiques, la route la plus courte possible d'elle-même à une autre machine, et où elle patienterait en cas de « bouchons. »

Le projet de Paul Baran est refusé par les militaires et ce n'est que 6 ans plus tard qu'il se concrétise.

En 1968, a été initié le projet pour la création d'un réseau d'ordinateurs, baptisé ARPANET.

2. Premières briques :

➤ ARPANET

En août 1969, un réseau décentralisé se met en place sur commande de l'ARPA à BBN (Bolt Beranek and Newman Inc., une SSII de Cambridge, Mass.). Il comprend quatre grands centres universitaires américains :

- Le Stanford Institute(SRI) ;
- L'université de Californie à Los Angeles (UCLA);
- L'université de Californie à Santa Barbara (UCSB);
- L'université d'Utah.

Ces quatre centres étaient reliés par des câbles 50Kbps, et utilisaient le NCP ("*Network Control Protocol*").

Le réseau ARPANET est aujourd'hui considéré comme le réseau précurseur d'internet. Il comportait déjà à l'époque certaines caractéristiques fondamentales du réseau actuel :

- Un ou plusieurs noeuds du réseau pouvait être détruits sans perturber son fonctionnement ;
- La communication entre machines se faisait sans machine centralisée intermédiaire ;
- Les protocoles utilisés étaient basiques.

En mars 1970, un cinquième site est connecté : la firme BBN (Bolt Beranek and Newman) à Cambridge.

En avril 1971, 15 sites sont connectés. et sa augmente de plus en plus.

3. La naissance d'Internet :

➤ TCP/IP :

En 1973 se développe se que l'on appellera plus tard le protocole TCP/IP, l'une des pierres d'angle de l'Internet actuel, sous la houlette de Vinton Cerf, de Stanford, et de Bob Kahn, de la DARPA (nouveau nom de l'ARPA).

Ce sont ces deux hommes qui, **en 1974**, parlèrent pour la première fois d'« **Internet** ». Le protocole TCP/IP sera adopté par le Département de la défense pour l'Arpanet **en 1976**, composé de 111 machines reliées entre elles.

En 1974, mise au point de la norme IP.

En 1981, le protocole « TCP/IP » a été standardisé en tant que protocole « public » pour Internet. Il s'agissait d'un accord précis (d'où le nom de « protocole »), toujours en vigueur de nos jours, et qui fixait les modalités des échanges de données entre les ordinateurs reliés par le biais d'un réseau. Ce protocole est constitué de règles et de liaisons formelles, destinées à régir la communication entre les ordinateurs. Le protocole TCP/IP permet donc aux ordinateurs de communiquer entre eux.

➤ **Le DNS :**

Au début de l'Arpanet, les informations nécessaires à la connection des machines entre elles (conversion nom <-> adresse) sont contenues dans un fichier nommé `hosts.txt`. Ce fichier est maintenu par le *Network Information Center* (NIC en abrégé) de l'Institut de recherche de Stanford. Chaque administrateur d'une machine reliée à l'Arpanet doit envoyer ses modifications au NIC qui les centralise et redistribue périodiquement le `hosts.txt` mis à jour. Au fur et à mesure que l'Arpanet se développe, le système devient trop lourd à gérer : le NIC ne peut plus faire face à la charge réseau, et il y a des problèmes de collision (deux machines qui ont le même nom) qui peuvent mettre en danger le bon fonctionnement de l'Arpanet.

En 1983, pour résoudre ce problème, un groupe constitué de Jon Postel, Paul Mockapetris et Craig Partridge rédige les RFC 882 et 883 : le DNS ("*Domain Name System*") est inventé. C'est une base de données distribuée qui permet une gestion locale des noms de domaine, tout en rendant l'information disponible à tous. La base de données est divisée en zones. Pour chaque zone, un ou plusieurs *serveurs de noms* (*name servers* en anglais) répond aux requêtes des *résolveurs*. Les résolveurs sont des programmes qui communiquent entre les programmes utilisés par une machine et les serveurs de noms, et permettent de faire le lien entre nom d'une machine (`clipper.ens.fr` par exemple) et adresse IP.

En 1984 se mettent en place les « *top level domains* », c'est-à-dire les suffixes comme `.com`, `.gov`, `.net` ou encore `.org`.

4. La naissance du World Wide Web (www) :

En 1989, le chercheur Tim Berners-Lee a proposé la création d'un réseau de documents. Le but était de relier logiquement les différents documents par le biais de ce que l'on qualifie d'hyperliens.

Fin 1990, Tim Berners-Lee met au point le protocole HTTP (Hyper Text Transfer Protocol), ainsi que le langage HTML (HyperText Markup Language)

permettant de naviguer à l'aide de liens hypertextes, à travers les réseaux. et signe la fin de l'ARNANET.

En 1991, le WWW (World Wide Web) devient accessible via internet.

Chaque jour, le World Wide Web croît et grandit, tant en ce qui concerne le nombre d'ordinateur connecté que celui des utilisateurs. Mais c'est quoi donc le World Wide Web, comment fonctionne t'il, quels sont ces caractéristiques..., C'est à ces questions que nous tenterons de répondre dans la suite de ce chapitre.

3. Définitions du Web:

Terme anglais : *World Wide Web* (ou *WWW* et par extension *Web* tout court). Traduction en français : Toile d'araignée mondiale. La toile est tissée par des liens hypertextuels qui relient les pages des sites web. Ces liens permettent la recherche d'information dans Internet, l'accès à cette information et sa visualisation. [Fabrice Molinaro 2004]

Le Web est un service internet spécifique. Il s'agit d'un ensemble de ressources mis à disposition par le biais d'Internet. Un document est par exemple considéré comme une ressource. Dans ce cadre, les ressources sont stockées sur différents serveurs Web, répartis sur toute la planète. Il s'agit donc d'une architecture partagée. Les serveurs Web sont accessibles via le protocole Internet « http » ou « HyperText Transfer Protocol ». L'affichage des données est effectué par le biais de pages HTML (HyperText Markup Language). Avant l'introduction du Web, les données ne pouvaient être consultées que sous forme de textes. [www.cases.lu]

4. Fonctionnement du Web :

Le web est fondé sur le modèle client/serveur. Dans ce modèle, deux programmes, **le client web** (navigateur) et **le serveur web**, travaillent ensemble pour accomplir une tâche spécifique en saisissant une URL (Uniform Resource Locator) dans la barre d'adresse. Les pages web accessibles par le navigateur sont stockées sur ces serveurs physiques sous forme de fichier HTML. L'URL permet d'indiquer aux navigateurs web comment accéder à toutes les ressources d'Internet.

- **Requête d'un navigateur web :**

Lorsqu'un navigateur effectue une requête, les étapes suivantes se réalisent :

- Les données de requête sont envoyées au serveur sous forme d'entête de requête http ;
- Lorsque le serveur reçoit la requête, il analyse les entêtes http, notamment celui permettant de localiser le fichier demandé ;
- Si le serveur trouve le fichier HTML demandé, il va envoyer au client (navigateur) un entête de réponse valide (généralement succès) et les données créées par l'application ;
- A la réception du document, le navigateur affiche le résultat à l'écran ;

5. Principaux termes du Web :

➤ **Page web :**

Les pages web représentent l'ensemble des informations et des connaissances proposées au visiteur d'un site web. Plus généralement, le contenu des pages est un ensemble de documents, de composants multimédias (texte, son, vidéo, etc.) ou de données diverses.

La technologie Internet permet de disposer de ce contenu en ligne ou de le télécharger.

On distingue deux types de pages web statiques dont le contenu ne varie pas et des pages web dynamiques dont le contenu varie en fonction du contexte.

C'est la page d'introduction d'un site web s'appeler « page d'accueil » ou « l'index ». Elle est très importante car elle détermine la première impression de l'internaute.

➤ **Site web :**

C'est un ensemble de données (texte, son, image, vidéo, etc....) hébergées sur un serveur ayant une adresse IP et accessible aux internautes. Il existe différents types de sites selon leurs contenus et la nature des services qu'ils offrent. Citons les sites FTP, web, etc.

➤ **Serveur Internet :**

C'est l'ensemble des ressources matérielles (espace disque, connexion Internet, contenu du site, etc.) et logicielles (les programmes) qui servent les clients d'où émanent les demandes de services.

➤ **URL (Uniform Resource Locator) :**

C'est une adresse universelle d'une page web.

Une URL s'apparente à un nom de serveur et une information sur le type de protocole de réseau à utiliser pour atteindre la ressource.

Exemple :

www.siteduzero.com

<http://www.commentcamarche.net/contents/networking/>

➤ **Hypertexte :**

C'est une forme non séquentielle non narrative d'organisation et d'accès à l'information représentée par un réseau de liens associatifs qui forment une structure d'hypertexte.

➤ **Hypermédia :**

Le terme hypermédia désigne un hypertexte dont on insiste sur la nature multimédia de son contenu informationnel.

➤ **Hyper document :**

Le terme hyper document désigne un hypertexte dont le contenu informationnel sont des documents.

Il désigne un ensemble de documents multimédia ayant une organisation et un fonctionnement hypertextuel.

➤ **Lien hypertexte :**

C'est un pointeur qui pointe un nœud de destination, souvent il est sous forme d'une icône, symbole, petite image, un mot dans un texte, etc. Il permet d'explorer le web avec un simple clic de souris .On distingue :

- Les liens internes qui renvoient à d'autres pages du même site.
- Les liens externes qui renvoient à d'autres pages d'un autre site.
- Les liens vers un e-mail qui permettent à l'internaute d'envoyer un e-mail directement depuis la même page.
- Les liens de téléchargement qui permettent de télécharger des documents depuis la même page.

➤ **Navigateur :**

C'est le logiciel qui offre une interface d'accès aux différentes informations disponibles sur le web et l'exploration de ce dernier, ce dernier interprète le texte et les commandes de formatage qu'il contient et affiche à l'écran la page correctement formatée. Les plus connus sont Internet Explorer, Netscape, Mozilla FireFox.

➤ **Moteur de recherche :**

Un moteur de recherche est un programme qui indexe le contenu de différentes ressources internet et plus particulièrement de sites web, il permet à l'internaute de rechercher de l'information à partir d'un navigateur web, selon différents paramètres, en se servant de mots clés, et qui permet d'avoir accès à l'information ainsi trouvée.

6. Les principales caractéristiques du Web:

Le web comporte un certain nombre de caractéristiques spécifiques qui en font le système d'information le plus perfectionné qui ait fait parti d'Internet à ce jour :

- **Le web est un système hypertexte** : dans lequel les utilisateurs passent d'un document à un autre par l'entremise de liens hypertextuels.
- **Le web est un système multimédia** : Au début d'Internet, la plus grande partie des informations accessibles sur le réseau se présentait sous forme de texte ASCII dépourvu de tous éléments graphiques ou de mise en forme (titres en gros caractères, italiques, images...). Grâce aux explorateurs graphiques, le web peut contenir les éléments suivants :

Texte ordinaire, Caractères spéciaux : (gros caractères, caractères gras et italiques, par exemple), Images : (graphiques, logos fantaisistes et illustrations, par exemple), Séquence audio : (son, musique, commentaires et messages vocaux), Séquence vidéo : (Séquence de films, animations, et simulations générées par ordinateur.).

- **Le web est un système de traitement distribué** : De façon générale, les documents mémorisés dans un système hypertexte sont tous réunis au même endroit, sur un disque dur ou un CD-ROM, par exemple. Sur le web, les documents peuvent être distribués dans toutes les mémoires reliées au réseau.

- **D'autres outils d'Internet sont intégrés au web** : Le web peut assurer la liaison avec d'autres outils d'Internet comme FTP, TELNET, GOPHER

Ainsi une page web peut renvoyer à une autre ressource d'Internet. Peu importe si celle-ci est accessible par l'entremise de GOPHER, TELNET ou d'un FTP. Le web et ses explorateurs constituent donc une méthode d'accès direct aux informations disponibles sur le réseau.

- **Le web assure l'interface avec les systèmes des bases de données** : La mise en œuvre d'une base de données sur un serveur web permet d'étendre les possibilités d'interaction avec les utilisateurs. A la différence des pages classiques basées sur du code HTML fixe (on parle de pages web statiques), les bases de données permettent de dynamiser le contenu d'une page web présentée à l'utilisateur (on parle de page web dynamique). Les fonctions dynamiques prises en compte concernent généralement :

-La présentation du résultat d'une requête de l'utilisateur.

-La mise à jour simple de la base de données par l'utilisateur

Cette capacité d'agir comme interface pour un système de bases de données relié à Internet est l'une des fonctionnalités les plus puissantes des explorateurs (navigateurs) web

7. Sites web :

7.1. Définition :

Un site web (aussi appelé site internet) est un ensemble de fichiers HTML stockés sur un ordinateur connecté en permanence à internet et hébergeant les pages web (serveur web).

Un site web est habituellement architecturé autour d'une page centrale, appelée «**page d'accueil**» et proposant des liens vers un ensemble d'autres pages hébergées sur le même serveur, et parfois des liens dits «externes», c'est-à-dire de pages hébergées par un autre serveur.

7.2. Intérêt d'un site web :

L'intérêt de mettre en place un site web est motivé par plusieurs raisons :

- Le besoin de visibilité : un site web, dans la mesure où il fait l'objet d'une bonne campagne de promotion, peut être un moyen pour une enseigne ou une organisation d'augmenter sa visibilité.
- Développer sa popularité auprès du public.
- Une formidable opportunité pour les entreprises de recueillir des données de leurs clients ou bien de démarcher de nouveaux prospects
- La vente en ligne : frileux aux débuts d'Internet, les internautes ont vite compris l'intérêt d'Internet pour l'achat de certains produits de consommation. Un site Internet peut représenter pour certaines entreprises une opportunité en termes de commercialisation
- La mise en place d'un support aux utilisateurs : de plus en plus de sociétés utilisent Internet comme support privilégié pour le service avant-vente ou après-vente. En effet, avec un site web, il est possible de mettre à disposition des internautes un maximum d'informations commerciales ou techniques, à moindre coût.

7.3. Classification des sites Web:

Les objectifs assignés aux sites web sont divers et permettent globalement de classer les sites web dans les catégories suivantes :

- **Sites web vitrine** : il s'agit des sites de présentation de produits et/ou d'informations sur un thème.
- **Sites web portails** : ce sont des sites qui ouvrent l'accès à d'autres sites. Ce sont des « annuaires thématiques », tels que le site **Yahoo**, accessible avec l'URL **www.yahoo.fr**
- **Sites commerciaux** : ce sont des sites de commerce électronique (e-commerce). Un site commercial présente une entreprise, une société ou une activité et propose ses produits et services à la vente en ligne
- **Sites communautaires et interactifs** : reposent sur leurs visiteurs pour créer et actualiser le contenu, tel que, les forums de discussion composés de plusieurs milliers de membres qui créent eux-mêmes les sujets de discussion et influencent ainsi l'évolution du site. Les clubs de membres sont aussi des sites communautaires administrés, maintenus et visités par et pour un groupe de personnes, les membres du site.

- **Sites personnels** : est fait pour se présenter, présenter sa famille, son parcours, parler de ses passions...

Par ailleurs, les sites web peuvent être classés selon le type de pages web qu'ils offrent

- **Les sites statiques** : présentent des pages web statiques. Le langage utilisé pour décrire le contenu et la présentation de ces pages est le langage HTML. Ces pages peuvent aussi contenir différents morceaux de programmes appelés scripts qui améliorent globalement l'ergonomie et le contrôle (menus déroulant, saisie des données contrôlées).

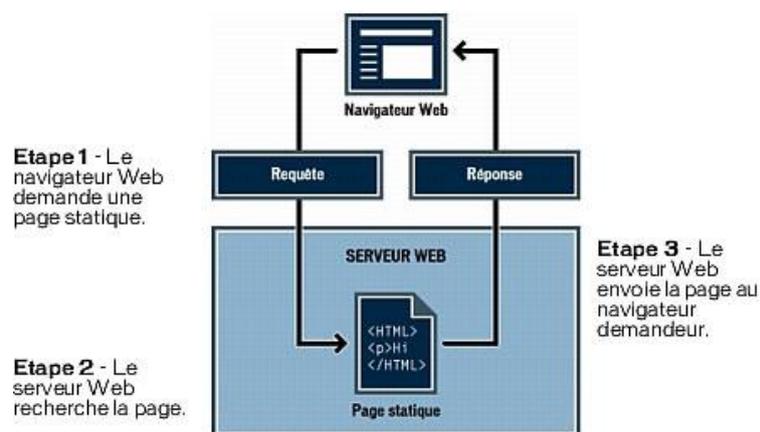


Figure 01. Traitement des pages web statiques

- **Les sites dynamiques** : Une application web offre des services qui dépassent la simple visualisation d'informations figées. L'application est souvent reliée à une base de données. Les fonctionnalités offertes sont plus importantes : recherches, formulaires de saisie ... Le développement d'une application web nécessite l'écriture de programmes qui s'exécutent sur le serveur.

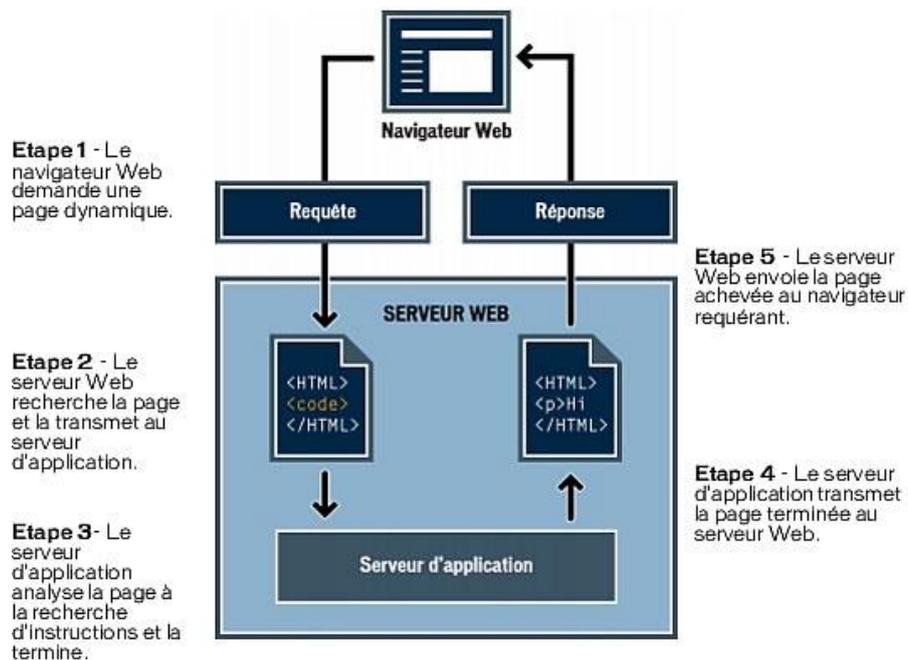


Figure 02. Traitement des pages dynamiques

7.4. Les technologies de programmation web :

a. Côté client :

Les technologies susceptibles d'être interprétées par le navigateur ont des fonctions limitées car elles n'agissent que sur la machine du client, mais elles permettent d'ajouter des fonctionnalités appréciables à l'application, à l'esthétique de la page web.

- ▶ HTML (Hyper Text Markup Language) : HTML est un langage de description de documents. Il utilise des marqueurs explicites (appelés aussi tags ou balises) qui précisent la structure et la mise en forme du contenu du document. Ces marqueurs seront reconnus par les navigateurs et interprétés comme des directives, afin de réaliser la présentation attendue sur le poste client.
- ▶ XML (Extensible Markup Language) : Tout comme HTML, le langage XML représente un langage de balises décrivant la structure et la nature des données d'un document. Mais la différence du langage HTML, statique, XML est davantage orienté traitement des données et publication des documents complexes.

- ▶ Les applets : Un applet, ou en français une « appliquette » ou parfois « applette », est un logiciel qui s'exécute dans la fenêtre d'un navigateur web. De manière générale, il s'agit d'un script java intégré aux documents HTML qui, compilé sur le serveur, est chargé par le navigateur pour augmenter l'interactivité et les fonctionnalités de l'interface visuelle. Le navigateur doit être compatible avec la technologie utilisée par l'appliquette ou, s'il ne l'est pas, de pouvoir l'être en installant un *plugin*. Pour des raisons de sécurité, les possibilités des applets sont limitées. Ils n'utilisent que les ressources de la machine virtuelle Java du navigateur. Ils ne peuvent pas accéder aux fichiers stockés sur les disques de la machine client et ne sont pas non plus habilités à communiquer avec des adresses réseaux autres que celles du serveur web à partir duquel ils ont été chargés.
- ▶ Les composants ActiveX : Produit par Microsoft, ActiveX a été spécialement conçu pour fonctionner avec les systèmes Windows. ActiveX est un processus par lequel Internet Explorer (IE) charge diverses applications dans le logiciel de navigation. Grâce à ActiveX, IE télécharge Windows Média Player, Flash, Quicktime, ou d'autres applications capables de lire les fichiers intégrées dans les pages web.
- ▶ JavaScript : Initialement développé par Netscape, JavaScript fut adopté à la fin de l'année 1995. JavaScript est un langage de scripts qui, incorporé aux balises HTML, permet d'améliorer la présentation et l'interactivité des pages web.

b. Côté serveur :

Les principales technologies côté serveur sont définies comme suit :

- ▶ JSP (Java Server Pages) : Ce sont des standards permettant de développer des applications web interactives, dont le contenu est dynamique. Les JSP sont intégrables au sein d'une page web en HTML à l'aide des balises spéciales permettant au serveur web de savoir que le code compris à l'intérieur de ces balises doit être interprété afin de renvoyer du code HTML aux navigateurs du client.
- ▶ ASP est le produit de remplacement Microsoft CGI. Il s'agit d'une fonctionnalité intégrée et livrée avec son serveur web IIS (Internet Information Server). Les ASP contiennent des scripts qui sont exécutés

sur le serveur. Ces scripts permettent d'appeler d'autres scripts contenus dans d'autres pages, de gérer des cookies ou d'exécuter des composants ActiveX. Ils peuvent être considérés comme des interpréteurs des scripts (VB Script, JavaScript) intégré au serveur web.

► **PHP** : PHP créée en 1995 est utilisée aujourd'hui sur plus de 25 millions de sites web. La version actuelle est la version 5.0. Le PHP est un langage complet écrit en C et en Perl, qui reprend une grande partie des spécificités techniques et sémantiques de ces langages. C'est un langage de script incorporé au document HTML, adapté au développement de pages dynamiques. PHP est interprété par un serveur web (Microsoft IIS, Apache étant le plus répandu pour l'utilisation de PHP). Les sites ou applications en PHP sont en général rattachés à une base de données (de nombreux SGBD sont supportés, mais le plus utilisé avec ce langage est MySQL).

8. Limites du web :

Depuis une quinzaine d'années, le World Wide Web est devenu un médium incontournable pour la diffusion et le partage d'information, ainsi que pour un grand nombre de services.

Le Web que nous connaissons et utilisons aujourd'hui est dédié à l'utilisation pour des êtres humains. Dans ce Web, les gens peuvent créer des pages Web, écrire ce qu'ils veulent dire, dessiner et décorer comme ils imaginent, mettre en page un document comme ils le souhaitent. Le but est de fournir des informations aux utilisateurs humains et de laisser ceux-ci interpréter ces contenus. Quant aux logiciels, ils ne comprennent rien, leur mission est de représenter les informations, les contenus exactement comme leurs auteurs souhaitent. Ils ne sont pas capables de comprendre qu'une page web donnée concerne une personne, qu'une autre page décrit les caractéristiques d'une nouvelle voiture de Renault. Il est impossible de demander à la machine de trouver de manière automatique le nom de la personne qui a conçu ce modèle de voiture.

L'utilisateur a facilement accès à un nombre très important de documents contenant des informations aussi diverses qu'abondantes. Cependant, outre le volume important d'information disponible, le web a des limites qui lui sont inhérentes :

- Information sur le Web essentiellement prévue pour être affichée (écran, imprimante) et lue par des humains.
- Il est essentiellement syntaxique : contenu quasi inaccessible aux traitements machines seuls les humains peuvent interpréter ces contenus.

Ajouter a cela :

- Hétérogénéité des formats, des informations
- Absence de description et d'indexation des ressources
- Imprécision de la recherche d'information (problèmes de polysémie (bruit) ; de synonymie (silence) ; du tri des milliers de résultats obtenus en réponse à une questionetc.)
- Absence de structure explicite globale du web : réseau de nœuds et de liens, mais pas d'exploitation sémantique des liens hypertexte.

9. Évolution du web :

Le Web est caractérisé par une évolution constante du fond et de la forme des pages Web.

9.1. Web 1.0 :

Le Web dit 1.0 est apparu au cours des années **1995-1996**, il se caractérise par l'émergence d'un nombre important de pages statiques au contenu codé en HTML. Ces sites 1.0 sont non-interactifs et ne disposent que de peu d'informations. (Il fallait bien passer au 2.0, pour s'apercevoir qu'il y avait un web 1.0).

Cette forme du web est encore utilisée aujourd'hui sur des sites simples ou des pages personnelles.

9.2. Web 1.5 :

En 2000 ; le Web a évolué au niveau 1.5, il est devenu dynamique. Il devient alors possible par exemple de consulter des sites recensant des milliers d'informations comme un BDI en ligne ou une boutique en ligne.

Ce Web dynamique est généralement basé sur l'association du langage de programmation PHP et des bases de données Mysql. Lorsque l'internaute accède au site dynamisé, il fait exécuter sur le serveur le langage PHP qui va chercher l'information dans la base de données pour la retranscrire dans la page HTML sur le poste utilisateur.

9.3. Web 2.0 :

De la simple consultation des données à un lieu d'expression rich media :

En 2004; le web va subir une nouvelle évolution, En effet c'est lors de l'organisation d'une conférence traitant du Web, que Dale Dougerthy a pour la première fois évoqué le Web 2.0, ce sera l'avènement du collaboratif, interactif et participatif.

Le Web 2.0 est plus orienté partage de données, il répond à la question que devait se poser tout propriétaire de site internet : Comment ce que je mets en

ligne peut-être partagé avec d'autres utilisateurs. L'internaute n'est plus simplement spectateur, on veut qu'il devienne acteur. Dans cette optique sont apparus les réseaux sociaux et le phénomène de syndication : les flux RSS. L'internaute est devenu acteur.

Avancées du Web 2.0 :

- Les pages Web toujours dynamisées adopteront une ergonomie et une interactivité nouvelle: il sera par exemple possible d'ajouter du contenu à certaines pages ou encore de déplacer des éléments sur celle-ci.
- Des applications aujourd'hui seulement disponibles en version logicielle (mono-utilisateur) seront accessible en ligne et permettront un travail collaboratif : multiutilisateurs
- Il sera possible sur certains sites web de modifier, de rajouter ou d'effacer du contenu, d'ailleurs une encyclopédie collaborative sera basée sur ce concept (multiutilisateurs)
- On pourra appartenir à des réseaux sociaux, avec lesquels on échangera par des techniques synchrones comme les messageries instantanées, la téléphonie sur internet, ... ou des méthodes asynchrones comme les forums, les wikis, la fonction commentaire de certaines pages, ou les messageries en ligne.
- On pourra également partager nous informations de la vie de tous les jours de manière asynchrone en créant un journal de bord (blog) en ligne. Ce blog pourra également être multiutilisateurs (collaboratif)
- Des plateformes de partages de fichiers multimédia et des Webtv verront également le jour. on partagera avec le monde entier nos films ou photos depuis chez nous.
- Il nous sera possible de réaliser une veille d'informations, en suivant les actualités de pages Web depuis une interface qui s'appellera : « agrégateur de flux »
- on assistera également à un développement considérable du e-commerce.

Cet ensemble d'innovations sera permise par l'apparition de nouvelles technologies comme le langage AJAX qui rendra les pages interactives et fluides. C'est cette technologie qui permettra le déplacement des éléments d'une page. Une des innovations majeures sera le Flux RSS, qui nous permettra de rester informés des actualités d'une interface Web. Des technologies permettront également d'interconnecter les applications Web 2.0 entre elles : ce seront les mashups. Par exemple un site de partage de photos pourra être connecté à un site de cartographie en ligne. Cette nouvelle application nous permettra de consulter les photos en choisissant un lieu dans le monde.

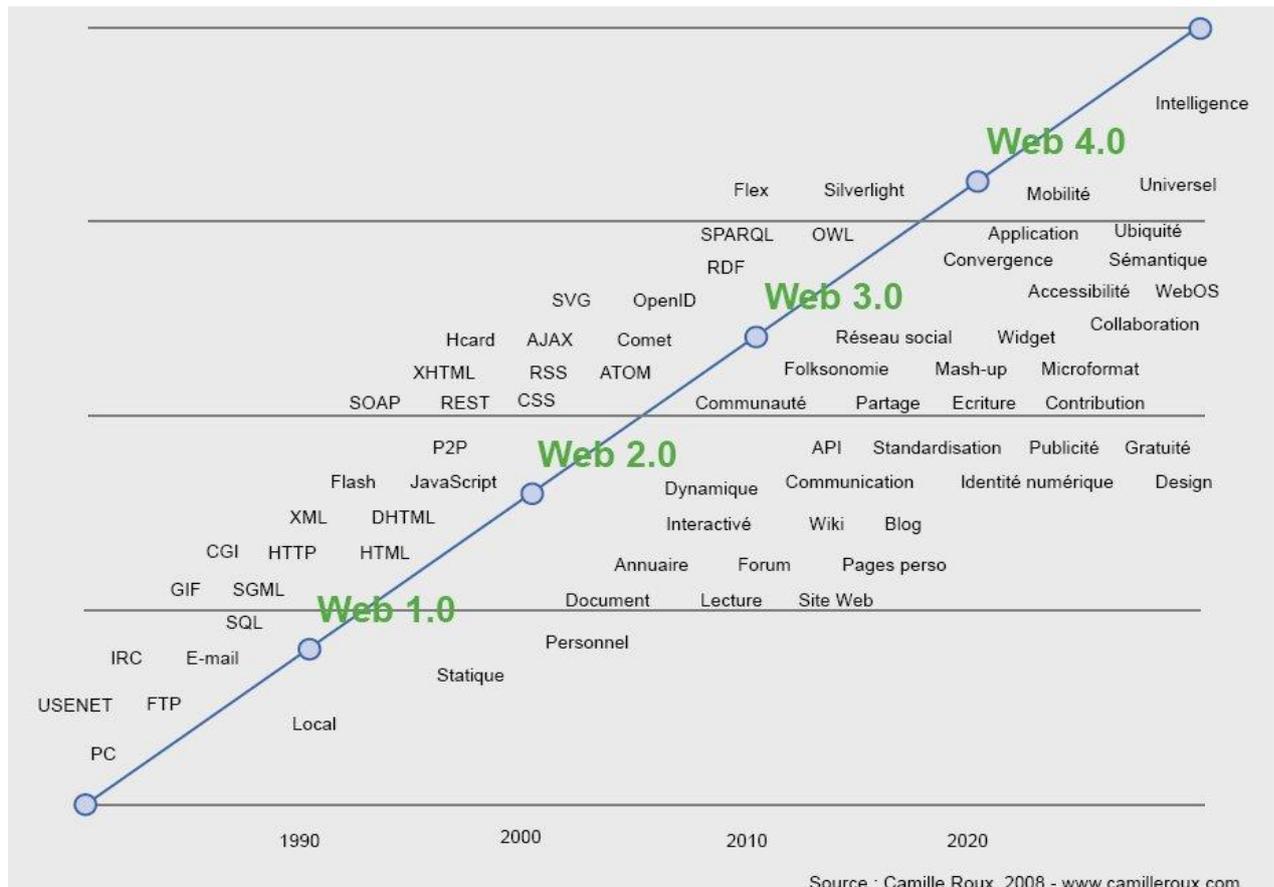


Figure 03. Évolution du Web.

9.4. Web 3.0 :

Le Web évolue, évolue vite. À peine installé, une nouvelle vague arrive... même plusieurs. En effet, le Web 3.0 n'en compte pas moins de trois en parallèle. Les changements apparaîtront sur des axes bien distincts : le Web sémantique, le Web mobile et les applications. Que nous allons voir brièvement.

9.4.1. Le Web mobile :

Le marché de l'Internet mobile est en pleine explosion. Les forfaits DATA illimités commencent à avoir des prix raisonnables, les téléphones facilitent de plus en plus la navigation sur Internet.

Avec plus de 28% de parts de marché aux États-Unis, l'iPhone a donné envie à de nombreux sites de développer des versions spécifiques, comme Facebook, Google.

Beaucoup décrivent le Web mobile comme une révolution à part entière du Web 3.0. Cependant, les mobiles risquent de s'adapter rapidement au Web classique. La course est peut-être inutile. L'avenir nous le dira...

9.4.2. Les applications Web :

Actuellement, une des tendances importantes du Web consiste à développer et adapter des applications uniquement connues jusqu'ici que sur nos bureaux en local. Depuis quelques temps, tout un tas d'applications Web fleurissent sur la Toile : traitement de texte, tableur, retouche photo, retouche vidéo, etc.

La plupart de ces sites ont utilisé et utilisent encore des technologies classiques du Web : HTML, CSS, AJAX... Cependant, pour obtenir un site très dynamique, ces langages deviennent de vrais freins au développement : problèmes de compatibilité entre navigateurs, développement et débogage très difficile vu l'hétérogénéité des langages, les interfaces graphiques ne sont pas aussi bien que celles vues habituellement sur les applications locales.

Mais comme à chaque fois, les techniques et les technologies évoluent afin de pallier les défauts des précédentes. Flex, créée par Macromedia en 2004 puis rachetée par Adobe en 2006, est le principal remplaçant. Il est à noter que Microsoft a lui aussi développé une technologie similaire, appelé Silverlight, mais elle reste encore jeune. Ces langages permettent de développer des applications Web exactement de la même façon que des applications locales. Le rendu est identique, voire meilleur. Il suffit de regarder une copie d'écran du traitement de texte récemment racheté par Adobe ou encore du logiciel de retouche photo Picnik pour s'en persuader. Le développement est plus rapide car il y a un langage côté client, un côté serveur et... c'est tout.

Tous les grands acteurs se sont jetés sur le marché des applications Web : Google, Microsoft, Adobe... Elles ont de nombreux avantages et beaucoup l'ont compris : pas d'installation, accessibles depuis n'importe quel ordinateur, pas de piratage, pas de problèmes de mise à jour...

Pour toutes ces raisons, les applications Web seront de plus en plus présentes sur la Toile. Les systèmes d'exploitation d'aujourd'hui deviendront peut-être de simples navigateurs Web...

9.4.3. Le Web sémantique :

L'Internet est aujourd'hui une gigantesque base de données très riche mais absolument pas organisée. Le seul soupçon d'organisation visible est sa structure en graphe orienté (généralisé par les liens hypertextes).

Pour s'y retrouver, il existe les annuaires (de moins en moins utilisés) et les moteurs de recherche. Ces derniers sont utilisés à longueur de journée pour trouver la moindre information. Ils apportent la plupart du temps la réponse à notre question quand elle est simple.

Cependant ils présentent tous certains défauts : Ils ne gèrent pas les homonymes. Par exemple, rechercher chat sur Google renvoie une liste de sites, certains concernant l'animal et d'autres les salons de discussions.

Ils ne gèrent pas non plus les synonymes. Si je cherche achat automobile, je suis aussi intéressé par les sites qui contiennent le mot voiture à la place d'automobile, mais aucun ne me les donnera dans les résultats.

Ils ne savent pas interpréter le langage naturel. Si je demande à Google quand Bill Gates est-il né ?, il tentera de trouver des sites contenant le plus possible d'expressions contenues dans la question. Mais cela ne donne aucune réponse pertinente.

Comme nous ne savons toujours pas apprendre une langue naturelle à un ordinateur (cela viendra peut-être un jour), c'est une fois de plus à l'homme de s'adapter.

Le Web sémantique consiste à ajouter des informations cachées destinées à être utilisées par des applications, des robots de moteurs de recherche, etc. Ces informations sont généralement présentées sous le format du Web sémantique, et c'est cette notion du web sémantique qui sera l'objet de notre étude dans notre prochain chapitre. [Camille Roux, 2008]

10. Conclusion :

Nous avons présenté dans ce chapitre une des principales applications d'internet qui est le « world wide web ». Nous avons commencé par retracer brièvement son historique, nous avons présenté ensuite ces concepts de base et les limites auquel est confronté le web.

Ensuite pour introduire notre prochain chapitre nous avons retracé l'évolution du web jusqu'à l'arrivé du web sémantique.

III

Le Web Sémantique

III. Le Web sémantique :

1. Introduction :

Le Web tel que nous le connaissons aujourd'hui est encore conforme à la vision qu'en avait Tim Berners-Lee il y a vingt ans : il s'agit d'un Web de documents. Ceux-ci sont écrits en HTML (Hypertext Markup Language), identifiés de manière unique par des URLs (Uniform Resource Locator) et reliés entre eux par des liens hypertextes. L'utilisateur surfe manuellement de page en page et peut depuis quelques années interagir avec le Web grâce aux technologies du Web 2.0 (Ajax). Cependant, l'information reste essentiellement textuelle et l'utilisateur ne voit que le sommet de l'iceberg : les données réelles, brutes et structurées, ne lui sont pas accessibles. Elles sont stockées, la plupart du temps, dans des bases de données et l'utilisateur n'en visualise que le rendu. Or toute la valeur du Web est en réalité dans ces données! Les exposer facilite la recherche de l'information ainsi que sa compréhension. L'étape suivante pour le Web est donc de pouvoir lier toutes ces données et de les combiner à loisir dans des applications composites. Le Web a besoin d'être équipé des technologies nécessaires à la création d'un Web de données (Web of Data). Depuis plusieurs années, toutefois, une nouvelle idée du Web prend corps : celle d'un Web Sémantique. C'est à partir de ce constat que les travaux du World Wide Web Consortium ont été orientés vers l'amélioration du Web afin de fournir de l'information pertinente aux utilisateurs, et de les décharger d'une grande partie de leurs tâches. Peu à peu, le World Wide Web Consortium (W3C) se dote de nouveaux langages et d'outils plus performants : XML, RDF, OWL, n'en sont que des exemples. Tous ont pour objectif commun de participer à une formalisation des savoirs, en permettant ainsi un meilleur partage et une transmission plus aisée.

3. Vers le Web Sémantique :

3.1 Problématique initiant au web sémantique :

En 1989, Tim Berners-Lee imagine pour le CERN² une architecture informatique distribuée permettant d'interconnecter les différents éléments du système d'information interne [Berners-Lee, 1989]. Il représente alors celui-ci comme un graphe où les noeuds, tout comme les arcs, sont typés et peuvent ainsi représenter (pour les noeuds) des outils, des documents, des projets ou des personnes ou bien encore (pour les arcs) des relations de production, d'inclusion

ou d'appartenance. Afin de faciliter la navigation dans un tel système, sa proposition se base sur l'utilisation de l'hypertexte, tel que défini par Ted Nelson dès les années 60 au sein du projet Xanadu3 [Nelson, 1965]. C'est cette proposition d'architecture décentralisée qui donnera par la suite naissance au World Wide Web tel que nous le connaissons aujourd'hui.

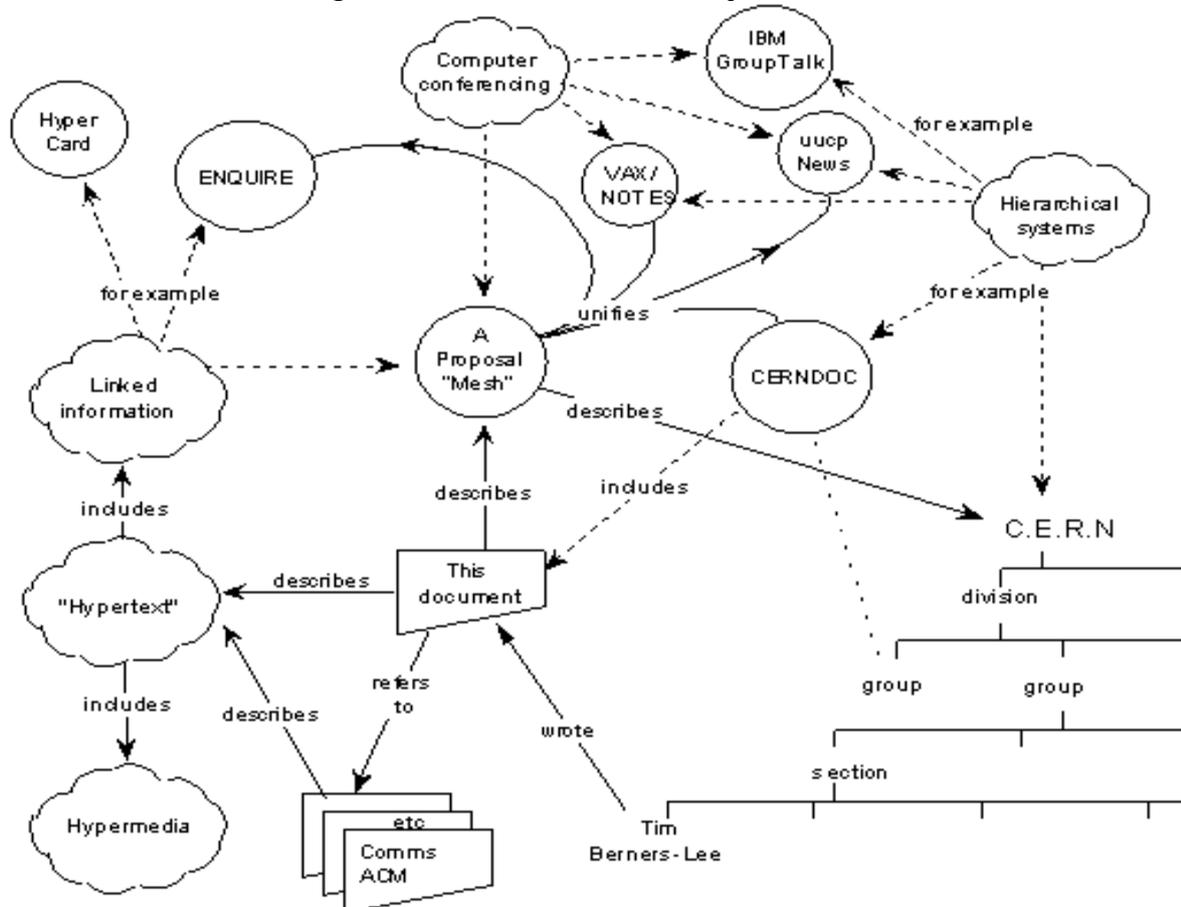


Figure 04: Proposition d'architecture distribuée qui conduira au World Wide Web [Berners-Lee, 1989]

Si l'on observe le schéma correspondant à cette vision d'origine du Web (Figure 04) et que l'on prend en compte l'état actuel de celui-ci, on ne peut s'empêcher de constater que là où la proposition initiale fait état de ressources et de liens fortement typés, le Web tel que nous le connaissons aujourd'hui ne considère que des documents, qu'ils soient textuels ou multimédia et des liens hypertextes non typés pour établir des relations entre ceux-ci. Ainsi, si un utilisateur est en mesure d'identifier le concept induit par un document (une personne ou un projet donné ...) ainsi que la nature du lien défini entre deux concepts (à partir des liens entre documents), cette identification n'est pas réalisable de manière simple par un agent logiciel. En effet, celui-ci ne considère que des documents plein-texte (encodés dans un langage dont il ne sait pas interpréter la sémantique) connectés entre eux par des hyperliens unidirectionnels non typés. De plus, les métadonnées associées à ces documents (auteur, date de création ...) sont elles aussi difficilement interprétables. Enfin, même pour un utilisateur, ces

interprétations peuvent être biaisées puisqu'elles font appel à l'expérience, la culture, et l'affect mental de celui-ci, qui peut différer selon les personnes pour un même document.

Ainsi se pose le problème d'un Web interprétable non seulement par les humains mais surtout par les machines (la machine ne comprend pas le sens du web, n'arrive pas à extraire et interpréter l'information utile). C'est en ce sens que se situe l'initiative du Web Sémantique qui vise à résoudre cette problématique d'interprétation des données par les agents logiciels.

Cette évolution du Web repose sur la présence d'annotations sémantiques, permettant de modéliser de manière formelle : les métadonnées (date de création, auteur, etc.) associées aux documents présents sur le Web et les données présentes au sein de ces documents.

Ces annotations sémantiques, qui permettent ainsi d'envisager l'interprétation des contenus en ligne, sont envisageables à partir du moment où l'on dispose :

- D'une part d'un modèle commun pour représenter les ressources sur le Web. C'est le rôle joué par l'utilisation des URIs (Uniform Resource Identifier) couplées à RDF (Ressource Description Framework).

- D'autre part de vocabulaires permettant de définir de manière formelle, mais surtout interprétable et interopérable, la sémantique de ces données. Les ontologies, au sens informatique du terme [Gruber, 1995], jouent ici un rôle important. Nous détailleront ces différents concepts dans la suite de ce document.

3.2 Qu'est ce que le web sémantique?

Le web sémantique est actuellement un Vaste chantier de recherche, mobilisant de nombreux acteurs, lancé et piloté par le W3C pour pallier les insuffisances actuelles du web.

3.2.1. Définition:

Le Web sémantique (plus techniquement appelé « le Web de données ») permet aux machines de comprendre la sémantique, la signification de l'information sur le Web. Il étend le réseau des hyperliens contenus dans les pages Web classiques par l'insertion de métadonnées lisibles par la machine et relie ainsi les données entre elles, au lieu des pages (ou documents) comme c'est le cas actuellement, permettant ainsi aux agents automatisés d'accéder plus intelligemment aux différentes données contenues sur le Web et, de cette manière, d'effectuer des tâches (recherche, apprentissage, etc.) plus précises pour les utilisateurs. [Julien Plu, 2011]

Le terme a été inventé par Tim Berners-Lee, co-inventeur du Web et directeur du W3C, qui supervise l'élaboration des propositions de standards du Web sémantique.

Il a défini le Web sémantique comme suite :

« Le web sémantique est une extension du web courant, dans lequel on donne à une information un sens bien défini pour permettre aux ordinateurs et aux personnes de travailler en coopération ». [Tim Berners-Lee et al., 2001]

Concrètement, le Web sémantique peut être vu comme une infrastructure au sein de laquelle chaque ressource est couplée à des métadonnées définies à l'aide de langages qui, permettent de décrire les ressources ainsi que les relations. Pour réaliser cette infrastructure, les ontologies ont été introduites.

L'ontologie est une technologie clé pour la réalisation du Web sémantique, et constitue une manière de représenter la sémantique dans le Web sémantique.

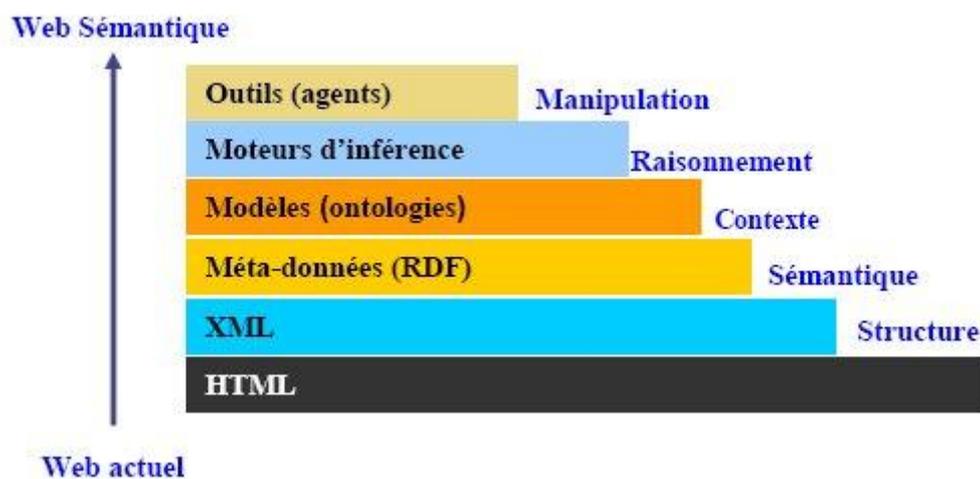


Figure 05: Evolution du Web actuel vers le Web sémantique

En peut résumer la différence dans le tableau suivant :

| Web actuel | Web sémantique |
|---------------------------------|-----------------------------|
| ✓ Ensemble de documents | ✓ Ensemble de connaissances |
| ✓ Basé essentiellement sur HTML | ✓ Basé sur XML et RDF(S) |
| ✓ Recherche par mots clé | ✓ Recherche par concepts |
| ✓ Utilisable par l'humain | ✓ Utilisable par la machine |

Tableau 01 : Comparaison entre le Web actuel et le Web sémantique

3.2.2. Propriétés :

On peut retenir quatre propriétés importantes du web sémantique:

- **Formalisé:** le web informel est déjà disponible, c'est la formalisation qui fait le web sémantique. Bien entendu, il faut savoir lier le formel à l'informel [Euzenat, 1996].
- **Ouvert:** le web ne peut pas être sémantique dans une organisation et syntaxique à l'extérieur. Ce qui est important est que ce qui peut circuler,

ce qui peut être appréhendé soit formalisé. C'est à cette condition qu'il sera possible aux machines d'exploiter le web sémantique et, en particulier, de mettre en relation les ressources.

- **Interopérable:** car il ne s'agit pas de communiquer dans un langage formalisé mais de savoir manipuler correctement cette connaissance formalisée.

- **Personnel:**
 - parce que le web a d'abord démarré à partir d'individus qui avaient des ressources à partager,
 - parce que l'informatique n'est plus dirigée par les besoins des grands groupes mais par le grand public (c'est grâce aux jeux que les processeurs sont rapides),
 - parce que c'est ainsi que se transmet la connaissance (d'individu à individu, en peer-to-peer).

3.3 Historique du Web sémantique :

Si le Web sémantique commence à être médiatisés ces dernières années, ses fondements sont plus anciens, ils remontent aux origines même du Web et ont connu plusieurs évolutions dont l'histoire permet de mieux appréhender les enjeux.

1989-1994 : la création du Web

Le Web est né au CERN, le centre européen de recherche nucléaire, à la fin des années 1980 porté, entre autres, par Tim Berners-Lee, alors chercheur dans ce laboratoire. Il part d'un constat simple : l'absence de cadre d'interopérabilité pour échanger dans un espace de machines en réseau les documents et les données contenus dans les ordinateurs des chercheurs du CERN. Pour régler ce problème, Tim Berners-Lee propose la mise en place d'un dispositif technologique pour mettre à disposition, lier et partager des documents sur un réseau de machines connectées composé de quatre briques technologiques :

- Un protocole de communication, HTTP, basé sur le protocole TCP/IP, c'est-à-dire Internet ;
- Un mécanisme d'identification, URL, qui permet d'atteindre un document sur un réseau distribué de machines ;
- Un principe de mise en relation des documents, l'hypertexte créé à l'issue de la seconde guerre mondiale par Vanevar Bush et adapté à l'informatique par Ted Nelson au milieu des années 1960 ;
- Un langage d'encodage des documents, HTML basé sur SGML, une norme de structuration hiérarchique de l'information.

Si ces quatre briques technologiques sont à l'origine du Web de documents que nous connaissons aujourd'hui, la proposition initiale de Tim Berners-Lee contenait également la mise en relation des données structurées contenues dans les bases de données des chercheurs. Néanmoins, de ce point de vue, les propositions étaient alors moins concrètes.

1994-2001 : la mise en place théorique du Web sémantique

En 1994, lors de la première conférence WWW à Genève, plus précisément au CERN, a lieu l'annonce de la création du W3C. C'est d'ailleurs à cette période que Tim Berners-Lee dresse les objectifs du W3C et montre les besoins d'ajouter de la sémantique au Web futur. Il montre alors en quoi les liens hypertextes ou, plus précisément, la façon dont on met en relation les documents sur le Web est dépassée pour permettre aux machines de relier automatiquement les données contenues sur le Web à la réalité. De par sa grande ambition, cette idée n'en rencontre pas moins quelques problèmes existants et tout particulièrement dans le domaine de l'intelligence artificielle.

Après cette conférence, à part la mise en place des recommandations nécessaires à la construction des documents, le W3C nouvellement créé entame les premières réflexions sur le démarrage du Web sémantique. Ces réflexions aboutissent à la publication d'un premier *draft* de recommandations sur le Web sémantique en octobre 1997 et d'un second en avril 1998. Cette même année, Tim Berners-Lee publie un document sur les toutes premières moutures de ce qui sera plus tard appelé le Web sémantique. Ces moutures consistent à élaborer les différentes technologies du Web sémantique. Dans ce document, il présente le Web sémantique comme une sorte d'extension du Web des documents, qui constitue une base de données à l'échelle mondiale, afin que toutes les machines puissent mieux lier les données du Web. Cette feuille de route se matérialise par une représentation graphique, le « layer cake », qui montre l'agencement des différentes briques technologiques du Web sémantique. (On verra le « layer cake » quand en étudiera l'architecture du WS).

Par ailleurs, en 1999, Tim Berners-Lee publie le livre *Weaving the Web* dans lequel il dresse un portrait du Web et les pistes pour son avenir. Les idées du Web sémantique n'en sont évidemment pas absentes. C'est d'ailleurs cette même année qu'il énonça sa célèbre citation :

“I have a dream for the Web [in which computers] become capable of analyzing all the data on the Web the content, links, and transactions between people and computers. A "Semantic Web", which should make this possible, has yet to emerge, but when it does, the day-to-day mechanisms of trade, bureaucracy and our daily lives will be handled by machines talking to machines. The "intelligent agents" people have touted for ages will finally materialize”.

J'ai fait un rêve pour le Web [dans lequel les ordinateurs] deviennent capables d'analyser toutes les données sur le Web - le contenu, les liens et les transactions entre les personnes et les ordinateurs. Un « Web sémantique », qui devrait rendre cela possible, n'est pas encore sorti, mais, quand ce sera le cas, les mécanismes d'échange au jour le jour, la bureaucratie et notre vie quotidienne seront traitées par des machines qui parlent à d'autres machines.

Certains nous ont vanté depuis des lustres les « agents intelligents » et cela va enfin se concrétiser. [Julien Plu, 2011]

2001-2011 : le long chemin vers le Web de données

Les différents travaux engagés depuis 1994 sont présentés pour la première fois au grand public à l'occasion d'un article publié dans la revue Scientific American en mai 2001. Écrit par Tim Berners-Lee, Ora Lassila et James Hendler, cet article présente un cas d'utilisation et les différentes technologies nécessaires à son accomplissement. Si cet article permet une introduction pédagogique aux objectifs poursuivis par le Web sémantique, il n'en reste pas moins exploratoire, trop peut-être. De plus, comme James Hendler l'avouera plus tard, il présente le défaut de reprendre certains concepts ou technologies, en particulier le principe des ontologies, qui renvoient aux problématiques de l'intelligence artificielle dont les fantasmes se sont transformés pour le grand public en espoir déçu. Enfin, le mot « sémantique » de par sa polysémie n'aide pas à une compréhension immédiate du concept et des objectifs visés.

Malgré les avis dubitatifs et les critiques grandissantes, le W3C continue le travail de normalisation avec la publication de recommandations essentielles : RDFS, OWL et une révision de RDF en 2004, SPARQL en 2008 et RDFa en 2008 sur lesquelles nous reviendrons plus en détail.

A partir de 2006, deux facteurs vont faire prendre au Web sémantique la direction qui est encore la sienne : Tout d'abord, le Web 2.0 marque l'apparition d'une réflexion dans la mise à disposition des données sur le Web via les Web services, des principes d'indexation collaborative (folksonomie), mais aussi de la structuration des données d'une page HTML avec le concept des microformats. Soient autant de sujets qui ont trait aux problématiques d'exposition, de structuration et de traitement des données structurées au cœur, également, de la réflexion sur les technologies du Web sémantique. Par ailleurs, conscient des malentendus engendrés par l'utilisation du mot « sémantique » et des concepts de l'intelligence artificielle, Tim Berners-Lee décrit dans une autre note l'idée du « Linked Data ». Il y rappelle, pour commencer, que le Web sémantique n'a pas vocation uniquement à poser des données dans le Web, mais à relier les données directement entre elles pour qu'une machine ou un humain puisse explorer le Web de données.

A la suite de cet article et à la vue de l'évolution du Web, deux chercheurs impliqués dans le Web sémantique et issus des domaines de l'intelligence artificielle et de la logique de description vont publiquement reconnaître leur erreur d'appréciation dans leur volonté d'introduire certaines notions complexes dans le Web sémantique. James Hendler parle de « côté obscur du Web sémantique » et avoue que l'introduction d'une logique de description complexe était une erreur stratégique. Chris Welty dans une keynote à ISWC 2007 intitulé « How I was right even when I was wrong » rappelle que l'important dans le Web sémantique, ce n'est pas la sémantique, mais le Web. Cette remise en question issue du bilan des recherches depuis 2001 aboutit à l'aveu de Tim Berners-Lee dans le magazine *La Recherche* en novembre 2007 :

« Le terme sémantique prête un peu à confusion car la sémantique s'intéresse au sens du langage pour en déduire des constructions logiques. Du coup, certains ont pensé qu'il s'agissait d'un Web qui permettrait par exemple d'effectuer des recherches sur Internet en posant des questions sous forme de phrases, en langage naturel. Or ce n'est pas son but. En fait, nous aurions dû l'appeler dès le départ "Web de données". »

Soutenue par le projet « Linking Open Data » piloté par le W3C, l'idée du *Linked Data* (qu'on traduira par Web de données) connaît sa plus importante réalisation dès février 2007 avec la création de **Dbpedia** par deux universités allemandes. Ce projet met à disposition selon les règles édictées par Tim Berners-Lee et, par conséquent, avec les technologies du Web sémantique, les données structurées extraits automatiquement de Wikipedia. En rencontrant une des réussites les plus médiatiques du Web 2.0, le Web de données acquiert immédiatement une base de travail solide mais aussi une bonne visibilité auprès des spécialistes du Web et de son évolution.

En 2008, il ya eu la médiatisation du web sémantique autour de quelques produits et acteurs de référence. On retrouve **SearchMonkey** (Yahoo), **Twine**, **Powerset** (Microsoft), **OpenCalais** (Reuters), le projet **DataPortability** ou encore l'API « **Google_Social_Graph** ».

Néanmoins, il faudra attendre 2009 et la communication de Tim Berners-Lee à la conférence TED au cours de laquelle il lance son appel « *Raw Data Now* » pour voir le Web de données atteindre une très large audience. Elle se manifestera en 2010 par l'élaboration du projet de mise à disposition des données gouvernementales britanniques, *data.gov.uk*, dirigé par Tim Berners-Lee et Nigel Shadbolt et basé pour une large partie sur les technologies du Web sémantique. [<http://www.archivesdefrance.culture.gouv.fr/thesaurus/historique-web-semantique.html>]

3.4. Fondements du web sémantique :

Plusieurs formalismes ont contribué au fondement du web sémantique à savoir : les logiques de description, les cadres ...etc.

3.4.1. Les logiques de description :

Une logique de description est principalement un formalisme pour la représentation de connaissance [Brachman et Levesque, 1984]. Elle s'oriente plus vers une représentation à base de classes et de propriétés. Dans ce contexte, les logiques de description sont considérées comme le cœur des systèmes de base de connaissances :

D'une part, par leurs capacités à spécifier la structure d'une base de connaissances, d'autre part, les mécanismes de raisonnements associés.

3.4.1.1. Vue générale sur les logiques de description :

Afin de comprendre la façon dont la logique de description (**DL**) représente les connaissances, nous proposons un exemple simple qui représente les membres d'une famille. Nous proposons les concepts suivants : personne, femme, parent, mère (d'autres concepts peuvent être ajoutés). En analysant ces termes, on peut commencer par établir un ensemble de relations comme : une femme est une personne, une mère est forcément un parent, un parent est une personne. Ces affirmations sont traduites par l'établissement d'une hiérarchie de concepts. Cette dernière est assurée par la relation est-un (**Is-A**) qui constitue la base de la classification des concepts dans la logique de description. Cette hiérarchie prend le nom de subsumption.

La figure 06 présente une représentation graphique de la structure proposée.

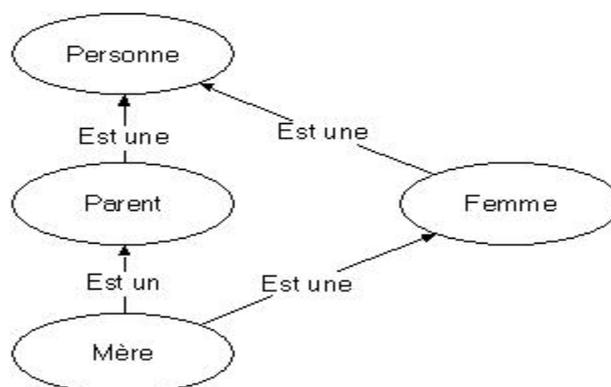


Figure 06 : Hiérarchie des concepts à base de la relation Est-Un

La *subsumption* qui résulte de l'utilisation de la relation *IS-A* fournit aussi la possibilité d'exprimer l'héritage des propriétés entre le subsumé et le subsumant. Ainsi, si le concept *Femme* a une propriété *âge*, le concept *mère* la possède systématiquement (comme dans le cas du paradigme *Orienté Objet*).

Néanmoins, l'intérêt des logiques de descriptions dépasse le simple fait d'exprimer la subsumption entre les concepts. En effet, les DL (Description Logic) permettent la représentation des relations sémantiques plus complexes, qui sont nécessaires pour une représentation fidèle des connaissances. Si on reprend l'exemple précédent, on peut l'étendre par la relation *A-Enfants* entre les concepts *parent* et *personne*. On peut même définir une cardinalité sur cette relation, dans le but de préciser la valeur minimale et maximale autorisées. (Cf. Figure 07).

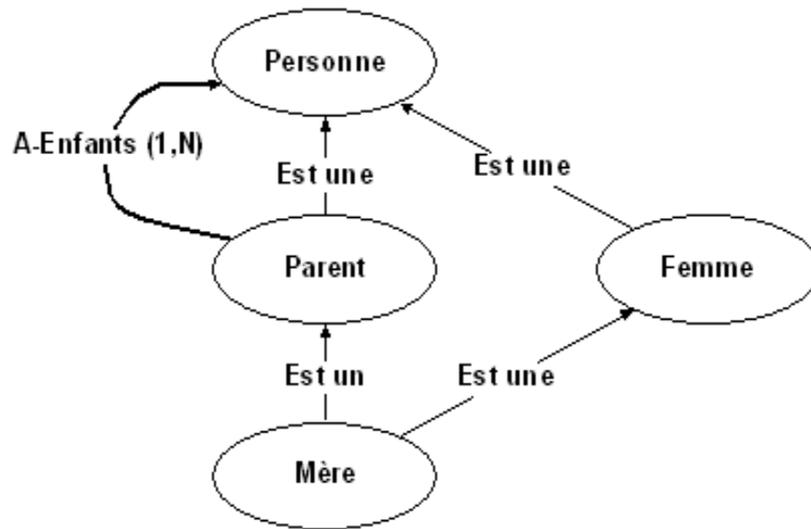


Figure 07 : Une hiérarchie en DL

Comme illustré dans la figure 07, les relations qui existent entre les concepts fournissent un enrichissement sémantique pour la base de connaissance. Mais, l'aspect sémantique ne se résume pas seulement aux relations explicites. En effet, d'autres relations implicites peuvent être constatées. Dans notre exemple, il n'y a pas une relation explicite entre *Mère* et *Personne*, mais on peut déduire qu'une mère est nécessairement une personne. Ce type de relations sémantiques implicites est le résultat d'un raisonnement approprié.

3.4.1.2. Syntaxe de la logique de description

La syntaxe du langage décrivant les logiques de description est basée sur un formalisme déjà connu dans d'autres logiques.

Ce langage se base sur la définition des éléments de base de l'alphabet, à savoir les *concepts atomiques* et les *rôles atomiques* qui désignent les relations entre les concepts. Les concepts atomiques servent à la création des autres concepts en utilisant des *constructeurs* spécifiques. Ces derniers donnent aussi une sémantique pour les nouveaux concepts en se basant sur leurs interprétations. Les principaux constructeurs sont ;

- La conjonction des concepts pour exprimer l'intersection entre deux concepts ($C \cap D$). La sémantique associée désigne l'ensemble des individus obtenu par l'intersection des individus des concepts *C* et *D*.

- La disjonction des concepts pour exprimer l'union entre deux concepts (CUD). La sémantique associée désigne l'ensemble des individus obtenu par l'union des individus des concepts C et D .
- La négation d'un concept pour exprimer son complément ($\neg C$). La sémantique associée désigne l'ensemble des individus qui appartiennent au domaine complément.

Dans l'exemple suivant, on propose un concept qui représente les *personnes qui ne sont pas des femmes*. Pour cela, on utilise la conjonction et le complément comme suit : $\text{Personne} \cap \neg \text{Femme}$.

L'établissement des relations entre les concepts est l'une des caractéristiques importantes de la logique de description. Alors, un langage pour la logique de description doit proposer les formalismes nécessaires pour exprimer les rôles et pour établir les restrictions nécessaires sur eux. Dans ce contexte, deux éléments sont fournis ; les *quantificateurs* et les *cardinalités*.

Les *quantificateurs* permettent d'établir des restrictions de nombres sur les rôles. On trouve principalement deux quantificateurs ; le quantificateur universel \forall et le quantificateur existentiel \exists . Par exemple, pour désigner un concept dont les individus ont tous des enfants féminins on peut l'écrire comme suit : $\forall \text{ A-Enfant .Femme}$.

Les *cardinalités* constituent un moyen important pour assurer une restriction des rôles. Elles permettent d'imposer des valeurs minimales ou maximales sur ces rôles.

Dans notre exemple concernant les membres d'une famille, on peut ainsi définir un concept qui a un nombre d'enfants entre 2 et 5 par : $\geq 2 \text{ A-Enfant} \cap \leq 5 \text{ A-Enfant}$.

Nous avons évoqué que l'intérêt de la logique de description réside aussi dans le raisonnement fourni. Ce dernier constitue la clé pour une utilisation réelle de la base de connaissance. Pour cela, le système de raisonnement se base sur deux entités ; le **TBox** et le **ABox**.

Le **TBox** constitue la partie terminologique de la base de connaissances. Il contient la déclaration des termes et des rôles. Chaque nouveau concept est défini à base des concepts déjà définis, pour fournir à la fin un ensemble de concepts de plus en plus complexes. Cet ensemble de concepts est classifié en vérifiant la subsumption des concepts qui positionne chaque concept à la bonne place dans la hiérarchie.

Le **ABox** contient les individus (ou instances) des concepts. Il utilise les concepts établis dans le **TBox** et permet de définir des instances pour chaque concept ou rôle. Par exemple, l'assertion suivante permet de définir la personne « Amina » comme étant une mère : $\text{Mère} \cap \text{Personne} (\text{Amina})$. Concernant les rôles, on peut définir l'assertion suivante : $\text{A-Enfant}(\text{Amina}, \text{Mohamed})$ qui indique que la personne « Amina » a un enfant « Mohamed ».

3.4.1.3. Quelques logiques de descriptions :

Il existe plusieurs familles de logiques de description. Nous présentons quelques unes, en commençant par les plus basiques pour aller vers les plus évoluées. Pour chacune d'elles, nous précisons les innovations apportées par rapport à celles qui la précèdent.

FL Cette logique de description a été proposée pour la sémantique des cadres de Minsky [Brachman et Levesque, 1984].

AL [Hollunder et al., 1990] est considéré comme la base de toutes les autres logiques. La Table 02 présente la syntaxe et la sémantique de cette logique.

| Constructeur | Syntaxe | Sémantique |
|----------------------------|---------------|---|
| Concept Atomique | A | $A^I \subseteq \Delta^I$ |
| Role atomique | R | $R^I \subseteq \Delta^I \times \Delta^I$ |
| Concept universel | \top | $\top^I = \Delta^I$ |
| Concept de base | \perp | $\perp^I = \emptyset$ |
| Négation atomique | $\neg A$ | $(\neg A)^I = \Delta^I \setminus A^I$ |
| Intersection | $C \sqcap D$ | $(C \sqcap D)^I = C^I \cap D^I$ |
| Quantificateur Universel | $\forall R.C$ | $\{a \in \Delta^I \mid \forall b. (a, b) \in R^I \rightarrow b \in C^I\}$ |
| Quantificateur existantiel | $\exists R.C$ | $\{a \in \Delta^I \mid \exists b. (a, b) \in R^I\}$ |

Table 02: Syntaxe et sémantique de la logique AL

On remarque que chaque constructeur ou axiome possède une sémantique appropriée. Cette dernière est spécifiée en utilisant la technique de *sémantique théorique du modèle* qui permet d'expliquer la relation entre la syntaxe du langage et le modèle voulu. Le model théorique est constitué principalement de deux éléments ;a) Le domaine (Δ^I) qui représente un ensemble d'objets, b) La fonction d'interprétation ($(\cdot)^I$) qui assure la correspondance entre :

- Les individus et les éléments du domaine. Par exemple, si **Nacer** est un individu, alors **Nacer**^j \in (Δ^I).
- Les classes et les sous-ensembles du domaine. Pour la classe **Personne**, on a **Personne**^j \subseteq (Δ^I).
- Les propriétés et les relations binaires du domaine. Par exemple, pour la propriété **Mère**, on a **Mère**^j \subseteq $\Delta^I \times \Delta^I$.

Notons que les objets dans le domaine n'ont pas un sens par leurs natures. Ce qui leur donne un sens, ce sont les relations entre les objets et les sous-ensembles. Par exemple, un individu z qui est une instance de la classe C est simplement interprété comme étant un élément de l'interprétation de C ($Z^I \in C^I$). La même situation existe pour les relations entre les classes. Si une classe C est

considérée comme une sous classe de D , alors l'interprétation de C est un sous-ensemble de l'interprétation de D ($C^j \subseteq D^j$)

ALC [Schmidt-Schaubß et Smolka, 1991] étend AL par la négation complexe des concepts. Par la suite, d'autres extensions ont été proposées en se basant sur ALC. Le résultat est une grande variété des logiques qui constituent la famille AL. Les extensions consistent à ajouter d'autres constructeurs afin d'enrichir la syntaxe. Chaque extension est représentée par un caractère spécial. C'est pour cela que chaque nouvelle logique possède un nouveau nom qui commence par AL puis une lettre appropriée. Par exemple, N pour la restriction des nombres, R pour la conjonction des rôles, R+ pour la transitivité des rôles etc. Horrocks détaille les caractéristiques de la famille ALC et propose une hiérarchie qui illustre les relations entre les logiques de cette famille [Horrocks, 1997].

La figure 08 présente les principales logiques proposées dans la famille ALC ainsi que les relations entre elles.

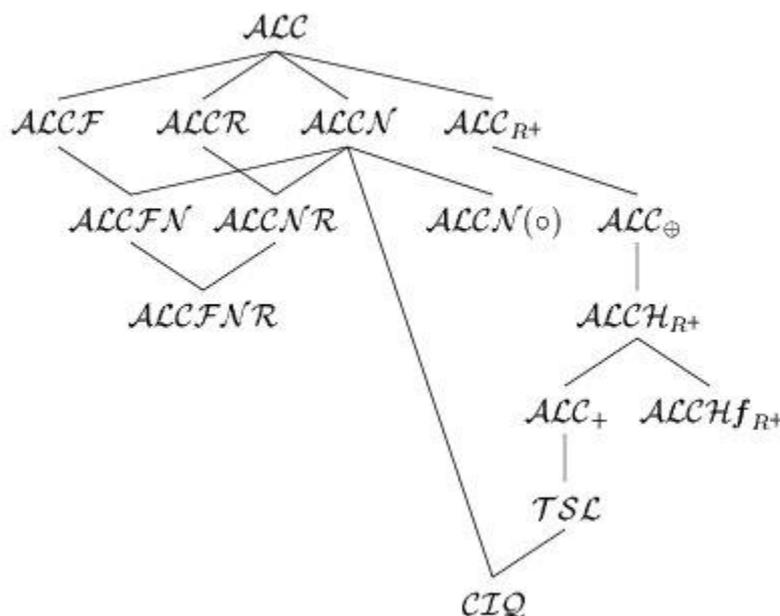


Figure 08 : Hiérarchie des logiques de la famille ALC

Ces logiques sont celles proposées avant les travaux de Horrocks. Ce dernier propose d'autres extensions qui constituent la base fondamentale de ce que va devenir le Web sémantique.

ALCHR+ étend la logique ALC par les rôles transitifs et la hiérarchie de rôles [Horrocks, 1997].

ALCIR+ étend ALC par les rôles transitifs et les rôles inverses. Elle a été proposée pour introduire une autre logique, à savoir ALCHIR+ [Horrocks et Sattler, 1999].

ALCHIR+ étend la logique **ALCIR+** par une hiérarchie de rôles [Horrocks et Sattler, 1999].

ALCHFIR+ [Horrocks et Sattler, 1999] se base sur **ALCHIR+** en ajoutant le support des restrictions fonctionnelles.

S est équivalente à **ALCR+**. La lettre **S** désigne toute logique de description qui supporte les rôles transitifs [Horrocks et al, 1999a]. L'utilisation d'une seule lettre **S** plutôt que **ALC R+** est motivée par une meilleure nomenclature des nouvelles logiques. On signale que cette logique n'est pas utilisée directement, elle constitue la base des autres logiques qui commencent par la lettre **S**.

SI est une extension de **S** en ajoutant le support des rôles inverses [Horrocks et al, 1999a], [Horrocks, et al, 2000a].

SHI est obtenue à partir de **SI** en ajoutant le support des rôles hiérarchiques [Horrocks et al, 1999a]. D'autres logiques, comme **SHN** et **SHQ** sont des variantes de **SH** dont le but est d'introduire les autres logiques et démontrer leur décidabilité.

SHIQ se base sur **SHI** en permettant en plus, des restrictions sur le « Qualifying number restriction » [Horrocks et al, 1999a], [Horrocks et al, 2000b]. Ainsi, on peut avoir des rôles de la forme ($\leq n R.C$) et ($\geq n R.C$), où **C** est un concept et **R** est un rôle simple qui peut être lui-même inverse. On mentionne le fait que la logique **SHIQ** est l'une des logiques les plus utilisées dans ce domaine.

SHIN est une restriction de la logique **SHIQ**. Elle impose une restriction sur « Qualifying number restriction » qui doit être de la forme ($\leq n R.T$) et ($\geq n R.T$) [Horrocks et al, 1999a].

SHIF. En plus de la restriction proposée par **SHIN**, elle impose que ces restrictions soient fonctionnelles de la forme : ($\leq 1 R$) et la négation ($\geq 2 R$) [Horrocks et al, 1999b].

SHOQ(D) Cette logique est proposée pour l'utilisation des types de données concrets [Horrocks et Sattler, 2001]. Elle se base sur la logique **SHQ** plutôt que sur **SHIQ** car le raisonnement avec les rôles inverses est plus difficile, spécialement quand on le combine avec les types concrets. L'introduction de la gestion des types de données concrets a poussé la séparation entre le domaine d'interprétation des concepts et celui des types. Cela se traduit par l'ajout d'un autre domaine d'interprétation spécialement pour les types de données. **SHOQ(D_n)** constitue une variante de **SHOQ(D)** en ajoutant le support des « n-ary datatypes predicate » [Pan et Horrocks, 2003].

SHOIN constitue la logique de base pour le Web sémantique actuel, spécialement le langage **OWL1.0** [Horrocks et al, 2003]. Vu son importance dans notre travail, nous allons lui consacrer une section à part.

SHOIQ est une extension de **SHOIN** [Horrocks et Sattler, 2005]. Elle permet d'exprimer des concepts de la forme ($\leq n R.C$) et ($\geq n R.C$) tandis que dans **SHOIN**, seuls ($\leq n R$) et ($\geq n R$) sont autorisés.

RIQ apporte une nouvelle caractéristique nommée « Propagation des propriétés » [Horrocks et Sattler, 2004]. RIQ se base sur la logique SHIQ en ajoutant le support de l'inclusion complexe des rôles. RIQ est décidable, contrairement à la logique indécidable SH+IQ qui constitue une première solution pour intégrer cette nouvelle caractéristique (la propagation des propriétés).

SRIQ se base essentiellement sur les deux logiques RIQ et SHIN [Horrocks et al, 2005]. Elle est considéré comme étant un travail en cours afin d'introduire la logique qui la succède, à savoir SROIQ.

SROIQ constitue actuellement une des dernières innovations dans le domaine des logiques de descriptions [Horrocks et al, 2006]. C'est la logique adoptée dans le Web sémantique pour la nouvelle version de OWL. SROIQ étend SHOIQ en ajoutant certaines propriétés telles que : la disjonction de rôles, la réflexivité, l'antisymétrique, l'assertion négative des rôles, inclusions complexe des rôles, la réflexivité locale de la forme $\exists .R.\text{Self}$. On remarque que les majeures innovations de SROIQ sont orientées vers le développement de l'aspect des rôles dans les logiques de description.

3.4.1.4. La logique SHOIN (D) :

La logique de description SHOIN(D) est similaire à la logique SHOQ(D) avec :

- 1) Une extension par les rôles inverses (I),
- 2) Une restriction pour unqualified number(N) [Horrocks et Patel-Schneider, 2003].

Une autre caractéristique importante de SHOIN(D) est la capacité de définir une classe en énumérant ses instances, par exemple on peut définir une classe comme suit : classe {Samedi, Dimanche, Lundi, Mardi, Mercredi, Jeudi, Vendredi} pour exprimer la classe semaine.

En plus de ces capacités et comme toutes les logiques de descriptions, SHOIN(D) possède une syntaxe et une sémantique appropriées, la table suivante présente l'ensemble des constructeurs et axiomes de SHOIN(D) :

| Constructor Name | Syntax | Semantics |
|--------------------------|-----------------------|---|
| atomic concept A | A | $A^I \subseteq \Delta^I$ |
| datatypes D | D | $D^D \subseteq \Delta_D^I$ |
| abstract role R_A | R | $R^I \subseteq \Delta^I \times \Delta^I$ |
| datatype role R_D | U | $U^I \subseteq \Delta^I \times \Delta_D^I$ |
| individuals I | o | $o^I \in \Delta^I$ |
| data values | v | $v^I = v^D$ |
| inverse role | R^- | $(R^-)^I = (R^I)^-$ |
| conjunction | $C_1 \sqcap C_2$ | $(C_1 \sqcap C_2)^I = C_1^I \cap C_2^I$ |
| disjunction | $C_1 \sqcup C_2$ | $(C_1 \sqcup C_2)^I = C_1^I \cup C_2^I$ |
| negation | $\neg C_1$ | $(\neg C_1)^I = \Delta^I \setminus C_1^I$ |
| oneOf | $\{o_1, \dots\}$ | $\{o_1, \dots\}^I = \{o_1^I, \dots\}$ |
| exists restriction | $\exists R.C$ | $(\exists R.C)^I = \{x \mid \exists y. \langle x, y \rangle \in R^I \text{ and } y \in C^I\}$ |
| value restriction | $\forall R.C$ | $(\forall R.C)^I = \{x \mid \forall y. \langle x, y \rangle \in R^I \rightarrow y \in C^I\}$ |
| atleast restriction | $\geq n R$ | $(\geq n R)^I = \{x \mid \#(\{y. \langle x, y \rangle \in R^I\}) \geq n\}$ |
| atmost restriction | $\leq n R$ | $(\leq n R)^I = \{x \mid \#(\{y. \langle x, y \rangle \in R^I\}) \leq n\}$ |
| datatype exists | $\exists U.D$ | $(\exists U.D)^I = \{x \mid \exists y. \langle x, y \rangle \in U^I \text{ and } y \in D^D\}$ |
| datatype value | $\forall U.D$ | $(\forall U.D)^I = \{x \mid \forall y. \langle x, y \rangle \in U^I \rightarrow y \in D^D\}$ |
| datatype atleast | $\geq n U$ | $(\geq n U)^I = \{x \mid \#(\{y. \langle x, y \rangle \in U^I\}) \geq n\}$ |
| datatype atmost | $\leq n U$ | $(\leq n U)^I = \{x \mid \#(\{y. \langle x, y \rangle \in U^I\}) \leq n\}$ |
| datatype oneOf | $\{v_1, \dots\}$ | $\{v_1, \dots\}^I = \{v_1^I, \dots\}$ |
| Axiom Name | Syntax | Semantics |
| concept inclusion | $C_1 \sqsubseteq C_2$ | $C_1^I \subseteq C_2^I$ |
| object role inclusion | $R_1 \sqsubseteq R_2$ | $R_1^I \subseteq R_2^I$ |
| object role transitivity | $\text{Trans}(R)$ | $R^I = (R^I)^+$ |
| datatype role inclusion | $U_1 \sqsubseteq U_2$ | $U_1^I \subseteq U_2^I$ |
| individual inclusion | $a : C$ | $a^I \in C^I$ |
| individual equality | $a = b$ | $a^I = b^I$ |
| individual inequality | $a \neq b$ | $a^I \neq b^I$ |

Table 03 : Syntaxe et sémantique de la logique SHOIN(D).

Un autre aspect très important dans SHOIN(D) est la gestion des types de données concrets, comme les entiers, les réels etc. Cette possibilité est extrêmement utile dans les applications pratiques. La logique SHOIN(D) intègre la gestion des types concrets en se basant sur les possibilités offertes par la logique SHOQ(D). En effet, cette intégration est assurée en faisant une séparation entre le domaine d'interprétation des classes et individus et celui des types de données. Pour cela, on introduit un nouveau domaine d'interprétation pour les types concrets qui est disjoint de celui du domaine d'interprétation des classes et individus. Dans ce nouveau domaine d'interprétation Δ_D^I , un type de données concret comme le type réel est interprété comme étant un sous-ensemble du domaine Δ_D^I . Une valeur réelle comme « 23,78 » est interprétée

comme un élément de Δ^1_D . Pour le cas des propriétés, il faut faire une différence entre les propriétés abstraites et celles de type concret. Par exemple, nous avons vu que la propriété abstraite **Mère** est interprétée comme une relation binaire dans le domaine Δ^1 (sous-ensemble de $\Delta^1 \times \Delta^1$). Or, pour une propriété de type de données concret comme **Age**, l'interprétation est différente. En effet la propriété **Age** est interprétée comme une relation binaire entre les domaines Δ^1 et Δ^1_D (sous-ensemble de $\Delta^1 \times \Delta^1_D$).

3.4.2. Autres influences sur le Web sémantique:

En plus des logiques de descriptions, d'autres paradigmes ont influencé le Web sémantique et qui sont nécessaire à sa compréhension et à son utilisation.

Le paradigme des cadres :

Dans ce paradigme [Minsky, 1975], le langage de formalisation utilise les *cadre* pour représenter une classe. Un cadre inclut le nom de la classe, identifie ses classes les plus générales et comporte un ensemble de slots. Ces derniers peuvent être des propriétés ou contraintes sur les valeurs. De point de vue sémantique, un cadre correspond à un axiome dans la logique de description. La notion d'hierarchie est supportée, chaque cadre est décrite comme étant une sous classe des classes plus générales. Cette structure est plus facile à comprendre et sa syntaxe abstraite a une correspondance directe avec la logique de description.

Le langage *RDF* (Resource Description Framework) [Lassila et Swick, 1999] est la première initiative dans la proposition d'un langage pour le Web sémantique. *RDF* permet de décrire la ressource selon le mécanisme de déclaration des triplets (en va voir ce langage en détail des les prochains chapitres).

3.5 Objectifs du Web sémantique :

Organisation responsable du devenir de la « Toile », le W3C et son président, Berners-Lee, ont été les premiers insatisfaits des nombreux inconvénients du Web, qui ont transformé celui ci en fourre-tout informationnel. Si le Web originel s'est révélé un fantastique outil pour la production, la publication et la diffusion de l'information, il n'a pu en revanche fournir encore les outils pour structurer et décrire les ressources de manière satisfaisante et permettre un accès pertinent à l'information.

L'objectif du **web sémantique** est donc d'offrir une **meilleure structuration du web** pour permettre et développer :

- l'intégration des sources d'information hétérogènes, des contenus structurés et dynamiques,
- une meilleure interopérabilité des ressources et des machines,
- un accès "intelligent" (sémantique, contextuel) à l'information,
- l'utilisation de connaissances formalisées,
- la capacité des machines à "raisonner" sur les contenus des ressources, à les exploiter et à les combiner afin de créer de nouveaux services.

Et cela va permettre de :

- transformer le web en un vaste espace d'échanges de ressources entre machines, permettant l'exploitation de grands volumes d'informations et de services variés
- libérer les utilisateurs d'une partie du travail de recherche et d'exploitation des résultatsetc.

3. Composants principale du Web Sémantique:

3.1. Les ontologies :

3.1.1 Introduction :

Durant les dernières années, la question de la conceptualisation des connaissances est devenue un thème central dans la littérature, et ce dans plusieurs domaines (philosophie, science médicale, informatique, etc.). Plusieurs raisons expliquent la nécessité de créer une base de connaissances fiable et utilisable. Parmi ces raisons, on peut citer : la structuration des données, l'enrichissement des données existantes par une sémantique appropriée, la proposition d'un vocabulaire commun dans un domaine, etc...

Ainsi, la conceptualisation est un des concepts majeurs pour satisfaire ces besoins. Dans ce contexte, la notion d'ontologie est considérée comme une approche intéressante pour assurer une gestion des connaissances, en couvrant les aspects sémantiques de cette dernière. Cependant, une compréhension précise de la notion d'ontologie est nécessaire pour une utilisation correcte. Précisément, si cette utilisation se fait dans un domaine tel que le Web sémantique. Ainsi, nous tenons à éclaircir cette notion afin de bien préparer le contexte de notre travail.

3.1.2 Origine et définitions :

Le terme « ontologie » a suscité beaucoup d'intérêt dans plusieurs domaines et disciplines. La définition et l'utilisation de ce terme a subi des changements suivant son utilisation et selon les disciplines qui le traitent.

Historiquement, l'ontologie est un terme philosophique qui signifie la science et la théorie de l'être. L'origine de ce terme est grec, c'est la composition de deux autres termes, **On** signifie : être et **Logos** signifie science.

Le terme « Ontologie » est introduit pour la première fois au 17ème siècle pour désigner la science de l'être en général. Puis, il a été étendu pour devenir « l'étude de tout ce qui est en rapport avec la recherche sur l'être en tant qu'être (**on hèn on** en grec) ». La diffusion de ce terme a pris une grande proportion grâce à l'ouvrage *Ontologia* de [Wolff, 1729] qui positionne l'ontologie dans le cadre de la métaphysique générale en la distinguant des autres classes de la métaphysique qui sont ; la théologie, la cosmologie et la psychologie rationnelle. Par conséquent, on remarque que le terme ontologie appartient principalement au domaine philosophique.

Malgré l'aspect philosophique de la notion d'ontologie, elle a suscité de l'intérêt dans l'informatique, et spécialement dans le domaine de l'intelligence artificielle. C'est pour cela, que [McCarthy, 1980] était l'un des premiers à affirmer, que les concepteurs des systèmes intelligents fondés sur la logique devaient d'abord énumérer tout ce qui existe en construisant une ontologie de notre monde.

Après l'introduction de la notion de l'ontologie dans le domaine de l'intelligence artificielle, une succession de définitions a eu lieu dans le but d'éclaircir ce concept.

[Neches et al, 1991] affirme qu'une ontologie définit les termes et les relations de base qui constituent le vocabulaire d'un domaine donné. En plus, l'ontologie doit établir les règles qui permettent la combinaison de ces termes et relations, afin de définir une extension de ce vocabulaire.

[Gruber, 1993] propose une définition considérée comme la plus répandue dans la littérature. Il affirme qu'une « *ontologie est une spécification explicite d'une conceptualisation* ». Cette définition a été approfondie dans le travail de [Borst, 1997] où il donne la définition suivante : « *Une ontologie est définie comme étant une spécification formelle d'une conceptualisation partagée* ».

Les concepts annoncés dans ces deux définitions ont été expliqués par [Studer et al, 1998] comme suit ;

Explicite : définition explicite des types de concepts utilisés et de contraintes sur leur usage.

Conceptualisation : une certaine vue du monde par rapport à un domaine, souvent conçu comme un ensemble de concepts, leurs définitions, leurs interrelations.

Formel : compréhensible par la machine.

Partagée : consensus, connaissance acceptées par un groupe.

Nous retiendrons dans le domaine du Web sémantique la définition suivante [CHARLET J. 2002] : « Une ontologie est une spécification normalisée représentant les classes des objets reconnus comme existant dans un domaine. »

L'ontologie doit être exploitable par un ordinateur tout en faisant sens pour les humains, définir de manière normalisée et consensuelle les concepts d'un domaine tout en répondant aux besoins de l'application visée.

3.1.3. Les composants d'une ontologie :

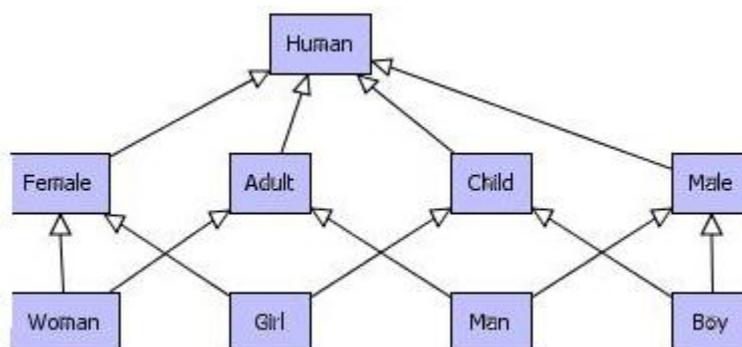
Les connaissances traitées par l'ontologie sont modélisées à travers des éléments propres à cette dernière. Ces composants sont principalement concepts, relations, fonctions, axiomes et instances [Gomez-Perez, 1999].

- **Les concepts** peuvent être une pensée, un principe, une notion profonde. Ils sont appelés aussi termes ou classes de l'ontologie, et peuvent être classifiés selon plusieurs dimensions :
 - Niveau d'abstraction (concret ou abstrait).
 - Atomicité (élémentaire ou composée).
 - Niveau de réalité (réel ou fictif).
- **Les relations** constituent les associations qui peuvent exister entre les concepts. Elles assurent une certaine interaction entre les différents concepts. La relation *sous-classe* est un exemple d'une relation binaire entre deux classes.
- **Les fonctions** sont un cas particulier des relations dont un élément d'une relation est unique par rapport aux éléments qui le précèdent. Un exemple d'une fonction binaire est *Mère-de* qui donne la mère d'un individu. Ce dernier, doit avoir une seule mère.
- **Les axiomes** modélisent les connaissances considérées comme vrais dans le domaine traité.
- **Les instances** constituent des valeurs concrètes et des occurrences pour les concepts et les relations par exemple les individus « Mourad » et «Wali» sont des instances du concept « personne ».

Exemple illustrant les composants d'une ontologie :

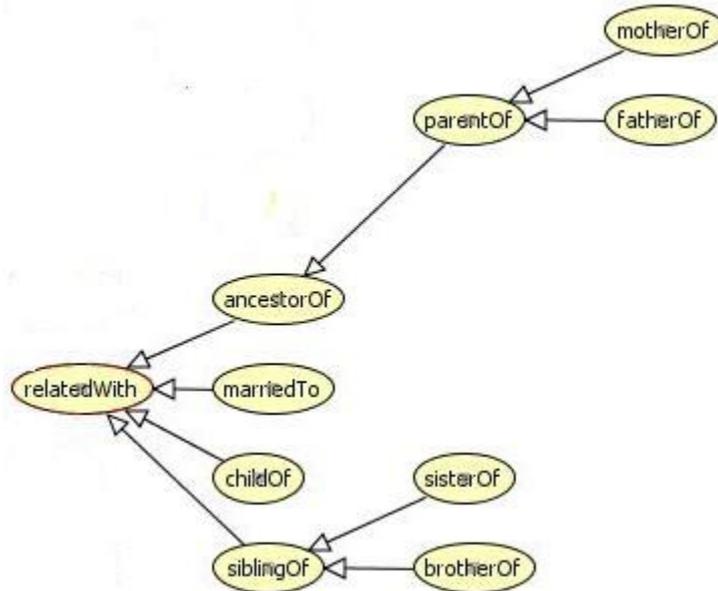
Faisons un petit exemple d'ontologie sur une famille.

Tout d'abord en a les concepts : {Human, Child, Boy, Girl, Male, Man, Female, Woman, Adult} qui nous donne la représentation suivante entre ces concepts :



Ensuite, occupons-nous des relations: {relatedWith (Human, Human), siblingOf(Human, Human), sisterOf(Female,Human), brotherOf(Male,Human), ancestorOf(Adult, Human), parentOf(Adult, Human), fatherOf(Man, Human), motherOf(Woman, Human), marriedTo(Adult, Adult), childOf(Human, Adult)}.

Ce qui nous donne la représentation suivante entre ces relations :



Maintenant admettons quelques règles (axiomes), logique pour un humain sur ces concepts et relations :

- **règle 1** : si un concept "Male" appartient à une relation "brotherOf" avec un concept "Female" alors le concept "Female" appartient à la relation "sisterOf" avec le concept "Male" ;
- **règle 2** : réciproque de la règle 1 ;
- **règle 3** : si un concept "Adult" appartient à une relation "parentOf" avec un concept "Human" alors le concept "Human" appartient à la relation "childOf" avec le concept "Adult" ;
- **règle 4** : réciproque de la règle 3.

De cette manière, on peut facilement imaginer toutes les règles possibles entre ces relations et concepts.

Pour finir prenons quelques individus : {Bart, Lisa, Homer}. On dit ensuite que l'individu "Bart" appartient(ou instance du) au concept "Boy", "Lisa" appartient au concept "Girl" et "Homer" appartient au concept "Man".

On dit ensuite que "Bart" est le frère de "Lisa" et que "Bart" et "Lisa" sont les enfants de "Homer" avec nos relations.

Notre ontologie ainsi construite, il nous est maintenant facile de poser les requêtes suivantes :

- est-ce que Lisa est la sœur de Bart ?
- est-ce que Homer est un parent de Lisa et Bart ?

Bien entendu cette ontologie est très petite et facile comparée aux ontologies de domaines plus complexes.

3.1.4. Cycle de vie d'une ontologie

Les ontologies 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 [Furst , 2002].

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). S'y ajoutent des activités transversales de support telles que l'évaluation, la documentation, la gestion de la configuration [Blazquez, 1998].

Un cycle de vie inspiré du génie logiciel est présenté dans [Furst , 2002].

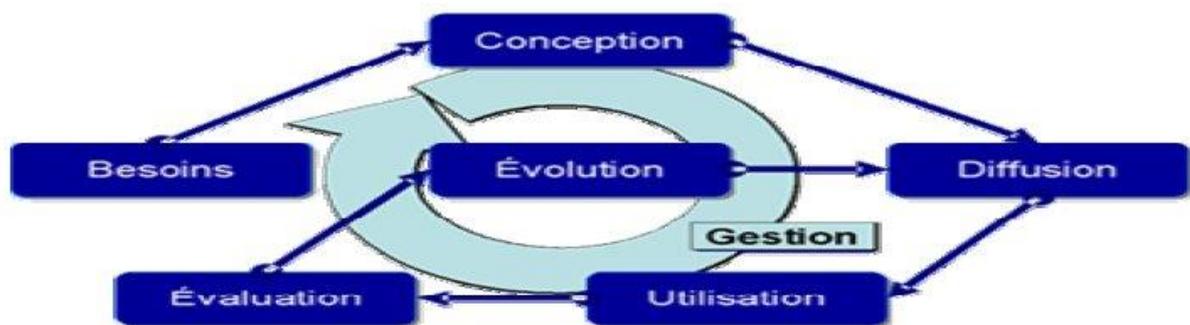


Figure 09 : Cycle de vie d'une ontologie

3.1.5. Rôles de l'ontologie :

Cette section présente les rôles et l'intérêt de l'ontologie dans le domaine informatique en générale et du web sémantique en particulier. Nous allons énumérer les plus importants. [Djaghloul Younes.2007]

Vocabulaire commun : afin de décrire un domaine particulier, un vocabulaire doit être proposé. La proposition d'un tel vocabulaire constitue le rôle fondamental d'une ontologie.

Structuration des données : L'ontologie peut être considérée comme une base de concepts. Ainsi elle fournit une classification et une structuration des données du domaine.

Explication de l'implicite : souvent les connaissances sont implicites. Cette situation peut causer des ambiguïtés dans un éventuel traitement. Dans ce contexte, l'ontologie explicite les connaissances implicites.

Proposition d'un méta-modèle : Par définition, un modèle constitue une abstraction d'une réalité. Comme la notion d'ontologie se base sur la définition

des concepts et des relations, elle s'intègre parfaitement dans une solution pour la création des modèles.

Interopérabilité sémantique : L'interopérabilité entre les systèmes d'informations hétérogènes constitue un défi majeur. Dans ce contexte, l'ontologie constitue une solution intéressante pour ce problème. Elle permet de définir clairement les différents concepts utilisés, en vue d'assurer une communication sans ambiguïté entre les systèmes hétérogènes.

3.1.6. Classification des ontologies :

La classification des ontologies permet de positionner chacune d'elles dans un contexte défini. Néanmoins, chaque ontologie peut être classée selon plusieurs dimensions. Ainsi, une classification judicieuse doit prendre en compte une classification multidimensionnelle.

Dans cette section, sont données les principales classifications des ontologies en investissant trois dimensions, à savoir ; objet de conceptualisation, niveau de détail et niveau du formalisme de représentation. [Djaghloul Younes.2007]

3.1.6.1. Dimension « objet de conceptualisation »

La classification selon l'objet de conceptualisation se divise principalement en six types :

Ontologie de haut niveau : L'ontologie de haut niveau est une ontologie générale. Elle s'intéresse à l'étude des catégories existantes dans notre monde (l'espace, le temps, les processus, les états...). Selon Guarino, une ontologie de haut niveau est basée sur trois théories fondamentales : la théorie de l'identité, la théorie de dépendance et la méréologie (étude des relations entre les parties et la totalité) [Guarino, 1997].

Sowa étend cette vision en introduisant deux autres concepts ; *Continuant* et *Occurrent* [Sowa, 1996]. En utilisant ces concepts et en combinant sept primitives, il obtient douze catégories dans l'ontologie de haut niveau qu'il propose.

Ontologie générique : L'ontologie générique appelée aussi **méta-ontologie** utilise des concepts moins abstraits que l'ontologie de haut niveau. Mais ces concepts sont assez généraux pour qu'ils puissent être utilisés dans des domaines précis [VanHeijst et al, 1997]. Ce type d'ontologie s'adresse principalement à résoudre des problèmes génériques pouvant être utilisé à travers plusieurs domaines. Borst propose plusieurs ontologies de cette catégorie [Borst, 1997]. Il donne comme exemple l'ontologie **méréologique** et l'ontologie **topologique**.

Ontologie de représentation de connaissances : Le rôle de cette ontologie est de capturer les concepts permettant la représentation et la formalisation des

connaissances. [Gruber, 1993] propose les primitives utilisées dans les langages de frames (Classe, propriété, facette, instance...).

Ontologie de domaine : L'ontologie de domaine propose l'ensemble des termes et des concepts d'un domaine, comme scalpel ou scanner dans la médecine. Selon Mizoguchi, l'ontologie de domaine caractérise les connaissances du domaine où la tâche est réalisée [Mizoguchi et al, 1995].

Ontologie de tâche : Selon [Mizoguchi et al, 2000], le rôle de cette ontologie est de caractériser les connaissances concernant une tâche particulière. Cette dernière n'appartient pas nécessairement à un seul domaine. Un exemple de cette ontologie pour le travail collaboratif est fourni dans le travail de [Inaba et al, 2000].

Ontologie d'application : Cette ontologie est considérée comme la plus spécifique. Elle traite les connaissances liées à une application déterminée. Un nombre important des ontologies proposées appartient à cette catégorie. Maedche propose un ensemble d'ontologies pour la gestion des connaissances au sein de l'entreprise [Maedche et al, 2003].

3.1.6.2. Dimension « détail de l'ontologie »

Cette dimension de classification permet de donner un degré de détail approprié pour l'ontologie proposée. Elle permet aussi d'estimer son raffinement et sa complexité.

On peut proposer deux niveaux de granularité pour exprimer ce degré de détail : granularité large et granularité fine.

Granularité large : Dans cette catégorie, même si l'ontologie traite un domaine ou une application précise, le vocabulaire est de nature générale. C'est le raffinement qui permet d'aboutir à des ontologies plus détaillées. On peut considérer une ontologie de granularité large comme étant une première version dans un processus de développement d'ontologies.

Granularité fine : Dans cette catégorie, l'ontologie possède un vocabulaire riche et détaillé. Elle permet de décrire de façon précise les concepts utilisés. Toute ontologie qui veut être opérationnelle, doit appartenir à cette catégorie.

3.1.6.3. Dimension « niveau de formalisme »

Cette dimension permet de classer les ontologies selon le niveau de formalisme du langage de représentation utilisé. [Uschold et Gruninger, 1996] traite cet aspect en indiquant que pour représenter une ontologie, on peut utiliser un langage formel ou informel. Une classification en quatre types de langage est donnée ; informel, semi-informel, semi-formel et formel.

Le langage informel représente le langage naturel. Ce langage est utilisé seulement dans la phase de discussion afin de présenter une vue globale de l'ontologie.

Le langage semi-informel représente un langage naturel limité et structuré en éliminant d'éventuelles ambiguïtés.

Le langage semi-formel représente un langage artificiel défini formellement. Le langage de modélisation *UML* (Unified Modeling Language) constitue un exemple de ce type. *UML* se base sur un langage graphique avec une définition rigoureuse pour les notations proposées.

Le langage formel représente un langage artificiel contenant une sémantique formelle et des théorèmes et preuves sur ces propriétés.

Afin de résumer la classification, nous proposons la figure suivante :

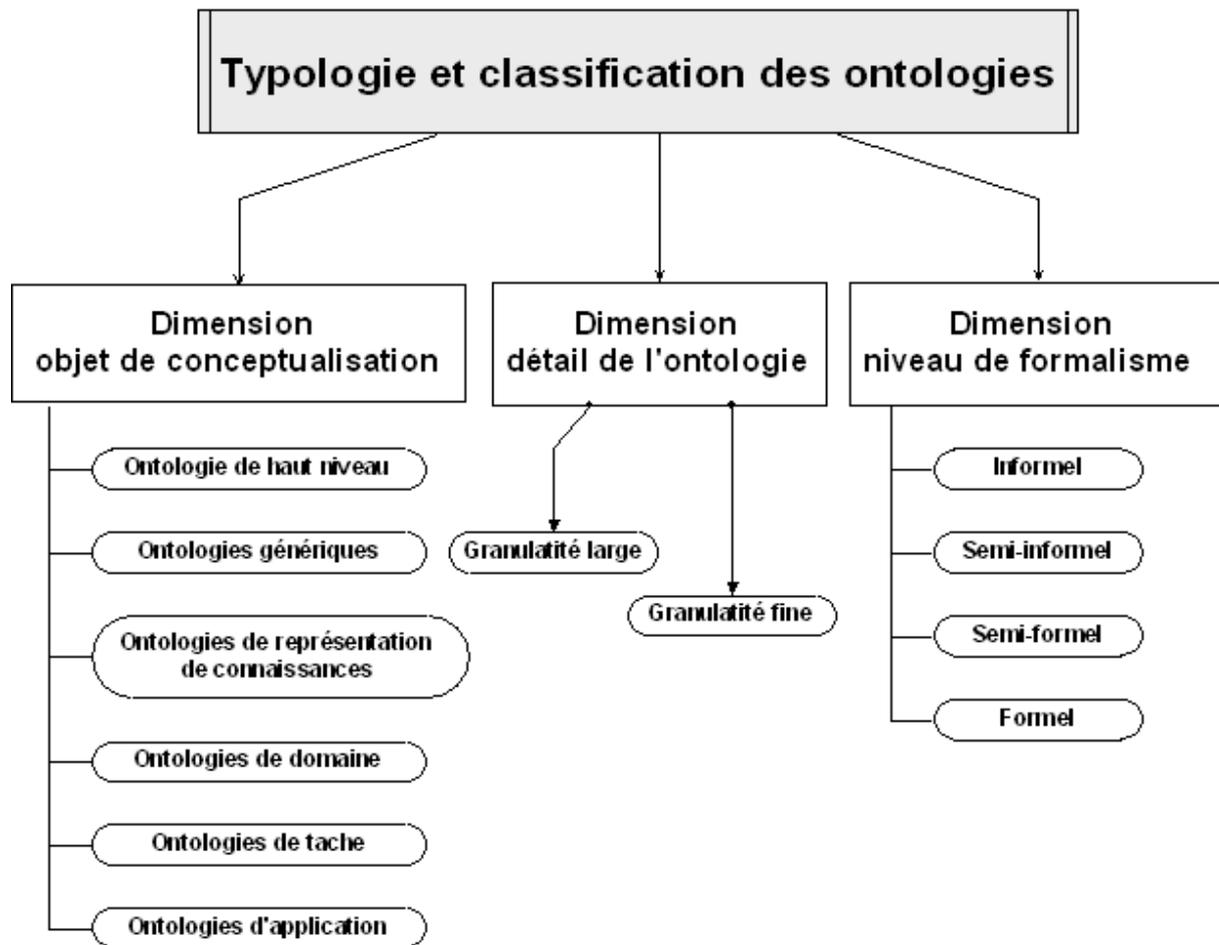


Figure 10 : Typologie et classification des ontologies

L'ontologie est considérée comme un pilier du Web sémantique. Des formalismes, des langages, des raisonneurs sont créés pour son développement dans le cadre du Web sémantique.

3.1.7. Langages de définition et de manipulation d'ontologies:

Il existe de nombreux langages informatiques, plus ou moins récents, spécialisés dans la création et la manipulation d'ontologies.

En voici quelques exemples :

- Open Knowledge Base Connectivity : OKBC 2.0, publié en 1997, est une API permettant d'accéder à des bases de connaissance (Knowledge Representation System).
- Knowledge Interchange Format est un langage destiné à faciliter des échanges de savoirs entre des systèmes informatiques disparates. Son objectif n'est pas de permettre une interaction avec l'être humain (bien que cela reste possible), mais plutôt une coopération entre systèmes hétérogènes.
- LOOM est un langage de représentation des connaissances dont le but avoué est de « permettre la construction d'applications intelligentes ».
- DARPA Agent Markup Language : fondé sur XML et RDF, DAML-ONT est apparu en Octobre 2000 suite à un effort du DARPA (Defense Advanced Research Projects Agency), afin d'autoriser l'expression de classes RDF plus poussées que ce que permettait à l'époque RDFS. Le programme DAML se poursuit encore à l'heure actuelle.

En réaction à l'apparition de ces nombreux langages poursuivant pour la plupart des buts communs, le World Wide Web Consortium (W3C) a mis sur pieds, en Novembre 2001, le groupe de travail « Web Ont », chargé d'étudier la création d'un langage standard de manipulation d'ontologies web. Le premier Working Draft « OWL Web Ontology Language 1.0 Abstract Syntax » paraît en Juillet 2002 et, au final, OWL devient une Recommandation du W3C le 10 Février 2004.

3.1.8. Techniques d'Intégration d'ontologie :

L'explosion du nombre de sources d'informations accessibles via le Web multiplie les besoins de techniques permettant l'intégration de ces sources. Afin que, les systèmes inter opèrent, l'enjeu est de développer des techniques facilitant l'interopérabilité sémantique. Trois techniques ont été proposées dans la littérature comme nous allons le voir [Euzenat 2005] :

3.1.8.1 Le mapping d'ontologies

La définition la plus pertinente est probablement celle de Noy pour qui le mapping d'ontologies est :

« Un processus qui spécifie une convergence sémantique entre différentes ontologies afin d'en extraire les correspondances entre certaines entités »[Noy 2007].

Les mappings sont des relations entre les éléments de deux ontologies, indiquant une similarité relative selon une mesure donnée. Le mapping d'ontologies permet de chercher une méthodologie pour un échange efficace et correcte entre les ontologies. La recherche en mapping d'ontologies est organisée autour de trois axes : la découverte qui constitue l'étape fondamentale de ce processus, la représentation et le raisonnement avec le mapping, ensuite, l'exploitation et l'exécution des mappings.

3.1.8.2 La fusion d'ontologies:

[Hyoil 2006] donne la définition suivante « **Ontology merging is the process of generating a single, coherent ontology from two or more existing and different ontologies related to the same subject** ». La fusion d'ontologies représente la création d'une nouvelle ontologie à partir de deux ou plusieurs ontologies existantes de même domaine ou des domaines proches, la nouvelle ontologie est supposée contenir les connaissances des ontologies initiales. L'ontologie résultante unifie et remplace les ontologies d'origine.

3.1.8.3 L'alignement des ontologies :

La tâche d'alignement d'ontologies représente un thème de recherche particulièrement très important dans l'intégration des systèmes, puisqu'il accepte la prise en compte associée de ressources décrites par des ontologies différentes. Ce thème de recherche est une tâche cruciale dans plusieurs domaines d'application, ce qui a donné lieu à de très nombreux travaux. Il met en évidence les relations sémantiques entre les entités de deux ontologies.

L'alignement des ontologies est considéré comme une solution au problème d'hétérogénéité des ontologies.

Il permet d'établir des liens sémantiques entre les concepts et des relations entre les ontologies.

L'alignement d'ontologies est un processus de découverte des correspondances entre deux ontologies source. Il est, généralement, décrit comme une application de l'opérateur MATCH [Bernstein 2001], dont l'entrée est constituée d'un ensemble d'ontologies et la sortie, formée des correspondances entre ces ontologies.

L'objectif du processus d'alignement est d'identifier le plus automatiquement possible, les relations entre les entités de différentes ontologies. Ces relations sont découvertes à travers les mesures de similarité entre ces entités.

En pratique, il s'agit d'identifier des concepts (ou des relations) de la première ontologie avec des concepts (ou des relations) de la seconde. De plus, les formalismes de représentation d'ontologie utilisés doivent être au moins compatibles, ainsi que, les paradigmes conceptuels.

3.1.9. Outils et environnements pour la construction d'ontologie :

Plusieurs environnements d'ingénierie ontologique ont été développés afin de systématiser l'ingénierie des ontologies. En voici les plus importants :

[Jean Charlet, Bruno Bachimont, Raphaël Troncy.2003]

1. PROTÉGÉ-2000

PROTÉGÉ-2000 [Noy *et al.*, 2000] est un environnement graphique de développement d'ontologies développé par le SMI de Stanford. Dans le modèle

des connaissances de PROTÉGÉ, les ontologies consistent en une hiérarchie de *classes* qui ont des attributs (*slots*), qui peuvent eux-mêmes avoir certaines propriétés (*facets*).

L'édition des listes de ces trois types d'objets se fait par l'intermédiaire de l'interface graphique, sans avoir besoin d'exprimer ce que l'on a à spécifier dans un langage formel : il suffit juste de remplir les différents *formulaire*s correspondant à ce que l'on veut spécifier. Ce modèle autorise d'ailleurs une liberté de conception assez importante puisque le contenu des formulaires à remplir peut être modifié suivant les besoins *via* un système de *méta*classes, qui constituent des sortes de « patrons » de connaissance. L'interface, très bien conçue, et l'architecture logicielle permettant l'insertion de *plugins* pouvant apporter de nouvelles fonctionnalités (par exemple, la possibilité d'importer et d'exporter les ontologies construites dans divers langages opérationnels de représentation ou encore la spécification d'axiomes) ont participé au succès de PROTÉGÉ-2000 qui regroupe une communauté d'utilisateurs assez importante et constitue une référence pour beaucoup d'autres outils.

2.OILEd :

OILEd [Bechhofer *et al.*, 2001], développé sous la responsabilité de l'université de Manchester, a été conçu pour éditer des ontologies dans le langage de représentation OIL (Ontology Interchange Language), un des précurseurs du langage OWL (*Ontology Web Language*) qui est une recommandation W3C. Officiellement, il n'a pas d'autre ambition que de construire des exemples montrant les vertus du langage pour lequel il a été créé. A ce titre, OILEd est souvent considéré comme une simple interface de la logique de description SHIQ. Néanmoins, il offre la plus grande partie de ce que l'on peut attendre d'un éditeur d'ontologies. On peut créer des hiérarchies de classes et spécialiser les rôles, et utiliser avec l'interface les types d'axiomes les plus courants. Cet éditeur offre également les services d'un raisonneur, FaCT, qui permet de tester la satisfaisabilité des définitions de classes et de découvrir des subsomptions restées implicites dans l'ontologie.

3.OntoEdit :

Contrairement aux deux outils précédents, ONTOEDIT [Sure *et al.*, 2002] n'est pas disponible gratuitement dans sa version complète³⁰. Il présente les fonctionnalités essentielles communes aux autres éditeurs (hiérarchie de concepts, expression d'axiomes, export de l'ontologie dans des langages divers) et a le mérite de s'appuyer sur une réflexion méthodologique significative. La modélisation des axiomes a fait l'attention de soins particuliers pour pouvoir être effectuée – en tout cas pour les types les plus répandus – indépendamment d'un formalisme privilégié et cela pour faciliter la traduction d'un langage de représentation à un autre. Il propose également une gestion originale des *questionnaires de compétences*. Des questions pour les réponses desquelles

l'ontologie doit fournir le matériel conceptuel, on peut extraire les termes appelés à intégrer l'ontologie. Un petit outil fait une comparaison lexicale des termes extraits des différentes questions pour en déduire automatiquement d'éventuelles subsomptions. Le procédé semble cependant loin d'être fiable puisqu'il repose sur l'hypothèse que le nom d'un concept se retrouve parfois dans le nom de ses spécialisations.

4. WebODE :

WebODE [Arpirez *et al.*, 2001], développé par le LAI de Madrid, est une plateforme de conception d'ontologies fonctionnant en ligne. D'un point de vue méthodologique, l'outil fait suite à ODE, un éditeur qui assurait fidèlement le support de la méthodologie maison METHONTOLOGY . Il illustre bien l'évolution des outils de construction d'ontologies, puisque les nombreuses tables de son prédécesseur ont été remplacées par une interface très travaillée, réalisant un pas supplémentaire vers une conception au niveau des connaissances. On peut cependant regretter que cette évolution se soit faite au détriment de l'application des contraintes méthodologiques : les *représentations intermédiaires* utilisées dans le processus de conception sont désormais moins mises en avant, à tel point que le guide de l'utilisateur ne les signale que pour « assurer la compatibilité conceptuelle avec ODE ». L'accent a plus été mis sur la possibilité d'un travail collaboratif ou sur la mise à disposition d'outils complémentaires, comme un moteur d'inférences.

5. DOE :

Le dernier outil présenté ici est DOE pour *Differential Ontology Editor* [Bachimont *et al.*, 2002; Troncy & Isaac, 2002]. Cet outil n'a pas pour ambition de concurrencer les grands environnements existants, mais plutôt de fournir un début d'implémentation à la méthodologie de structuration différentielle proposée par B. Bachimont . A l'instar des autres éditeurs, il offre une représentation graphique des arbres de concepts et des relations de l'ontologie et permet d'interagir avec les hiérarchies. L'outil assiste également la saisie des principes différentiels issus de la méthodologie en automatisant partiellement cette tâche. Le modèle de représentation de l'ontologie est finalement proche de celui du langage RDFS, à ceci près qu'il autorise la modélisation de relations n-aires. Au niveau formel, l'éditeur est capable de faire quelques inférences en vérifiant la consistance de l'ontologie.

6. Des outils à parfaire :

Tous ces outils sont capables de produire des ontologies dans les langages DAML+OIL et RDFS. Mais comme le montrent les expérimentations effectuées dans le cadre des *Workshops* EON [Angele & Sure, 2002], les ontologies générées sont loin d'être interopérables malgré un langage commun. Le support de ces outils vis-à-vis du nouveau langage OWL est, lui, très hétérogène :

PROTÉGÉ, OILEd et DOE produisent des ontologies OWL valides mais seul DOE est capable d'en importer. En plus ils n'ont pas toutes les fonctionnalités, de plus en plus nombreuses au fur et à mesure que les recherches se précisent, que réclame le développement distribué du Web sémantique.

3.1.10. Systèmes de raisonnement sur les ontologies :

Les systèmes de raisonnement permettent la classification des nouveaux concepts intégrés dans l'ontologie. Ainsi de vérifier et corriger la classification dans une ontologie.

Il existe plusieurs systèmes parmi eux nous citons :

RACER (Renamed Abox and Concept Expression Reasoner) [Haarslev 2001]: est un système implémentant une logique de description. Prise en compte des représentations au format *DAML+OIL*. RACER permet le test de satisfiabilité d'un concept (vérifier qu'un concept C admet des instances) le test de subsumption de concepts (vérifier qu'un concept C est subsumé par un concept D), et le test d'instanciation (vérifier qu'un individu a est instance d'un concept C, si seulement si a \hat{I} C).

JENA : est une bibliothèque de classes Java qui facilite le développement d'applications pour le web sémantique. Elle permet de : manipulation de déclarations RDF, lecture et écriture RDF/XML, Notation, Stockage en mémoire ou sur disque de connaissances RDF, Langage d'interrogation d'une base RDF, Gestion d'ontologies : RDFS.

3.1.11. La place des Ontologies dans le Web sémantique :

Sous l'expression Web sémantique, se regroupe un ensemble de programmes de recherche et de travaux très diversifiés. Leur objectif commun est de permettre aux machines de mieux exploiter automatiquement les contenus des ressources Web afin de décharger leurs utilisateurs d'une partie de leurs tâches et de leur proposer une aide afin d'atteindre un niveau de services qualitativement nouveau. Ces ressources vont être ainsi munies de descriptions sémantiques, simples méta-données documentaires, schémas d'annotations plus riches jusqu'à des cartes complexes. Dans ce contexte du Web sémantique, les ontologies doivent trouver une place centrale puisqu'elles vont fournir le vocabulaire et les structures sémantiques.

Les recherches sur les ontologies sont essentielles pour la réalisation du Web sémantique. En effet une fois construite et acceptée par une communauté particulière, une ontologie doit traduire un certain consensus explicite et un certain niveau de partage, qui sont essentiels pour permettre l'exploitation des ressources Web par différentes applications ou agent logiciels.

3.1.12. Limites de l'approche ontologique pour le web sémantique:

L'interopérabilité sémantique entre sources d'information hétérogènes est une problématique importante, du fait du nombre croissant de sources d'information disponibles sur le Web. L'utilisation des ontologies est une voie très prometteuse pour permettre l'interopérabilité, seulement, les ontologies elles même peuvent être hétérogènes. L'alignement des ontologies est le noyau de cette interopérabilité, et qui revêt toute son importance dans des applications nécessitant la prise en compte d'une interopérabilité sémantique. Cependant, la génération automatique des correspondances entre deux ontologies est d'une extrême difficulté, qui est dû aux divergences entre les différentes communautés de développement des ontologies. Les questions autour principalement de la comparaison, de la fusion des ontologies vont alors devenir cruciales.

En effet, La comparaison des ontologies pour partage ou fusion n'est possible que si les langages de représentation sont identiques. Leur puissance d'expression permet alors des comparaisons logiques, sur les propriétés, etc. très fructueuses mais le point crucial reste le sens réel des concepts : 2 concepts A et B, respectivement dans 2 ontologies O1 et O2, qui semblent semblables correspondent-ils réellement au même concept ? Le nom du concept, son label dans l'ontologie ne suffit pas, *surtout pas*, à répondre : en effet, ce n'est pas le label du concept qui fait sa signification dans une ontologie, c'est sa place dans l'arborescence. Si ce label nous aide à deviner la signification du concept, il n'est porteur de rien vis-à-vis de la représentation et peut même laisser croire à des identités qui n'en sont pas.

3.1.13. Conclusion et perspectives:

On abordera dans cette conclusion quelque perspective future sur les ontologies pour le web sémantique.

****comparaison et fusion des ontologies :**

Les travaux de recherche sur la fusion ou la gestion des versions ont déjà une dizaine d'année et leur poursuite est indispensable pour que les ontologies servent le Web sémantique : que ce soit dans la dimension syntaxique, où les travaux du Web sémantique sont bien avancés même s'ils soulèvent de nombreuses questions de représentation, ou dans la dimension logique où l'utilisation d'une ontologie doit pouvoir être caractérisée et assurée durant l'évolution de celle-ci. Plus important encore, la dimension sémantique des ontologies doit être prise en compte dans les comparaisons/fusions, c'est-à-dire que les ontologies doivent pouvoir être comparées pour la signification des concepts et pas seulement, même si c'est nécessaire, pour leur caractéristique

logique. Cette prise en compte est ainsi cruciale dans de nombreux points du Web sémantique.

****comment rendre les technologies du WS plus simple à utiliser :**

[Jean Charlet, Bruno Bachimont, Raphaël Troncy.2003]

Un des aspects importants, en vue de l'adoption du Web sémantique, est la capacité d'aider des utilisateurs non spécialistes à créer et à exploiter des ressources exploitables dans le cadre des infrastructures et des outils proposés par le Web sémantique. En d'autres termes, comment cacher la complexité des technologies du Web sémantique ?

Avant de répondre à cette dernière question, il est nécessaire de prendre conscience que le niveau de conceptualisation des ontologies n'est pas toujours accessible aux utilisateurs, même spécialistes du domaine. Ainsi, les ontologies en médecine sont inaccessibles à d'autres que leurs créateurs : d'abord parce qu'elles représentent un niveau de conceptualisation utile pour la tâche qui est demandé au SBC mais inutile à l'expression des concepts médicaux dans la pratique médicale, ensuite parce qu'en médecine comme dans d'autres domaines, l'expression des connaissances se fait en langue naturelle et que cette expression est normée par des thésaurus développés et utilisés depuis de très nombreuses années. Ainsi, l'accès aux ontologies par des utilisateurs, mêmes professionnels, nécessite de gérer le lien entre les concepts des ontologies et les termes du langage naturel, que ce soit pour une simple compréhension ou pour l'indexation et la construction de requêtes destinées à des tâches de recherche d'information.

Les solutions mises en œuvre à ce jour passent par deux types de propositions :

– des méthodologies séparant explicitement les termes et les concepts d'un domaine et c'est ce qui est expérimenté en médecine dans des *serveurs de terminologie* [Rector.A .L, 1998] ou dans d'autres domaines dans ce que les chercheurs appellent des *thésaurus sémantiques* [Roussey *et al.*, 2002] ;

– des méthodologies s'écartant de la formalisation des ontologies et recherchant des proximités conceptuelles dans les termes d'un domaine permettant d'en appréhender intuitivement la complexité [Bourigault & Lame, 2002];

Cette dernière approche interroge directement les chercheurs du Web sémantique : saurons-nous mettre en place des ressources formelles, des ontologies, pour le Web sémantique ou devons-nous en passer par des ressources moins formelles mais appréhendables par les utilisateurs ? Ces deux possibilités n'étant d'ailleurs pas exclusives.

3.2. Métadonnées et annotations dans le Web sémantique:

3.2.1. Introduction :

Tout le potentiel des technologies WS (RDF...) repose sur la mise en relation sémantique des ressources, à l'aide de propriétés que l'on leur attribue dans des formalismes standards. Ces relations sont aussi bien des relations de typage par classes ou des relations spécifiques aux différents domaines. Elles offrent des capacités d'organisation, de navigation, d'agrégation d'information, de requêtes et d'inférences. On retrouve les notions bien connues de métadonnées et d'annotations qui sont d'ores et déjà largement utilisées dans le cas des documents. Ces deux appellations sont couramment utilisées pour caractériser des informations ajoutées. Dans le cas du WS, elles sont utilisées pour des ressources en général en s'appuyant sur le fait que toute ressource est munie d'une URI.

Un des grands principes du Web sémantique est qu'il est nécessaire d'associer aux ressources du Web des informations exploitables par des agents logiciels afin de favoriser l'exploitation de ces ressources. Ces informations sont appelées métadonnées/annotations, et c'est cette notion que nous allons voir dans cette section.

3.2.2. Généralités :

3.2.2.1. Qu'est-ce qu'une métadonnée ?

Le terme "meta" vient du grec et dénote quelque chose de nature plus élevée ou plus fondamentale. Les métadonnées sont littéralement "des données relatives à d'autres données" (*data about data* : données sur des données). Toutefois, l'importance des métadonnées aujourd'hui mérite quelques précisions dans leur définition.

Nous citons ici la définition donnée par le National Information Standards Organisation (NISO), dans un article paru en 2004, intitulé "Understanding Metadata" :

« Une métadonnée (du Grec, "méta", ce qui dépasse, englobe) est une donnée à propos d'une autre donnée. En sciences de l'information, les métadonnées sont des ensembles de données structurées décrivant des ressources physiques ou numériques, ou, sur un plan plus fonctionnel, "de l'information structurée qui décrit, explique, localise la ressource et en facilite la recherche, l'usage et la gestion" ». [Morel-Pair 2001]

3.2.2.2. Qu'est-ce qu'une annotation?

Le Petit Robert définit le terme **annotation** comme une « note critique ou explicative qui accompagne un texte – une note de lecture qu'on inscrit sur un livre ».

Ainsi, le terme **annotation** réfère à une note, une critique, une explication ou encore à un commentaire. Or, nous rédigeons une note sur un sujet ou bien nous critiquons, expliquons, commentons un sujet. Une annotation seule ne fait pas sens, elle est toujours associée à l'objet qui a été annoté. C'est pourquoi les annotations sont considérées comme des **métadonnées**. Comme le souligne Handschuh [HANDSCHUH S. 2005], si une métadonnée est une donnée sur une donnée, une annotation constitue un cas particulier d'une métadonnée puisqu'elle représente une nouvelle donnée attachée à une ressource documentaire.

Il est important ici de préciser la notion de **ressource documentaire** : elle peut correspondre à l'ensemble d'un document ou bien seulement à un fragment de celui-ci et contenir du texte, de l'image, du son, de la vidéo ou une combinaison de ces contenus.

On le voit, les termes de métadonnée ou d'annotation prennent bien en compte cette notion d'ajout d'information à une ressource, et on pourra a priori les utiliser indifféremment pour décrire ces informations que le Web sémantique doit ajouter au Web pour le rendre plus utilisable par des machines.

Pour autant, si ces deux termes existent, c'est qu'ils n'ont pas le même sens.

La communauté anglophone du Web sémantique, circonspecte sur la différenciation, considère par exemple dans le schéma qui suit, que les annotations de pages Web *deviennent* des métadonnées dès qu'elles sont stockées dans une base sur un serveur. On différencie alors l'information en tant qu'elle est attachée à (et présentée avec) une ressource et l'information manipulable et exploitable de façon plus indépendante de celle-ci.

D'un point de vue plus lié à la pratique de l'annotation / métadonnées (du point de vue du processus de mise en place), et s'il est nécessaire de distinguer, nous proposons ce distinguo : [Yannick Prié1, Serge Garlatti. 2003]

- une métadonnée sera plutôt attachée à une ressource identifiée en tant que telle sur le Web – aura plutôt une pertinence a priori et sera plutôt saisie suivant un schéma. Par exemple, la description normalisée d'un service Web, l'auteur d'un document, qui permettront de mettre en place des inférences.

- une annotation sera plus *située* au sein de cette ressource et *écrite* au cours d'un processus d'annotation / lecture. Par exemple, un commentaire libre associé à un fragment d'une page Web – quelques mots, un paragraphe – déterminé au besoin.

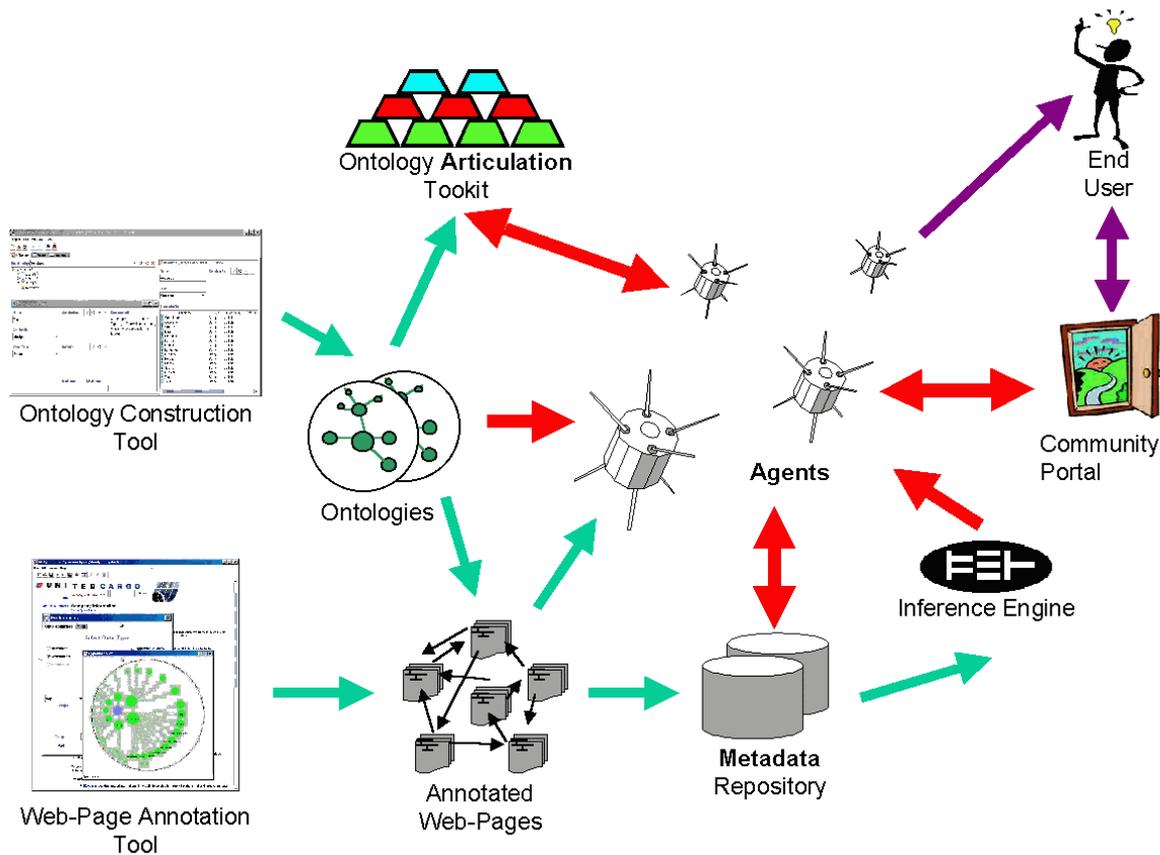


Figure 11 : vision de l'utilisation des métadonnées sur le Web sémantique.
(Proposée sur le site : <http://www.semanticweb.org/>)

La Figure ci-dessus donne une vision de l'utilisation des métadonnées sur le Web sémantique. Des pages Web sont annotées à partir de connaissances disponibles dans une ou plusieurs ontologies (qui ont pour objectif de normaliser la sémantique des annotations), et ces annotations, regroupées en entrepôts de métadonnées deviennent utiles pour des agents de recherche d'information, faisant ou non appel à des moteurs d'inférence permettant de déduire de nouvelles connaissances formelles des annotations.

3.2.2.3. Intérêts des métadonnées/annotation :

Le terme « métadonnée » est utilisé depuis longtemps dans certains domaines d'activité, comme la description des documents géographiques, la gestion des ressources images et multimédias ou les bases de données. Ce concept est aussi au cœur de certains métiers comme ceux des bibliothèques et de l'archivistique. Mes surtout celui du web sémantique auquel elle constitue un de ces composant principale.

Les fonctions des métadonnées peuvent en effet être déclinées en six groupes [Morel-Pair 2007a] :

1. Améliorer la recherche d'information et la « découverte » des ressources : avec des métadonnées descriptives du contenu : titre, résumé, mots-clés, classement ... Cet objectif, premier dans l'histoire des métadonnées, peut inclure d'autres éléments comme la qualité du document.

2. Gérer les ressources: grâce à deux grands sous-ensembles de métadonnées :

- D'une part, des métadonnées administratives, portant sur la propriété intellectuelle, la responsabilité, les droits d'usage et les sources utilisées.
- D'autre part, des métadonnées instanciées, ou techniques et de structure, regroupant les caractéristiques physiques ou informatiques, telle que le format, les propriétés techniques détaillées de documents particuliers comme les images, dates significatives dans le cycle de vie, structure ou place dans une hiérarchie, logiciels de consultation...

3. Gérer les « archives » : au sens du Record management ; les archives sont ici l'ensemble des documents utiles à court et moyen terme. Dans ce processus, il s'agit « d'identifier, authentifier, localiser, et contextualiser les données, ainsi que les personnes, les processus et les systèmes qui les créent, les gèrent ou les utilisent et les politiques qui les régissent » pour garantir la qualité, la fiabilité, l'accessibilité et la pérennité des ressources [Day 2005].

4. Faciliter le partage de données et leur réutilisation: cette fonction, importante pour limiter les coûts, passe par l'amélioration de l'interopérabilité au travers de standards, ainsi que par la présence d'informations contextuelles pour guider l'interprétation des données. Dans le cadre du Web sémantique, l'interopérabilité entre ensembles de données concerne les machines.

5. Participer à la pérennité des ressources numériques : qui garantit que « l'essence du contenu est accessible pour toujours ». On utilise pour cela des métadonnées de préservation qui décrivent « entre autres les actions réalisées en vue d'assurer la pérennité et l'accès pérenne, telles les migrations ou contrôles d'intégrité des fichiers » [Day 2005].

6. Décrire les utilisateurs pour gérer les accès : leur permettre des personnalisations de consultation, analyser les comportements d'usage...

3.2.3 Processus d'annotation sémantique :

Derrière le processus général d'annotation sémantique de documents par des ontologies se cache plusieurs phases qui font référence à des annotations de natures un peu différentes. Ces phases sont au nombre de trois :

Repérer : processus manuel [Handschuh , 2001], [Heflin, 2002] ou automatique qui consiste à placer dans le document des références aux concepts de l'ontologie qu'il contient. Ces éléments sont considérés comme des métadonnées.

Instancier : processus manuel ou automatique permettant de valuer les attributs des concepts à l'aide des informations présentes dans le document (là encore, ce sont des métadonnées).

Enrichir : processus manuel visant à ajouter des informations par l'intermédiaire des attributs de concepts qui n'ont pas pu être valus à la phase précédente.

Les deux premières étapes sont des étapes d'insertion de métadonnées : il n'y a pas ajout d'information mais plutôt *localisation et caractérisation de l'information* déjà présente de manière intrinsèque. Par contre, la dernière est plutôt une étape d'annotation plus classique, car il y a *ajout d'information* : le document est enrichi d'information qui n'est pas explicitement présente dans le document. Cette annotation est directement "formalisée" par des métadonnées.

3.2.4. Familles d'annotations et de métadonnées :

Nous nous contenterons ici de donner un certain repérage des familles d'annotations et de métadonnées. [Philippe LAUBLET]

La première famille de métadonnées les plus facilement caractérisables peuvent être appelées documentaires ; elles décrivent les conditions de production des ressources : Qui ? Quand ? Où ? Comment ? Elles sont bien sûr largement utilisées, indépendamment du WS, pour des tâches de documentation. Elles ne sont qu'un premier niveau dans la description des ressources. Dans le cadre du WS, l'ensemble de métadonnées documentaires le plus largement repris et qui a vocation de standard est le Dublin Core.

Ses principaux descripteurs sont : *titre, créateur, sujet, description, éditeur, contributeur, date, type, format, identifiant, source, langue, relation, couverture, droits*

. Chacun des descripteurs est défini par un ensemble d'attributs provenant de la description standard ISO/IEC 11179. Dans le cas du WS, les métadonnées, correspondant au Dublin Core, sont alors exprimées en RDF et sont souvent intégrées à d'autres ensembles de métadonnées.

Le CanCore est un exemple de jeu de métadonnées qui étend la notion de métadonnées documentaires. Il s'agit d'un standard canadien pour la recherche et la localisation des ressources pédagogiques en ligne (ou pour des parties de ces ressources) appelées « objets pédagogiques ». Elles incluent :

– des métadonnées générales, attributs de l'objet pédagogique (titre, langue, sujet, description...)

- des métadonnées sur le cycle de vie de l'objet, sur les circonstances de son développement (nom, date de publication, information de publication et de version...);
- des méta-métadonnées sur le dossier des métadonnées (contributeurs, langue, date, validation);
- des métadonnées techniques et éducationnelles (format technique, taille, emplacement...), pédagogiques (type de ressources, contexte, niveau d'âge...), sur les droits etc.

La deuxième grande famille de métadonnées ou annotations porte sur le contenu des ressources. Elles peuvent indiquer quel est le sujet, à quoi cela fait référence et, de manière plus générale, elles relient des ressources à d'autres ressources. Dans les approches que nous voulons souligner, ce sont les concepts et les relations d'une ontologie du domaine qui sont à la base de ces annotations. Si l'on prend simplement l'exemple d'un système de recherche d'information, celui-ci pourra mieux satisfaire les besoins de ses utilisateurs en utilisant les différentes relations de l'ontologie.

3.2.5. Outils d'annotation sémantique :

Un outil d'annotation sémantique est un outil logiciel qui permet d'insérer et de gérer des annotations sémantiques liées à au moins une ressource donnée.

Dans le cadre du Web Sémantique, les outils d'annotation sémantique utilisent une ontologie, ou tout au moins un modèle formel, qui formalise et structure les annotations produites en fonction des concepts et des contraintes définis dans cette ontologie. Un outil d'annotation sémantique peut aussi être utilisé pour peupler une ontologie, i.e. pour instancier la base de connaissance contenant les instances de l'ontologie de référence.

Nous présentons ci-dessous quelques exemples d'outils. Nous n'avons considéré que les outils d'annotations de ressources textuelles, mais il existe également des outils ou projets s'intéressant également à l'annotation sémantique de ressources multimédia.

1. Annotea :

C'est un système de génération (manuelle) d'annotations RDF pour les pages web [Kahan et al, 2001] développé au sein du W3C. L'idée principale est de proposer à chaque utilisateur un outil lui permettant : (i) d'annoter un document en le consultant, (ii) de consulter toutes les annotations associées à un document et (iii) de typer les annotations en leur associant des métadonnées (par exemple dire que cette annotation est un commentaire ou un erratum pour le document) en se basant sur un modèle prédéfini en RDFS. Un langage permettant de décrire des ressources et les relations entre ces ressources.

2. MnM :

MnM fournit un environnement pour la génération semi-automatique d'annotations sémantiques associées aux documents Web [Vargaz-vera et al,

2002 a] [Vargaz-vera et al, 2002 b] Il est basé essentiellement sur des techniques d'apprentissage et des méthodes de TALN (Traitement Automatique de la Langue Naturelle). Dans un premier temps, un corpus d'apprentissage est fourni aux utilisateurs afin de l'annoter manuellement en se basant sur une ontologie du domaine. Ce corpus est ensuite passé à l'outil Amilcare [Ciravegna et al, 2004a] qui génère un ensemble de règles d'extraction qui seront ensuite appliquées sur les autres documents pour générer automatiquement les annotations.

3. KIM :

KIM [Popov et al, 2003] fournit une plate-forme de génération automatique d'annotations sémantiques et de recherche documentaire basée sur ces annotations. L'approche de KIM est basée sur l'extraction des entités nommées présentes dans le texte à annoter afin d'instancier les concepts d'une ontologie de haut niveau (KIMO) représentée en RDFS. Ces instances sont ensuite utilisées pour annoter les documents et pour enrichir la base de connaissances de KIM. Ces annotations permettent ainsi de faire des recherches documentaires plus précises en utilisant les restrictions sémantiques offertes par l'ontologie. Le processus d'extraction d'informations de KIM est basé sur l'outil GATE.

4. S-CREAM :

C'est une plate-forme pour la création semi-automatique d'annotations sémantiques basées sur une ontologie [Handschuh et al, 2002]. Ce sont des travaux inspirés d'Annotea ont conduit à des systèmes d'annotation par rapport à une ontologie en DAML+OIL. Cette plate-forme fournit deux approches, toutes les deux implémentées dans l'outil OntoMat: La première est basée sur une phase d'apprentissage manuelle (similaire à celle de MnM), et la deuxième est basée sur une méthode originale nommée PANKOW [Cimiano et al, 2004] qui est entièrement automatique.

Beaucoup d'autres systèmes ont été proposés pour la génération (semi)automatique d'annotations sémantiques. Ils proposent, le plus souvent, des annotations en RDF (qui est un standard pour la représentation des métadonnées).

3.2.6. L'annotation sémantique/métadonnée et le Web Sémantique :

Dans le cadre du Web Sémantique, l'objectif est de décrire le contenu des ressources en les annotant avec des informations non ambiguës afin de favoriser l'exploitation de ces ressources par des agents logiciels [PRIE Y. & GARLATTI S 2004]. Or, les données actuelles du Web sont encore trop souvent écrites en langage naturel, car destinées aux humains. Le langage naturel étant par essence trop ambigu, des alternatives formelles et sémantiquement explicites doivent être mises en place pour lever les ambiguïtés du langage naturel, aussi bien dans le contenu des ressources que dans leurs annotations. La tâche d'annotation pour le Web Sémantique consiste donc à prendre en entrée une ressource documentaire et fournir en sortie le même contenu enrichi par des annotations

sémantiques basées sur des représentations de la connaissance plus ou moins formelles.

3.2.7. Travaux sur les métadonnées /annotations pour web sémantique :

Il nous apparaît que les travaux sur le Web sémantique, considérés du point de vue des annotations et des métadonnées peuvent se répartir comme suit: [Yannick Prié¹, Serge Garlatti. 2003] :

- travaux sur les langages de description d'ontologies et de métadonnées ;
- travaux sur la construction des ontologies pour associer des métadonnées, décrire des pages Web ;
- travaux sur l'utilisation de métadonnées pour la personnalisation de pages Web ;
- travaux sur des applications pilotes, mettant en jeu :
 - ✓ des schémas / ontologies pour annoter,
 - ✓ des outils pour annoter conformément à ces schémas,
 - ✓ des applications :
 - faisant usages de métadonnées en tant que telles (recherche d'information),
 - faisant également usage des annotations pour les présenter en même temps que les pages à l'utilisateur,
 - ✓ des architectures pour soutenir ces applications,
 - ✓ des « leçons apprises ».

3.2.8. Conclusion et perspective :

Il nous apparaît que les travaux liés aux annotations / méta-données visant à mettre en action les théories du web sémantique sont nombreux, et que force est de constater qu'ils commencent à fournir des résultats techniquement impressionnants en terme de prototypes et d'architectures, concrétisant énormément d'efforts au niveau international.

Cependant, même si les bénéfices potentiels apportés par ces nouvelles possibilités d'applications sont indéniables, il ne nous semble que des recherches impliquant parfois des équipes pluridisciplinaires (informatique/sciences humaines et sociales) doivent être menées sur quelques thèmes fondamentaux pour le WS.

Il ne fait pas de doute que les recherches actuelles vont être poursuivies, sans doute dans toutes les directions à la fois, étant donnée l'effervescence actuelle autour du Web sémantique. Cependant, une certaine attention de recherche doit être portée dans quelques directions, qui à notre sens fondamentales.

On mentionnera quelques points en transposant :

- Doit-on annoter avec une ontologie préexistante ou bien construire une ontologie adaptée au cours de l'annotation ?
- Que dire de la notion d'indexation du document par l'auteur lui-même ?

- Peut-on faire confiance à un utilisateur pour bien décrire les ressources (à comparer par exemple avec la fonction d'un documentaliste) ?
- Comment prendre en compte les problèmes de droit des annotations (diffusion des métadonnées, responsabilité, respect de la vie privée) ?
- Une métadonnée peut être publiée dans le cadre d'une tâche, ce qui suppose la maîtrise de l'utilisation qui peut en être faite par son auteur. Or, des usages non prévus peuvent surgir. Se pose aussi le problème de la « validité » des métadonnées par rapport à l'évolution des documents, mais aussi d'un point de vue temporel (cycle de vie, qualité et validation des métadonnées/annotations).
- Comment prendre en compte les problèmes de l'évolution de l'annotation sémantique. (analyse des effets des changements ontologiques sur l'annotation, la détection des annotations inconsistantes et surtout la résolution des annotations inconsistantes.)

3.3. Les agents:

Le pouvoir véritable du Web sémantique sera atteint quand les gens créeront de nombreux programmes qui collecteront les contenus du Web à partir de sources diverses, qui traiteront l'information et échangeront les résultats avec d'autres programmes. L'efficacité de ces agents logiciels croîtra de manière exponentielle au fur et à mesure que seront disponibles des contenus du Web lisibles par des machines et des services automatisés (comprenant d'autres agents). Le Web sémantique promet cette synergie : même les agents qui ne sont pas expressément fabriqués pour travailler ensemble peuvent transférer des données entre eux si les données sont accompagnées de sémantique.

Un aspect important du fonctionnement des agents sera l'échange de " preuves " écrites dans le langage unifié du Web sémantique (langage qui exprime les inférences logiques en utilisant des règles et des informations comme celles qui sont spécifiées par les ontologies). Par exemple, supposons que l'information pour contacter Ms Cook ait été localisée par un service en ligne, et à votre grande surprise il la situe à Johannesburg. Vous voulez naturellement vérifier cela, donc votre ordinateur demande au service de prouver sa réponse, ce qu'il fait rapidement en traduisant son raisonnement interne en langage unifié du Web sémantique. Un moteur d'inférence dans votre ordinateur vérifie que la Ms Cook en question est bien celle que vous cherchiez, et il peut vous montrer les pages Web pertinentes si vous avez encore des doutes. Bien qu'ils soient encore loin d'atteindre les profondeurs du potentiel du Web sémantique, quelques programmes sont déjà capables d'échanger des preuves de cette façon, en utilisant les premières versions courantes du langage unifié.

Un autre trait vital sera les signatures digitales, qui sont des blocs encryptés de données que les ordinateurs et les agents peuvent utiliser pour vérifier que l'information attachée a été fournie par une source fiable. Vous voulez être tout à fait sûr qu'un état envoyé à votre programme comptable qui dit que vous devez de l'argent à un détaillant en ligne n'est pas une contrefaçon produite par l'ordinateur de l'adolescent d'à côté. Les agents devraient se méfier des affirmations qu'ils lisent sur le Web sémantique jusqu'à ce qu'ils aient vérifié les sources d'information.

4. Architecture et langages du web sémantique:

4.1. Le World Wide Web Consortium (W3C):

Fondé en 1994, pour développer les protocoles nécessaires à l'évolution du Web, c'est un consortium international qui regroupe de nombreux professionnels de l'industrie, du service, de la recherche et de l'enseignement partageant les mêmes objectifs d'évolution et de stabilisation à long terme des technologies du Web.

4.2. L'architecture du Web Sémantique :

L'architecture du web sémantique s'appuie sur une pyramide de langages, proposée par Tim Barners-lee pour représenter des connaissances sur le web en satisfaisants les critères de standardisation, d'interopérabilité et de flexibilité.

Un langage de la couche haute doit être une extension du langage de la couche au-dessous.

Jusqu'à aujourd'hui, seulement les couches basses sont relativement stabilisées.

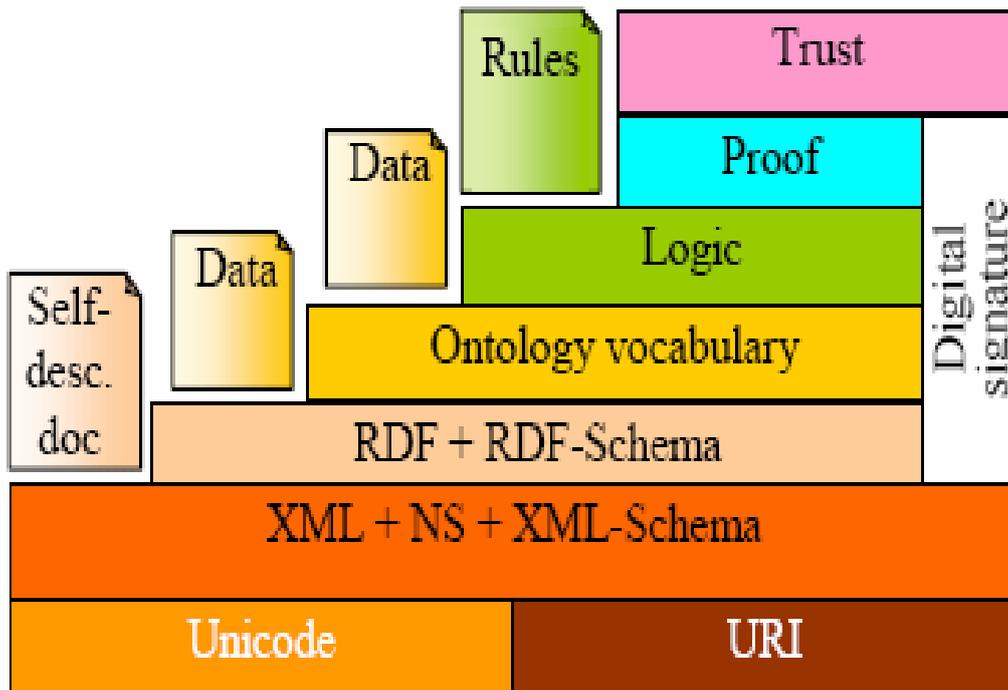


Figure 12 : les couches du Web Sémantique [Tim Barners-lee, 2001]

Deux types de bénéfices peuvent être attendus de cette organisation :

- ✓ Elle permet une approche graduelle dans les processus de standardisation et d'acceptation par les utilisateurs.
- ✓ Si elle est bien conçue, elle doit permettre de disposer du langage au bon niveau de complexité, celle-ci étant fonction de l'application à réaliser.

La liste suivante introduit la fonction principale de chaque couche dans l'architecture du web sémantique :

La couche URI : permet d'attribuer un identificateur unique à une ressource.

La couche XML : fournit une surface syntaxique pour les documents structurés mais ne fournit aucune contrainte sémantique sur le sens de ces documents.

La couche RDF : 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.

La couche ontologie : fondé sur une formalisation commune, spécifie la sémantique des métadonnées fournies dans le web sémantique.

La couche logique : s'appuie sur des règles d'inférence qui permettent de faire des raisonnements intelligents exécutés par des agents logiciels.

La couche preuve et confiance : supportent un mécanisme de communication inter-agents pour valider les résultats des raisonnements, cela pourra assurer la fiabilité des services automatiques du web sémantique.

4.3. Les Langages du Web sémantique :

La manipulation des ressources du Web par des machines requiert l'expression ou la description de ces ressources. Plusieurs langages sont donc définis à cet effet, ils doivent permettre de structurer les données pour une manipulation syntaxique (XML), d'exprimer données et métadonnées (RDF), les interroger (SPARQL) et de disposer d'un modèle abstrait de ce qui est décrit grâce à l'expression d'ontologie (RDFS, OWL).

Ces langages doivent accepter les caractères propres au Web à savoir sa distribution (il faut être capable de tirer parti d'information dont on ne dispose pas localement) et son ouverture (n'importe qui peut ajouter de l'information à tout instant).

4.3.1. Les précurseurs :

Nous citerons HTML-A, et SHOE.

1. HTML-A

Une extension d'HTML, appelée HTML-A est utilisée pour insérer des annotations sémantiques dans les pages Web. Cette approche ne spécifiait pas le langage d'implémentation de l'ontologie de référence. Par contre, comme décrit dans [Benjamins et al, 1999], les agents logiciels du Web devaient ensuite savoir interpréter cette extension pour l'exploiter correctement.

```
<html>
<head><Title>Le Clan coppola</Title>
<A ONTO="Personnalité:FFCoppola"/>
</head>
<body>
Francis Coppola naît le <A ONTO="Personnalité[dateNaissance=body]">7 avril 1939</A> à <A
ONTO="Personnalité[lieuNaissance=body]">Detroit</A>, dans le <A
ONTO="Personnalité[lieuNaissance=body]">Michigan</A>.
</body>
</html>
```

Figure 13: Exemple d'une annotation sémantique en HTML-A

2. SHOE : Simple HTML Ontology Extension

L'approche de SHOE [Heflin, 2001] crée une extension d'HTML ayant pour objectif l'insertion d'annotations sémantiques pour la description des ressources du Web.

Au lieu d'utiliser l'attribut **ONTO** dans l'élément **A**, SHOE proposait d'utiliser un ensemble d'éléments prédéfinis tels **INSTANCE**, **CATEGORY**, **RELATION**, ...etc.

Ces éléments étaient directement insérés dans le code HTML de la page Web annotée.

Les annotations devaient ensuite être interprétables par des agents Web. L'objectif de ce langage était de permettre aux agents de glaner de la

connaissance au sujet de pages Web afin d'améliorer les mécanismes de recherche d'information et de fouille de données.

Les ontologies SHOE décrivaient les relations entre les classes, appelées catégories en SHOE, et un ensemble de règles d'inférence simplifiées. Les concepts de l'ontologie de référence sont directement déclarés dans le corps du document et les annotations sont débarquées vis-à-vis du texte originel.

```

<html>
<head><Title>Le Clan coppola</Title></head>
<body>
Francis Coppola naît le 7 avril 1939 à Detroit, dans le Michigan.

<INSTANCE KEY="FFCoppola">
<USE-ONTOLOGY ID="People-Ontology" URL="http://www.elle.com/SHOE/people.html"
VERSION="1.0" PREFIX="people">
<CATEGORY NAME="people.FFCoppola">
<RELATION NAME="people.dateNaissance">
  <ARG POS=1 VALUE="FFCoppola">
  <ARG POS=2 VALUE="7 avril 1939">
</RELATION>
<RELATION NAME="people.lieuNaissance">
  <ARG POS=1 VALUE="FFCoppola">
  <ARG POS=2 VALUE="Detroit">
</RELATION>
</INSTANCE>

</body>
</html>

```

Figure 14 : Exemple d'une annotation sémantique en SHOE

4.3.2. Le langage XML :

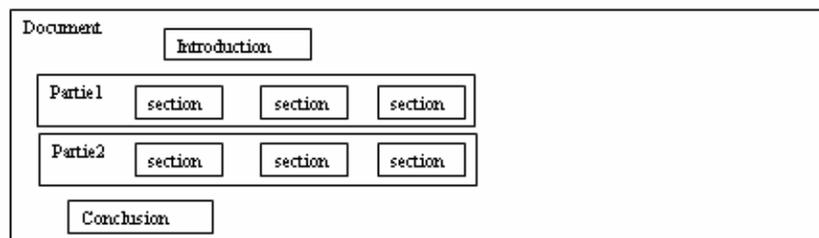
4.3.2.1 Introduction du langage :

Actuellement, Internet est un simple moyen d'accès à du texte et à des images ...etc. Il n'existe pas de normes pour la recherche intelligente, l'échange de données, la présentation adaptable. Internet doit être plus qu'un espace d'accès à des informations et une norme d'affichage. Il doit définir une norme de gestion de l'information, une manière commune de représenter les données afin que les logiciels puissent plus facilement rechercher, déplacer, afficher et manipuler des informations. Le HTML en est incapable car il s'agit d'un format qui décrit l'affichage d'une page Web; il ne représente pas les données.

4.3.2.2. Présentation du langage XML :

Recommandation du W3C depuis le 10 février 1998, le langage eXtensible Markup Language (XML) [Michard, 1999] est un langage de description et d'échange de documents structurés, issu de SGML (Standard Generalized Markup Language) et défini par le consortium Web. XML permet de décrire la structure arborescente 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 offre la possibilité de créer ses propres balises. XML ne pose

aucune contrainte sémantique sur la description des informations, il ne constitue donc pas un langage de modélisation d'ontologie à lui seul. En réalité les balises XML décrivent le contenu plutôt que la présentation (contrairement à HTML). Ainsi, **XML permet de séparer le contenu de la présentation**, ce qui permet par exemple d'afficher un même document sur des applications ou des périphériques différents sans pour autant nécessiter de créer autant de versions du document que l'on nécessite de représentations !



Correspond à :

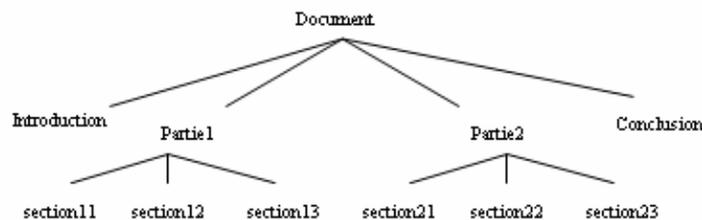


Figure 15 : Exemple de représentation arborescente d'un document XML

XML fournit un moyen de vérifier la syntaxe d'un document grâce aux DTD (Document Type Definition). Il s'agit d'un fichier décrivant la structure des documents y faisant référence grâce à un langage adapté. Un document suivant les règles de XML est appelé document bien formé. Un document XML possédant une DTD et étant conforme à celle-ci est appelé document valide. La norme XML n'impose pas l'utilisation d'une DTD pour un document XML, mais elle impose par contre le respect exact des règles de base de la norme XML.

Il est essentiel pour le receveur d'un document XML de pouvoir extraire les données du document. Cette opération est possible à l'aide d'un outil appelé analyseur (en anglais parser, parfois francisé en parseur) comme DOM (Document Object Model) et API.SAX (Simple API for XML).

XML a conquis depuis de multiples formats, les nouveaux comme les anciens. De xhtml au format de documents openoffice.org, nombreux sont les documents qui profitent aujourd'hui du cadre de XML.

4.3.2.3. Structure d'un fichier XML

- ✓ Une source de données est un document XML si elle est « bien formée », c'est à dire si elle correspond parfaitement à la spécification de XML.

- ✓ Un document XML est représenté physiquement sous la forme d'un fichier texte structuré en éléments, à l'aide de balises éventuellement imbriquées.
- ✓ En en-tête du document doit figurer un « prologue », une déclaration qui identifie le document comme un document XML. Ce prologue indique la version de XML employée, le codage de caractères, et si le document est associé à une DTD ou s'il est autonome.
- ✓ Il existe un élément particulier : l'élément « racine », encore appelé « élément document ».
- ✓ Cette racine doit contenir tous les autres éléments du document et ne peut apparaître qu'une fois dans un document XML.

Exemple de document XML simple

L'exemple suivant de document XML permet de décrire une personne :

```
<?xml version="1.0" encoding="UTF-8"?>
<personne>
  <nom>Kaci</nom>
  <prenom>Yacine</prenom>
  <naissance>
    <lieu>
      <ville>Tizi Ouzou</ville>
      <pays>Algerie</pays>
    </lieu>
    <date>
      <jour>14</jour>
      <mois>7</mois>
      <annee>1979</annee>
    </date>
  </naissance>
</personne>
```

Figure 16 : Exemple de fichier XML simple

4.3.2.4. Le web sémantique et l'XML :

XML reste un langage très important pour structurer le contenu du web malgré le nombre assez réduit des document XML actuellement disponibles.

Or, cette tendance va s'inverser, car d'après une étude de Gartner, l'XML a largement dépassé actuellement tout les autres formats de publication de données, et cela pour tous les contenus multi-supports .XML est utilisé sur le web car il permet la description de documents électronique par l'intermédiaire d'une DTD.il fait le lien entre les documents et les logiciels qui les utilisent. En effet, le but de l'XML est de faciliter la diffusion et l'échange d'informations sur internet.

Parmi les avantages que XML peut apporter au web en général et au SW en particulier :

- Son indépendance par rapport aux plates-formes.
- Souplesse et puissance.
- Son exploitation possible par un système informatique.
- La séparation de la représentation du contenu.
- Normalisation des données.
- C'est un langage de description facilement extensible en fonction des besoins des applications.

4.3.2.5. Avantages et insuffisances de XML:

1. Les avantages :

Le XML présente beaucoup d'avantages, on peut citer:

- **Recherche plus significative :**

Les données peuvent être balisées de façon très précise en XML. Ainsi un utilisateur peut spécifier qu'il cherche des livres de `Tim Barners-lee` plutôt que des livres qui parlent sur `Tim Barners-lee`. En effectuant la même recherche et si le contenu n'est pas décrit en XML on aura comme réponse les deux types de livres simultanément.

- **Vues multiples sur les données**

En XML on décrit les données et non pas la manière dont elles seront affichés, ainsi les mêmes données peuvent avoir différentes représentations selon les préférences des utilisateurs.

- **Traitement et manipulation de données en local**

Les données au format XML, une fois transmises au client, peuvent être analysées, modifiées et manipulées en local, le traitement étant réalisé par les applications clientes.

- **Interopérabilité**

Des données provenant de plusieurs sources peuvent être intégrées et manipulées par différentes applications.

En plus de ces avantages on peut citer encore :

- La lisibilité, une structure arborescente : permettant de modéliser la majorité des problèmes informatiques,
- universalité et portabilité, déployable (il peut être facilement distribué par n'importe quels protocoles à même de transporter du texte, comme http),
- intégrabilité à des applications pourvue d'un parser (c'est-à-dire un logiciel permettant d'analyser un code XML),
- extensibilité (un document XML doit pouvoir être utilisable dans tous les domaines d'applications).

2. Insuffisances de XML:

Dans le cadre du Web Sémantique où l'on cherche à donner du sens à l'information, on se rend rapidement compte de la limite de XML. En d'autres termes XML traite la structure syntaxique des documents mais ne permet de spécifier ni la sémantique d'une balise, ni le typage de son contenu, ni des relations normalisées entre les balises. C'est la raison pour laquelle un nouveau langage a été défini : RDF (Ressource Description Framework).

4.3.3. Le langage RDF:

4.3.3.1 Introduction au langage:

L'arrivée de XML, en 1998, a donné un cadre à la structuration des connaissances, rendant ainsi possible la création de nouveaux langages web destinés non plus à un rendu graphique à l'écran pour un utilisateur humain, mais à un réel partage et à une manipulation des savoirs.

C'est dans cet esprit qu'a été créé en 1999 RDF, un langage XML permettant de décrire des métadonnées et facilitant leur traitement.

4.3.3.2 Définition:

RDF [Lassila & Swick, 1999][Klyne & Carroll, 2003] est un langage formel qui permet d'affirmer des relations entre des « ressources ». Il sera utilisé pour annoter des documents écrits dans des langages non structurés, ou comme une interface pour des documents écrits dans des langages ayant une sémantique équivalente (des bases de données, par exemple).

Un document RDF est un ensemble de triplets de la forme < sujet, prédicat, objet >. Les éléments de ces triplets peuvent être des URIs (Universal Resource Identifiers) [Berners Lee et al. 1999], des littéraux ou des variables.

Cet ensemble de triplet peut être représenté de façon naturelle par un graphe plus précisément un multi-graphe orienté étiqueté où les éléments apparaissant comme sujet ou objet sont des sommets, et chaque triplet est représenté par un arc dont l'origine est son sujet et la destination est son objet.

Ce document sera codé en machine par un document RDF/XML, mais est souvent représenté sous une forme graphique.



Figure 17: Triplet RDF.

Quand les données de XML sont déclarées au format RDF, les applications peuvent comprendre une grande partie de la traduction des données. Il existe

cependant un certain nombre de situations où l'on est en droit d'être plus exigeant. Par exemple, certaines fois, nous avons besoin de connaître des informations sur les ressources identifiées, comme par exemple ce qu'elles représentent exactement : Si nous avons une propriété représentant un auteur, nous pouvons en effet exiger que la valeur d'une telle propriété fasse référence à une personne (et non une voiture ou une maison). C'est pour cela que l'on associe à RDF le standard RDF Schéma (RDFS), qui est un peu l'équivalent des DTDs (*Document Type Definition*) pour le XML.

RDFS (*Ressource Description Framework Schéma*) : est une extension de RDF. Permettant de décrire les ressources dans le contexte du Web sémantique, RDFS ajoute à RDF la possibilité de définir des hiérarchies de classes dont les ressources seront des instances et des propriétés dont l'applicabilité et le domaine de valeur peuvent être contraintes à l'aide des attributs *rdfs : range* et *rdfs : domaine* respectivement. Chaque domaine applicatif peut être associé à un schéma identifié par un préfixe particulier qui correspond à une URI unique. RDFS s'écrit toujours à l'aide du triplet RDF.

4.3.3.3 Objectif de RDF:

Le développement de RDF par le W3C a été entre autres motivé par la perspective des applications suivantes :

- ✓ manipulation et classification des métadonnées Web, afin de fournir des informations sur les ressources Web et les systèmes qui les utilisent.
- ✓ développement de modèles d'information ouverts plutôt que fixés pour certaines applications (par exemple les activités de planification, de description de processus organisationnels, d'annotation de ressources web, etc.)
- ✓ faire avec l'information traitable par machine ce que le Web a fait pour l'hypertexte, en permettant aux informations d'être manipulées en dehors de l'environnement particulier dans lequel elles ont été créées, éventuellement à l'échelle d'Internet. C'est le concept d'interopérabilité des savoirs.
- ✓ optimisation de la coopération entre applications, en permettant de combiner les données de plusieurs applications, pour générer de nouvelles informations.
- ✓ faciliter le traitement automatique de l'information du Web par des agents logiciels, transformant ainsi le web d'un regroupement d'informations uniquement destinées aux humains, en un état de réseau de processus en coopération. Dans ce réseau, le rôle de RDF est de fournir une *lingua franca* compréhensible par tous les agents.

4.3.3.4 Avantages et insuffisances de RDF:

1. Les avantages de RDF :

Grâce à RDF, un service effectue des affirmations prouvant que des entités possèdent des particularités propres à elles avec des valeurs bien précises. Par exemple, l'entité « personne » peut posséder des propriétés comme « est le frère de », « est l'inventeur de » avec des valeurs « nom d'une autre personne », « l'objet inventé ». Cette structure s'avère la voie la plus naturelle pour décrire les données qui vont être traitées par les machines. Le sujet, le verbe et l'objet sont identifiés par des URIs « Universal Resource Identifier ». Pour créer un nouveau concept ou un nouveau verbe, il suffit de définir au sein du WEB un nouveau URI qui fait référence à lui. Toutes les entités même celles qui sont physiques peuvent être pointées par les URIs. Cela signifie que RDF peut être utilisé pour décrire par exemple des téléphones cellulaires ou des télévisions. De cette manière, un moteur de recherche qui sait exploiter le principe et la syntaxe de RDF saura interpréter la nature d'un objet, Est-ce une personne ? – un service ? – une adresse ?

Cela aura pour conséquence : une recherche plus pertinente.

2. Les insuffisances de RDF:

L'utilisation de RDF et de XML ne règle pas tous les problèmes et pour cause, il pourrait y avoir différentes sources utilisant différentes structures, mais exprimant la même information.

Un autre problème peut surgir lorsque la description du contenu faite par l'utilisateur ne décrit pas exactement l'information contenue dans le document. C'est pour essayer de résoudre ses différents problèmes et de rendre RDF plus universel, que les notions d'inférences et d'ontologie ont fait leur apparition dans le domaine du SW.

Il existe aussi un autre obstacle qu'on ne peut pas régler sans l'introduction d'ontologie et qui est lié au problème de la confiance. Où un être humain peut voir que la promotion consistant à vendre un portable Athlon à 20 euros est une escroquerie, un ordinateur ne peut pas faire la différence. Les liens entre métadonnées RDF sont là pour prouver la fiabilité de l'information et pointer sur des certificats d'authenticité de la fiabilité. On peut élaborer une autre méthode, qui analyse toutes les offres indisponibles sur le web (une machine qui possède la même configuration ou proche), et évaluer le degré de la dite offre.

4.3.4. Le langage OWL:

4.3.4.1 Introduction au langage:

On a vu que RDF et RDFS permettent de définir, sous forme de graphes de triplets, des données ou des métadonnées. Cependant, de nombreuses limitations bornent la capacité d'expression des connaissances établies à l'aide de

RDF/RDFS. On peut citer, par exemple, l'impossibilité de raisonner et de mener des raisonnements automatisés (automated reasoning) sur les modèles de connaissances établis à l'aide de RDF/RDFS. C'est ce manque que se propose de combler OWL.

4.3.4.2. Origine et définition:

Le langage d'ontologies OWL [Jim Hendler et al, 2004] est le résultat du développement des précédents langages d'ontologie OIL, DAML et DAML+OIL qui sont des langages de développement d'ontologies pour le web sémantique. Recommandation du W3C, il est dédié aux définitions des classes et des types de propriétés.

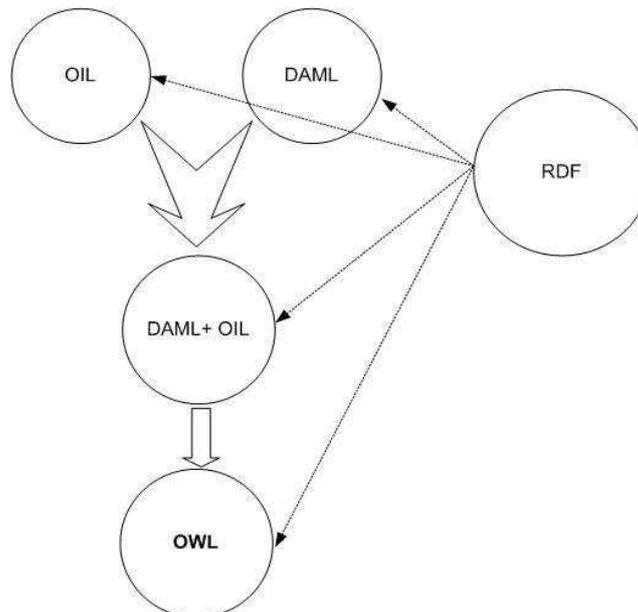


Figure 18 : Origine du langage OWL.

OWL (Ontology Web Language) est le langage actuellement utilisé pour la création des ontologies Web sémantique [Bechhofer et al, 2004]. En se basant sur les fondements du Web sémantique cités dans la section précédente, le W3C fournit un langage qui possède deux caractéristiques importantes. La première est sa base théorique consistante, vu sa relation avec les logiques de descriptions. La deuxième est la possibilité que *OWL* soit utilisé dans un environnement pratique tel que le Web, en se basant sur les standards Web actuels. Néanmoins, les aspects théoriques et formels des logiques de descriptions existent avant la proposition d'*OWL*. C'est sa capacité à utiliser les standards Web qui permet de le considérer comme le langage d'ontologie par excellence dans le domaine du Web sémantique.

4.3.4.3. Différentes déclinaisons d'OWL:

Plus un outil est complet, plus il est, en général, complexe. C'est cet écueil qu'a voulu éviter le groupe de travail WebOnt du W3C en dotant OWL de trois sous-

langages offrant des capacités d'expression croissantes et, naturellement, destinés à des communautés différentes d'utilisateurs :

1. **OWL Full** : (qui n'est pas décidable) destiné aux utilisateurs qui demandent un maximum d'expressivité avec la liberté syntaxique de RDF sans aucune garantie de calculs. Par exemple, une classe peut être traitée comme une collection d'individus et en même temps peut être vue comme un seul individu. OWL Full permet aussi à une ontologie d'augmenter le sens du vocabulaire prédéfini (RDF et OWL).
2. **OWL DL** : dont la sémantique est basée sur celle de la logique de description SHOIN(D), Supporte les utilisateurs qui demandent un maximum d'expressivité tout en maintenant la complétude (garantie de calculer toutes les conclusions) et la décidabilité (tous les calculs doivent finir en un temps fini). OWL DL contient tout les constructeurs du langage OWL mais sont utilisables avec des restrictions (par exemple, lorsqu'une classe peut être une sous classe de plusieurs autre classes, une classe ne peut être une instance d'une autre classe).
3. **OWL Lite** : (se base plutôt sur la logique SHIF) Supporte les utilisateurs qui ont besoin des hiérarchies de classifications et des caractéristiques de contraintes simples. Par exemple, lorsqu'il supporte les contraintes de cardinalité, il permet seulement les valeurs 0 et 1.

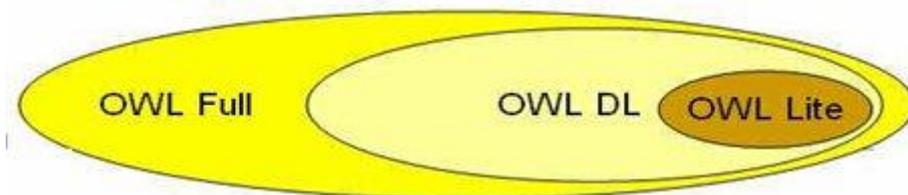


Figure 19 : la relation entre les trois versions de l'OWL

4.3.4.4. Avantages et insuffisances de l'OWL:

1. Avantages du langage OWL :

OWL est, tout comme RDF, un langage XML profitant de l'universalité syntaxique de XML.

Fondé sur la syntaxe de RDF/XML, OWL offre un moyen d'écrire des ontologies web. OWL se différencie du couple RDF/RDFS en ceci que, contrairement à RDF, il est justement un langage d'ontologies. SI RDF et RDFS apportent à l'utilisateur la capacité de décrire des classes (ie. avec des constructeurs) et des propriétés, OWL intègre, en plus, des outils de

comparaison des propriétés et des classes : identité, équivalence, contraire, cardinalité, symétrie, transitivité, disjonction, etc.

Le langage **OWL** reprend toutes les possibilités offertes par les langages qui l'ont précédé.

** Le langage *RDFS* permettait, par exemple, déjà de :

- Créer des classes et les nommer comme *Personne*, *Mère*, *Pays*, *Algérien*, *Etudiant*.
- Etablir une relation de subsumption entre les deux classes *Mère* et *Personne*.
- Indiquer que *tizi ouzou* et *Alger* sont des instances de la classe *Willaya*.
- Déclarer que *age* est une propriété dont le domaine est une *Personne* et le *co-domaine* est le type entier.
- Affirmer que *Younes* est une instance de la classe *Personne* et sa propriété *âge* a la valeur 29.

En plus de ces possibilités, déjà existantes dans *RDFS*, avec *OWL* on peut :

- Déclarer que les deux classes *Personne* et *Pays* sont des classes disjointes.
- Affirmer que *Younes* et *Mounir* sont des individus distincts.
- Déclarer que la propriété *A-Enfant* est l'inverse de la propriété *A-Père*.
- Déclarer que la classe *Algérien* est composée des personnes qui ont *Algérie* comme valeur de la propriété *Nationalité*.
- Déclarer que *Personne- Multi-nationalités* est définie comme étant la classe dont les membres sont les personnes qui ont au moins 2 valeurs de la propriété *Nationalité*.
- Indiquer que la propriété *âge* est fonctionnelle.

Ainsi, *OWL* offre aux machines une plus grande capacité d'interprétation du contenu web que *RDF* et *RDFS*, grâce à un vocabulaire plus large et à une vraie sémantique formelle.

2. Insuffisances du langage OWL:

Très expressif et général, le langage *OWL* présente cependant un certain nombre de limitations de représentation. Parmi ces limitations, on trouve l'absence de structures adaptées pour la modélisation de relations n-aires, et pour la définition des différents rôles attachés aux objets impliqués dans des relations n-aires, l'absence de primitives pour exécuter du code (i.e. réflexe, attachement procédural), pour spécifier des valeurs par défaut, l'impossibilité de définir et d'utiliser des types complexes [A. Napoli et al, 2004]. Il est aussi important de noter qu'il est difficile de travailler avec des valeurs concrètes (nominals), d'exprimer des contraintes lors de la définition d'un concept ou d'une relation, d'exprimer des comportements particuliers, etc.

4.3.4.5. Évolution OWL 2 :

Le langage OWL 2 est une extension ainsi qu'une révision du langage OWL, développé par le groupe de travail W3C Web Ontology Working Group, avec l'objectif de faciliter la création d'ontologies et leur partage via le Web. Il vient compléter le langage OWL par un ensemble d'axiomes (dont l'ajout a été explicitement demandé par diverses communautés d'utilisateurs de OWL, afin de faciliter certaines descriptions terminologiques ou pour augmenter l'expressivité des ontologies), pour lesquels il existe à présent des algorithmes de raisonnement efficaces, et qui sont soutenus par les développeurs d'outils basés sur OWL [P. F. Patel-Schneider and I. Horrocks, 2007]. L'objectif initial de OWL 2 a été d'exploiter les résultats récents de la recherche autour des Logiques de Description afin d'éliminer certaines des limites d'expressivité du langage OWL [[B. C. Grau et al, 2008]].

La structure des ontologies OWL 2 est précisée à l'aide de diagrammes UML [B. Motik et al, 2007], ce qui suggère un rapprochement entre les ontologies et le paradigme objet. En dehors de ce nouveau regard sur les ontologies, Motik et al. définissent également une nouvelle syntaxe pour l'encodage des ontologies, appelée OWL 2 Functional-Style Syntax , [B. Motik et al, 2009]. Celle-ci permet de définir des ontologies plus compactes, et qui sont plus faciles à lire et à éditer par les utilisateurs. La nouvelle syntaxe est similaire à la syntaxe abstraite de OWL, mais elle n'est cependant pas compatible avec celle-ci [B. Motik et al, 2007].

4.3.5. Le langage SWRL:

SWRL (Semantic Web Rule Language) [Francis Lapique, 2006] est un langage (pour niveau logique dans l'architecture du ws) qui enrichit la sémantique d'une ontologie définie en OWL. Il permet contrairement à OWL, de manipuler des instances par des variables ($?x, ?y, ?z$). SWRL ne permet pas de créer des concepts ni des relations, il permet simplement d'ajouter des relations suivant les valeurs des variables et la satisfaction de la règle.

Les règles SWRL sont construites suivant ce schéma: antécédent \rightarrow conséquent. L'antécédent et le conséquent sont des conjonctions d'atomes.

Un atome est une instance de concept, une relation OWL ou une des deux relations SWRL same-as ($?x, ?y$) ou different-from ($?x, ?y$).

Le fonctionnement d'une règle est basé sur le principe de satisfiabilité de l'antécédent ou du conséquent.

Pour une règle, il existe trois cas de figure:

- L'antécédent et le conséquent sont définis. Si l'antécédent est satisfait alors le conséquent doit l'être.
- L'antécédent est vide, cela équivaut à un antécédent satisfait ce qui permet de définir des faits.
- Le conséquent est vide, cela équivaut à un conséquent insatisfait, l'antécédent ne doit pas être satisfiable.

4.3.6. Le langage SPARQL (Protocol and RDF Query Language):

4.3.6.1 Définition :

Alors que RDFS et OWL permettent de définir des ontologies sur le Web Sémantique et RDF de modéliser des assertions en se basant sur celles-ci, il est nécessaire pour en tirer parti de disposer d'un langage de requête adapté. SPARQL – SPARQL (Protocol and RDF Query Language) [Prud'hommeaux et Seaborne, 2008] propose ainsi à la fois un langage et un protocole pour interroger des données modélisées en RDF. Ces travaux s'inscrivent dans la continuité de RDQL [Seaborne, 2004] et l'on peut voir SPARQL comme le SQL du Web Sémantique : "Tenter d'utiliser le Web sémantique sans SPARQL revient à exploiter une base de données relationnelle sans SQL".

SPARQL utilise le principe d'identification de chemins dans un graphe [West, 2000] pour récupérer les résultats d'une requête donnée. Ainsi, une requête SPARQL se compose d'un opérateur (définissant le type de requête), d'un patron (la partie nécessaire pour l'identification des graphes correspondants) et de modifieurs (par exemple, ORDER BY). Une requête peut interroger un ou plusieurs documents RDF, soit par l'utilisation d'un attribut FROM en début de requête, soit par l'intermédiaire d'APIs (Application Programming Interface) qui permettent de considérer simultanément plusieurs sources, soit via l'utilisation d'entrepôts de données RDF associés à des points d'accès SPARQL.

Le langage SPARQL est basé sur la correspondance des patrons de graphe (matching graph patterns). Le patron de graphe le plus simple est le patron de triplets (comme un triplet en RDF) mais il possède la capacité d'exprimer des variables de requête dans les positions du sujet, de la propriété ou de l'objet d'un triplet. D'autre part, SPARQL intègre également des balises (tags) spécifiques telles que le patron de graphe optionnel, l'union et l'intersection des patrons, le filtrage, les opérateurs de comparaison des valeurs... permettant d'effectuer des requêtes plus efficaces et flexibles.

Exemple :

Nous présentons par la suite un exemple de requête en SPARQL qui demande le titre (i.e. ?title) et le prix (i.e. ?price) des ressources dont le prix est inférieur à 30.

| | |
|----|--|
| 1. | PREFIX dc: <http://purl.org/dc/elements/1.1/> |
| 2. | PREFIX ns: <http://example.org/ns#> |
| 3. | SELECT ?title ?price |
| 4. | WHERE { ?x ns:price ?price . |
| 5. | ?x dc:title ?title . |
| | FILTER (?price < 30) . |
| 6. | } |

SPARQL dispose des quatre opérateurs suivants :

– SELECT qui comme son nom l’indique va sélectionner différents éléments selon un patron de requête particulier.

Une requête destinée à récupérer la localisation de NAFTAL pourrait être :

```
SELECT ? pays
WHERE { athena :NAFTAL geonames : locatedIn ? pays }
```

Exemple de requête SPARQL SELECT

– CONSTRUCT qui permet de transformer un graphe RDF en un autre graphe. Par exemple, pour passer de notre modèle à un autre vocabulaire, on peut utiliser:

```
CONSTRUCT { ? entreprise mon_ontologie : situeDans ? pays }
WHERE { ? entreprise geonames : locatedIn ? pays }
```

Exemple de requête SPARQL CONSTRUCT

– ASK qui permet de répondre à une requête, en identifiant si oui ou non le patron recherché est présent dans le graphe interrogé. Ainsi, "NAFTAL est-il situé en Algérie ?" peut s’écrire :

```
ASK { athena :NAFTAL geonames : locatedIn geonames :3043382/ }
```

Exemple de requête SPARQL ASK

– DESCRIBE qui renvoie sous forme d’un graphe RDF l’ensemble des triplets ayant pour sujet la ressource passée en argument. Par exemple, pour connaître l’ensemble des assertions relatives à NAFTAL, on écrira :

```
DESCRIBE athena : NAFTAL
```

Exemple de requête SPARQL DESCRIBE

4.3.6.2. Limite de SPARQL:

SPARQL souffre cependant de différentes limites, notamment par rapport à un langage comme SQL. Par exemple, il ne propose pas pour le moment de fonctions d’agrégat, ni de possibilité d’ajouter des données dans un graphe, SPARQL étant uniquement dédié à des requêtes en lecture seule. Diverses extensions veillent cependant à résoudre ces limites et ajouter de nouvelles fonctionnalités.

Citons pas exemple la recherche par chemins et plus uniquement par triplets (SPARQLer [Kochut et Janik, 2007]) ou l’approximation de requêtes

(iSPARQL [Kiefer et al., 2007]), des fonctionnalités proches étant implémentées dans le moteur SPARQL Corese [Corby et al., 2004].

Pour en revenir aux fonctions d'agrégat, si elles ne sont pas définies par la sémantique de SPARQL, elles sont malgré tout implémentées dans des moteurs comme ARC2 (<http://arc.semsol.org>) ou Virtuoso (<http://virtuoso.openlinksw.com/>).

Enfin, [Pérez et al., 2006] ont montré que certains types de requêtes faisaient partie de la catégorie des problèmes *NP complets* [Garey et Johnson, 1979] étant donné le principe de parcours de graphes qu'utilise SPARQL.

Cependant, il est intéressant de constater que les requêtes peuvent, de façon plus générale, être optimisées en fonction de l'ordre des patrons de requêtes, de manière à réduire successivement le graphe où la requête s'applique [Stocker et al., 2008].

Nous pouvons imaginer qu'à l'avenir, ces stratégies d'optimisations seront implémentées dans la plupart des moteurs SPARQL, à la manière de ce qui se fait pour la réécriture automatique de requêtes dans les systèmes SQL [Kraft et al., 2003].

4.4. Conclusion :

La manipulation des ressources du Web par des machines requiert l'expression ou la description de ces ressources. Plusieurs langages ont été définis à cet effet, ils permettent d'exprimer données et métadonnées, et de disposer d'un modèle abstrait de ce qui est décrit grâce à l'expression d'ontologies (RDFS, OWL). Ces langages sont la base pour la réalisation du Web sémantique, et c'est pour cela que nous avons présenté dans cette section ces différents langages. Cependant, il reste beaucoup de choses à faire pour standardiser le tout, mais comme l'écrit, Tim Berners-Lee, *le Web sémantique est ce que nous obtiendrons si nous réalisons le même processus de globalisation sur la représentation des connaissances que celui que le Web fit initialement sur l'hypertexte.*

5. Le web services sémantiques :

5.1. Introduction :

Les services Web permettent de définir les données transmises entre deux applications, mais aussi comment traiter ces données, et comment les relier à l'interne et à l'externe d'une application logicielle sous-jacente d'une manière standard [Eric Newcomer, 2002]. Un service Web est donc la partie middleware d'une application, tel que derrière le service Web on trouve une implémentation de l'application (ou des applications) qui fournit le service.

Les services Web [Michel Leblanc, 2003] sont des applications qui relient des programmes, des bases de données ou des processus d'affaires à l'aide de XML et de protocoles Internet standards. Ils sont développés à l'aide de langages tel que visual basic, C, C++, Java et autres.

Les services Web permettent aux entreprises et individus de publier des liens vers leurs applications de la même manière qu'ils publient des liens vers leurs pages Web. Conséquemment, les services Web peuvent faire en sorte que toutes les ressources informatiques, dont une entreprise a besoin, soient des ressources distribuées à la grandeur de l'Internet.

Une définition globale d'un service Web, au niveau technologique, peut être [Michel Leblanc, 2003] : Une application logicielle, légèrement couplée, à interaction dynamique, identifiée par une URI, pouvant interagir avec d'autres composantes logicielles, et dont les interfaces et liaisons ont la capacité d'être publiées, localisées et invoquées via XML et protocoles Internet communs.

HTTP est le protocole de base pour le transport des messages entre services Web. Il existe une multitude de standards XML, ZapThink (www.zapthink.com) en a dénombré au-delà de 450 organisés en quatre catégories : les spécifications XML de bases, les spécifications orientées messages, les spécifications orientées document et les vocabulaires de communauté. Selon cette catégorisation, les standards des services Web (SOAP, WSDL, UDDI, BPEL4WS) appartiennent à la catégorie "*spécifications orientées message*" (une présentation détaillée sur les services web et ces standards est disponible dans l'annexe A).

5.2. Problématique initiant au web services sémantiques :

Les services Web sont une technologie pour l'intégration et l'interopérabilité des systèmes répartis. Basés sur le standard XML, ils sont caractérisés par leurs indépendances aux plates formes et aux systèmes d'exploitation, ce qui a impliqué leur adoption par les différentes organisations commerciales et industrielles offrant leurs services à travers le Web, et par conséquent

l'augmentation du nombre de services offerts. La découverte de service devient de ce fait un des aspects les plus importants relatifs aux services Web.

La technologie fondamentale pour la découverte de services Web est le registre UDDI. Destiné à être utilisé par les utilisateurs humains, UDDI permet la recherche (ou le passage en revue des descriptions des services) et la sélection manuelle des descriptions des services Web. UDDI fournit une API de recherche basée mots clés.

De plus il permet aussi de faire la recherche des services en se basant sur leurs descriptions WSDL. Cependant la recherche dans UDDI (comparaison de la requête avec les descriptions des Services Web) n'est faite qu'au niveau syntaxique, cette méthode présente quelques limitations ; la recherche syntaxique ne permet pas de trouver le service demandé à chaque fois ; De plus, un agent logiciel ne peut pas examiner la description textuelle destinée pour des humains, il ne peut pas distinguer entre deux services Web différents ayant la même description syntaxique, ce qui présente un handicap pour l'automatisation de la découverte, composition, et collaboration des services Web.

Pour permettre l'automatisation (qui est un concept clé qui doit être présent à chaque étape du processus de conception et de mise en œuvre des web services) des diverses tâches liées aux services Web, l'idée consiste à enrichir les descriptions des services Web (limitées qu'aux aspects fonctionnels) par d'autres informations supplémentaires compréhensibles par machines.

Ces informations sont des données sur comment interpréter les descriptions des services Web. En d'autres termes, c'est la sémantique des descriptions des services Web, puisque deux services Web peuvent avoir la même description syntaxique mais avoir deux sens différents. Des services Web dotés de description sémantique sont des services Web sémantiques.

5.3. Présentation et objectifs du web service sémantique :

Le Web sémantique est un projet qui a été créé pour la modélisation des données hétérogènes, par contre la modélisation des programmes disponibles sur le Web, c'est-à-dire les services Web constitue un nouveau projet relativement récent s'appelle les services Web sémantiques, ce dernier s'intéresse principalement d'ajouter de la sémantique aux services Web. Des services Web sémantiques seraient intuitivement des programmes dont les effets sur leur environnement seraient connus, et dont les données manipulées possèderaient aussi une sémantique. [Bruijn et al., 2008]

Les web services sémantiques se situent à la convergence de deux domaines de recherche importants qui concernent les technologies de l'Internet : le Web sémantique et les web services. Le Web sémantique s'intéresse principalement aux informations statiques disponibles sur le Web et les moyens de les décrire de manière intelligible pour les machines. Les web services, quant à eux, ont

pour préoccupation première l'interopérabilité entre applications via le Web en vue de rendre le Web plus dynamique.

L'objectif visé par la notion de web services sémantiques est de créer un Web sémantique de services dont les propriétés, les capacités, les interfaces et les effets sont décrits de manière non ambiguë et exploitable par des machines et ce en utilisant les couches techniques sans pour autant en être conceptuellement dépendants. La sémantique ainsi exprimée permettra l'automatisation des fonctionnalités suivantes qui sont nécessaires pour une collaboration interentreprises efficace [Patrick Kellert et Farouk Toumani, 2003]:

- Découverte de services Web, c'est à dire localisation automatique des services Web qui fournissent une fonctionnalité particulière et qui répondent aux propriétés demandées par l'utilisateur. Pour pouvoir effectuer une découverte automatique, le procédé de découverte devrait être basé sur la similitude sémantique entre la description déclarative, faite par l'utilisateur, du service demandé et celle du service offert.
- Processus de description et de publication des services ;
- Sélection des services ;
- Composition des services ;
- Fourniture et administration des services ;
- Négociation des contrats.
- L'interopérabilité des services Web.

5.4. Architecture Étendue :

L'architecture étendue correspond à l'architecture de référence agrémentée d'une couche de sémantique. Le Figure qui suit décrit un exemple d'une telle architecture. La pile est constituée de plusieurs couches, chaque couche s'appuyant sur un standard particulier. Nous retrouvons, au-dessus de la couche de transport, les trois couches formant l'infrastructure de base. La couche Sécurité s'occupe de la sécurité de toutes les couches. La couche Transactions s'occupe de l'interopérabilité transactionnelle des SW. La couche Administration s'occupe de l'administration des SW. La couche QoS (Quality of Service) s'occupe des contrats entre un fournisseur de service et un consommateur de service [Patrick Kellert1 et Farouk Toumani, 2010]

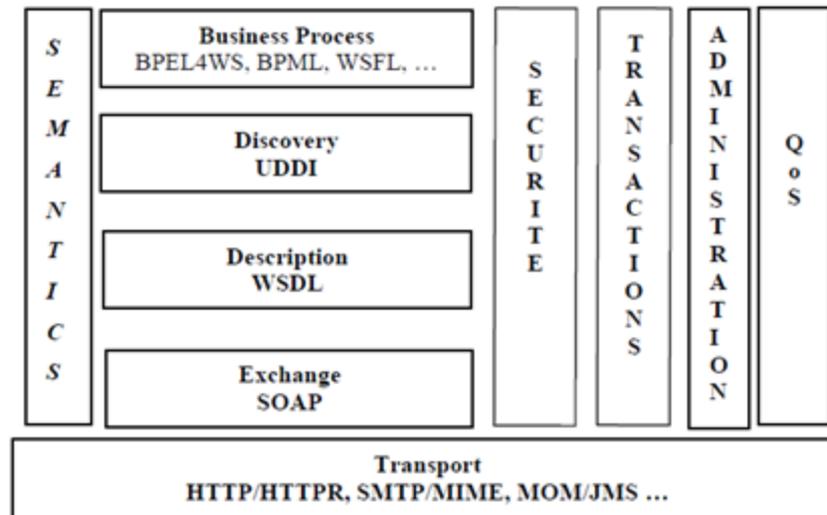


Figure 20 : Pile des SW [Patrick Kellert1 et Farouk Toumani, 2010]

5.5. Approches pour la réalisation des services Web sémantiques :

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

5.5.1. DAML-S :

DAML-S [DAML Services Coalition, 2002] est une ontologie de web services basée sur le langage DAML+OIL [Horrocks *et al.*, 2002a ; Horrocks, 2002b]. DAML-S définit un ensemble de classes et de propriétés spécifiques à la description des services. Dans une description DAML-S, la section *profile* spécifie les informations relatives aux capacités d'un service. Elle consiste dans trois types d'informations :

- Une description du service compréhensible par les humains;
- Le comportement fonctionnel du service représenté comme une transformation des paramètres d'entrée du service vers ses paramètres de sortie ;
- Plusieurs attributs fonctionnels qui spécifient des informations supplémentaires relatives au service (*ex.*, prix du service).

Dans l'approche DAML-S, la section *profile* est utilisée à la fois par les fournisseurs pour publier leurs services et par les clients pour spécifier leurs besoins. Par conséquent, elle constitue l'information utile pour la découverte et la composition de services.

5.5.2. WSDL-S (Web Service Description Language-Semantic)

WSDL-S [J.Miller et al, 2005] est un langage de description sémantique des services Web. Une description WSDL-S de service Web est une description WSDL augmentée de sémantique, cette sémantique est ajoutée en deux étapes :

La première étape consiste à faire référence, dans la partie définition de WSDL, à une ontologie dédiée au service à publier. La seconde étape consiste à annoter les opérations de la définition WSDL de sémantique en ajoutant deux nouvelles balises ; la balise *Action* et la balise *Contrainte*.

- La balise *Action* permet de représenter l'action de l'opération
- La balise *Contrainte* représente les prés et post condition d'une opération

```
<operation name = "checkStatus" pattern="mep :in-out" >
  <action element = "rosetta :QueryOrderStatus" />
  <input messageLabel = "statusQuery" element = "rosetta
:PurchaseOrderStatusQuery" />
  <output messageLabel = "status" element = "rosetta
:PurchaseOrderStatusResponse" />
  <pre condition="PurchaseOrderStatusQuery.orderStatusDoc. ?PurchaseOrder
!=null" />
</operation>
```

Figure 21 : Opération WSDL augmentée de sémantique

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

OWL-S est un langage d'ontologie pour services Web. Il est basé sur le langage OWL [Jim Hendler et al, 2004]. Cette ontologie a pour objectif de décrire de façon non ambiguë les services Web de telle sorte qu'un agent logiciel puisse exploiter automatiquement ces informations. OWL-S permet : La découverte automatique, la composition et l'interopérabilité de services Web ainsi que la surveillance automatique de leur exécution.

OWL-S décrit un service à l'aide des trois classes suivantes :

- *ServiceProfile* : définit le service Web.
- *ServiceModel* : définit le fonctionnement du Web service.
- *ServiceGrounding* : définit comment accéder au service Web.

ServiceProfile

Pour décrire un service OWL-S définit la classe *ServiceProfile*. La classe *ServiceProfile* spécifie trois informations.

- Nom du service, contacts et description textuelle du service : le nom du service est utilisé comme identificateur du service, tandis que les informations contacts et la description textuelle sont destinées aux utilisateurs humains.
- Description fonctionnelle du service : Elle spécifie ce que le service exige en termes d'entrées (inputs) attendues et de résultats produits en sortie (outputs). Elle indique également les prés conditions et les effets du service.
- Classification taxinomique.

ServiceModel

La classe *ServiceModel* décrit le fonctionnement du service Web. Ceci est fait en exprimant les transformations faites par le service Web sur les données (input à output), et transformation d'état (pré conditions et effets).

Les services Web peuvent être modélisés avec OWL-S en tant que processus grâce à la classe *Process*. La classe ainsi définie est une sous-classe de *ServiceModel*. Pour décrire un processus, on spécifie ces Entrées, Sorties et ses états. Les transitions d'un état à un autre sont décrites par les pré-conditions et les effets de chaque processus.

Il existe trois types de processus :

1. Les processus atomiques (*AtomicProcess*) : exécutable en une seule étape, un processus atomique ne peut pas être décomposé de façon plus profonde. Son exécution correspond à une unique avancée dans l'exécution du service, il est directement invoqué par l'utilisateur du service.
2. Les processus composites (*CompositeProcess*) : un processus composite est constitué par l'assemblage d'autres processus (composite ou non composite). Les processus composites associent des processus à l'aide de structures de contrôle permettant de décrire leur logique d'exécution. Les structures de contrôles sont les suivantes :
 - séquence (*Sequence*) représente une suite ordonnée de processus.
 - exécutions concurrentes de processus sont décrites par *Split*.
 - synchronisation peut être décrite par *Split+Join*.
 - exécution de processus sans ordre particulier avec *Unordered*.
 - Le choix est décrit par *Choice*.
 - Les branchements conditionnels du type si/alors/sinon sont décrits par *If-Then-Else*.
 - *Repeat*, *Iterate* et *Repeat-Until* permettent d'effectuer des itérations.
3. Les processus simples (*SimpleProcess*) : Les processus simples ne sont pas invocables mais comme les processus atomiques, leurs exécutions s'effectuent en une seule étape. Les processus simples sont employés comme éléments d'abstraction ; un processus simple peut être employé pour fournir une vue d'un certain processus atomique, ou une représentation simplifiée d'un certain processus composé.

ServiceGrounding

La classe OWL-S *ServiceGrounding* définit les détails techniques permettant d'accéder au service Web. Les deux premières classes *ServiceProfile* et *ServiceModel* d'une description OWL-S s'attachent à abstraire la représentation d'un service Web. *ServiceGrounding* est la forme concrète d'une représentation

abstraite, elle fournit les détails concrets d'accès au service Web, tels les protocoles, les URIs, les messages envoyés, etc.

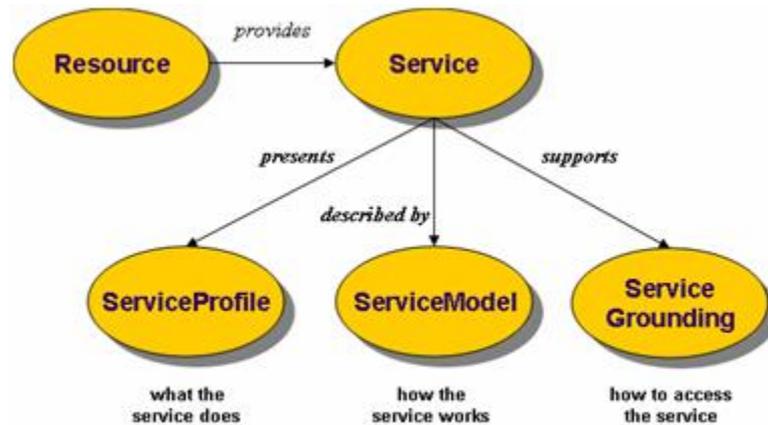


Figure 22: Organisation de Module d'OWL-S

Les concepteurs de OWL-S ont choisi de combiner OWL-S avec WSDL. L'objectif de cette combinaison est de tirer profit de chacun d'entre eux. Les types abstraits des messages employés dans les messages WSDL sont définis sémantiquement à l'aide de classes OWL. Les bindings WSDL servent ensuite à définir comment ces messages sont formatés. Les relations entre OWL-S et WSDL sont les suivantes :

- Un processus atomique OWL-S correspond à une opération WSDL.
- Les entrées et sorties d'un processus atomique OWL-S correspondent chacune à un message WSDL (d'entrée ou de sortie).
- Les types des messages d'un processus atomique OWL-S correspondent à l'élément *Types* d'une description WSDL.

La classe OWL-S Grounding sert de base à *WsdI grounding* qui contient une liste OWL d'instances de *WsdIAtomicProcessGrounding*. Chaque instance de cette dernière classe se réfère à des éléments de la spécification WSDL en utilisant un jeu de propriétés effectuant la liaison.

Au niveau des éléments propres à WSDL, des attributs XML OWL-S sont ajoutés aux balises WSDL. Ces extensions sont :

- *owl-s-parameter* (message WSDL) : indique le nom qualifié (QName) de l'objet OWL-S utilisé dans la définition d'un paramètre de message.

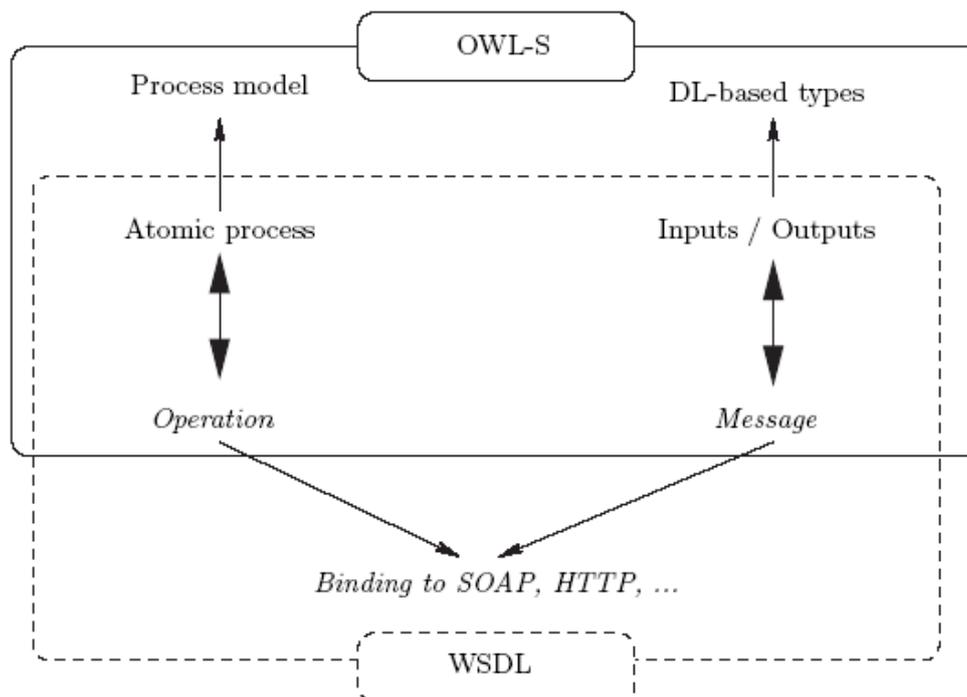


Figure 23 : Relations entre OWL-S et WSDL

- `encodingStyle` (binding WSDL) : spécifie la façon d’encoder une classe OWLS dans un message transmis.
- `owl-s-process` (opération WSDL) : indique le processus atomique WSDL attaché à l’opération.

Avec OWL-S, un service Web est une entité qui présente une description de ce qu’il offre, comment il fonctionne et enfin comment on peut l’employer.

5.5.4. IRS-II (Internet Reasoning Service):

IRS-II [Enrico Motta et al., 2003] est une architecture pour les services Web sémantiques. IRS-II est basée sur la structure UPML (Unified Problem Solving Method Development Language) [Crubezy-Met al., 2003] où diverses ontologies sont définies :

- Ontologie du domaine (*Domain model*) : elle permet de décrire le domaine d’une application (ex : véhicules, maladies).
- Ontologie de tâche à résoudre (*Task models*) : elle fournit une description générique de la tâche à résoudre, spécifie les types d’entrées (input) et sortie (output), le but à atteindre et les pré conditions à satisfaire.
- Ontologie des méthodes de résolution d’un problème (*Problem Solving Methods (PSMs)*) : sépare la description de ce qu’un service fait des paramètres et des contraintes d’une mise en oeuvre particulière.
- Liens (*bridges*) : permettent la correspondance entre les différents modèles d’une application.

Les principaux composants de l'architecture IRS-II sont : le serveur IRS-II (*IRS-II server*), éditeur de services (*IRS-II Publisher*) et la partie client (*IRS-II Client*). Ces trois composants interagissent entre eux via le protocole SOAP.

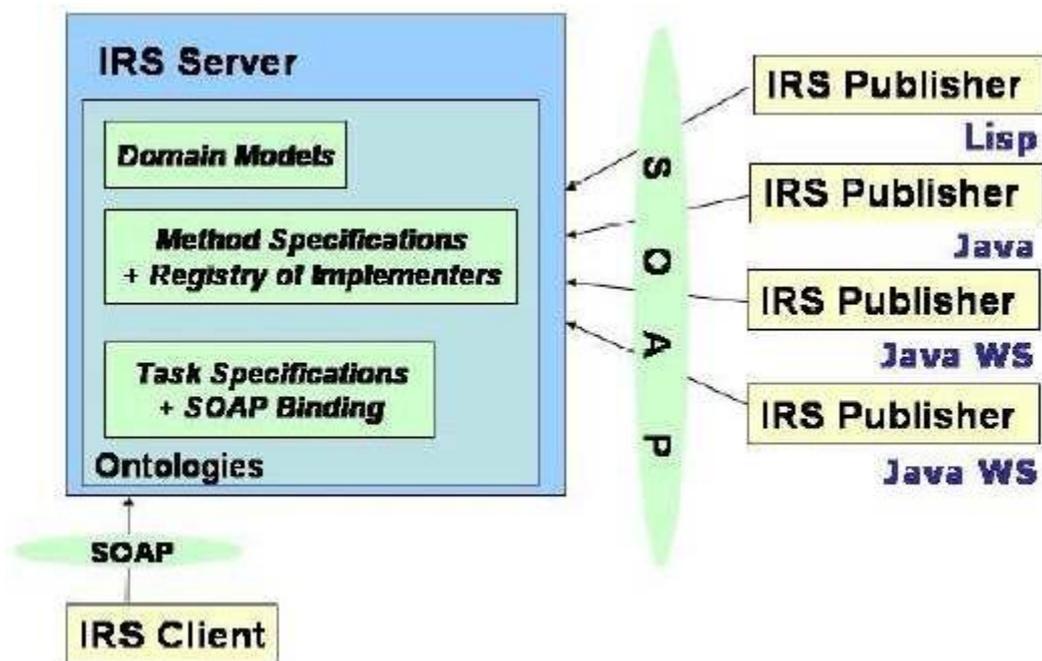


Figure 24 : Architecture d'IRS-II

Le serveur IRS-II contient les descriptions des services Web sémantiques. Ces descriptions sont faites sur deux niveaux. Au niveau connaissance, une description est sauvegardée selon la structure UPML des tâches, PSM et l'ontologie du domaine. De plus, deux types de mise en correspondances sont utilisés pour lier les descriptions aux niveaux connaissances à un service Web spécifique.

Le composant *éditeur* (*IRS-II Publisher*) a deux fonctions. Premièrement, il permet de lier les services Web à leurs descriptions sémantiques respectives dans le serveur. Chaque PSM est associé à exactement un service Web. Un service Web peut être associé à plusieurs PSM, puisque un service Web peut avoir plus d'une fonction.

Deuxièmement, il génère automatiquement un programme qui enveloppe le code LISP ou Java du service Web, afin de l'invoquer, comme c'est le cas d'un service Web dans sa description WSDL.

Un client IRS-II invoque un service Web en envoyant une requête de la tâche à traiter. Sur la base de cette tâche le serveur sélectionne le PSM approprié et invoque le service Web avec lequel est lié ce PSM.

5.5.5. WSFM :

WSFM comprend quatre éléments principaux [Gardien, 2002] [Dallons, 2004] :

- Des ontologies qui fournissent la terminologie utilisée par les autres éléments ;
- Un répertoire d'objectifs qui définit les problèmes qui doivent être résolus par les services Web ;
- Des descriptions des services qui définissent les différents aspects liés aux services Web ;
- Des médiateurs qui sont en charge des problèmes d'interopérabilité.

L'implémentation de WSFM est partagée en deux projets : le projet SWWS; et le projet WSMO.

L'objectif de SWWS est de définir une structure de description et de découverte de services Web, ainsi qu'une plateforme de médiation pour services selon une architecture conceptuelle. Le projet WSMO permettra de raffiner WSMF, en plus, du développement d'ontologie formelle de service et un langage pour les services Web sémantiques [Casati et al., 2000].

5.5.6. WSMO

Le WSMO est un projet de l'union européenne qui constitue un cadre compréhensible pour SESA et définit un modèle conceptuel avec un langage de spécification, comme, il fournit une implémentation avec plusieurs outils. L'implémentation WSMX fournit un environnement de développement et d'exécution pour SESA à base de WSMO. L'approche WSMO définit les ontologies, les services web, les buts et les médiateurs comme ses éléments de haut niveau avec un modèle conceptuel qui prend en charge ces derniers. Ce modèle conceptuel a pour but la structuration des annotations sémantiques des services.

WSMO permet la description des services, en considérant les aspects suivants [Newcomer, 2004] :

- Les propriétés non fonctionnelles incluent des propriétés telles que la performance, la fiabilité, la sécurité et la robustesse du service Web;
- La fonctionnalité du service est décrite en termes de préconditions, postconditions, hypothèses et effets ;
- Une interface de description d'un service Web est constituée d'informations telles que les erreurs gérées par le service, une description d'orchestration si le service fait appel à d'autres services, une description de la conversation du service appelée échange de message, et des stratégies de compensations utilisées dans le cas où certaines opérations exécutées par le service doivent être annulées;
- Le grounding spécifie les informations concrètes pour l'accès au service Web. Il est clair que de nombreux aspects définis dans cette ontologie sont également couverts par OWL-S. Par exemple, les préconditions, postconditions, hypothèses et effets dans WSMO. D'autre part, les deux ontologies se basent sur WSDL pour la liaison. Un autre point commun

est la description de la conversation d'un service dans les interfaces WSMO qui est également décrite dans le process Model de OWL-S. Ceci dit qu'il existe quelques différences entre les deux approches.

Contrairement à OWL-S qui n'exige aucun style architectural, l'approche proposée par WSMO est basée sur l'architecture WSMF. Cette architecture définit notamment la notion de médiateurs. Les médiateurs sont des composants logiciels utilisés par les services Web pour répondre à des problèmes d'interopérabilité tels que l'incompatibilité des types. D'autre part, dans WSMO, certaines propriétés telles que les hypothèses et effets sont exprimées dans le langage formel F-Logic qui permet le raisonnement automatique sur les valeurs de ces propriétés. Dans OWL-S, des propriétés telles que les préconditions et effets doivent également être exprimées sous forme d'expressions logiques. Ceci dit, à l'heure actuelle, ces expressions sont exprimées sous formes de structures XML [Burststein et al., 2004].

5.6. Conclusion et perspectives :

Les services Web sémantique sont un domaine de recherche émergent. A l'état actuel il existe des technologies pour le développement des services Web. Cependant, ces technologies exigent que l'utilisateur humain interagisse avec le système tout au long du processus de découverte de services Web. Les technologies du web sémantique ont été utilisées pour palier à ce problème en enrichissant les services Web de sémantique, ce qui permet l'automatisation des divers aspects relatifs aux services Web.

Les avantages de l'utilisation du Web sémantique pour la description des services Web sont nombreux. En plus, de rendre l'interface du service Web accessible automatiquement par des machines, ils permettent également, la description de propriétés non fonctionnelles telles que la qualité de services, les contraintes de sécurité, et l'intégration effective des services Web dans des applications industrielles, d'une manière uniforme compréhensible par tous.

Plusieurs avenues de recherche méritent d'être explorées afin de mieux définir les frontières et de quantifier les approches des services Web sémantique. Après la présentation de ces standards, les travaux futurs doivent couvrir les secteurs suivants :

- Fournir un modèle compréhensible de description des services Web.
- Etablissement d'une connexion étroite des efforts industriels tel que XML, RDF, WSDL, WSFL et les efforts de recherche tel que DAML+OIL, OWL, DAML-S etc., popularisation des approches du Web sémantique dans l'industrie et finalement indication des besoins industriels pour les nouvelles technologies.

- Définition d'un modèle de découverte des services Web qui offre une base d'ontologie avancée et une métadonnée destinée à la découverte des services qui dépasse les limites de UDDI.
- Application et amélioration des technologies existantes du Web sémantique dans les applications du monde réel.
- Manipulation sûre et robuste : Le caractère distribué et individuel du Web sémantique fait que son contenu ne saurait être exempt d'erreurs, il faut donc savoir s'en accommoder.

6. Quelques applications du web sémantique:

6.1. Introduction :

Nous avons vu que le Web sémantique était une infrastructure pour permettre l'utilisation de connaissances formalisées en plus du contenu informel actuel du Web. Cependant, il ne faut pas se limiter à cette infrastructure mais plutôt aux applications développées sur celles-ci. Parmi les applications du Web sémantique, nous pouvons citer :

- Les mémoires d'entreprise,
- Les applications médicales,
- Le e-learning,
- La recherche d'information,
- Les wikis sémantiques

6.2. Les mémoires d'entreprise :

A l'heure actuelle, les sources d'informations et de connaissances dans les entreprises deviennent de plus en plus importantes et elles jouent aussi un rôle crucial pour le développement de l'entreprise. Le partage du travail nécessite un partage de connaissances au sein des entreprises et ce besoin peut devenir encore plus crucial dans l'avenir. Pour réussir, les entreprises doivent bien gérer ces sources de connaissances et supporter leur utilisation effective.

Les connaissances développées lors des activités d'une entreprise peuvent être exprimées explicitement dans des actes, des documents, des expériences. Nous appelons *mémoire d'entreprise* ou mémoire organisationnelle tous ces types de connaissance, des ressources et les informations cruciales de l'entreprise.

Pour faciliter et améliorer l'accès, le partage, la réutilisation de cette mémoire d'entreprise, voire permettre la création ou la déduction des nouvelles connaissances à partir de celle-ci, il est nécessaire de disposer des moyens et des outils permettant de matérialiser la mémoire et d'indexer les contenus dans cette mémoire d'entreprise [Dieng et al., 2004].

En faisant l'analogie entre les ressources d'une mémoire d'entreprise et celles du Web, l'on peut employer les technologies du Web pour la gestion, la représentation et l'exploitation des connaissances dans une organisation, l'implémentation et l'utilisation de l'Intranet au sein de l'organisation.

Avec l'évolution du Web vers le Web sémantique, les nouvelles technologies pour ce dernier peuvent aussi être appliquées pour l'organisation, les mémoires d'entreprise sont construites sous forme du Web sémantique d'entreprise [Dieng et al., 2004]. Les technologies du Web sémantique sont employées pour réaliser le Web sémantique d'entreprise : les annotateurs annotent sémantiquement des ressources d'entreprise : description du contenu des documents, compétences des personnes, caractéristiques des services,... en employant le vocabulaire

commun et partagé au sein de l'entreprise (l'ontologie de cette entreprise). Les utilisateurs dans l'entreprise, grâce à cette ontologie commune, pourront alors interpréter des annotations créées par des annotateurs grâce aux significations des termes, des concepts bien prédéfinies dans l'ontologie. Cela permet aux anciens utilisateurs de partager leurs connaissances, leurs expériences, leurs compétences et aux nouveaux utilisateurs d'apprendre, de consulter et de s'adapter mieux à l'entreprise. Cela facilite les utilisateurs dans leurs tâches individuelles et collectives.

Les technologies du web sémantique seront fortement contributives pour les services suivant :

- Accès des employés en situation de mobilité à la mémoire de l'entreprise;
- Partage entre employés d'une même communauté (P2P – *Peer-to-Peer computing*) où la construction de la connaissance (Ontologie et annotations) s'opère de manière naturelle et consensuelle ;
- Intégration des mémoires d'entreprises décentralisées et multinationales ;
- Formation professionnelle continue (*e.Learning*) sur le portail de l'entreprise sur lequel l'employé se voit offrir des parcours de formation diversifiés et surtout personnalisés.

Exemples d'application du web service d'entreprise :

Nous présentons dans cette section des projets basés sur l'approche du Web sémantique d'entreprise :

Samovar :

Samovar [Golebiowska et al., 2001] est un système/méthode de capitalisation de connaissances dans le domaine automobile. L'objectif de ce projet était d'améliorer l'exploitation des informations stockées dans un système de gestion de problèmes afin de les mettre à disposition pour les projets futurs chez Renault.

L'approche Samovar repose sur l'utilisation de plusieurs ontologies (Problème, Pièce, Prestation et Projet) construites semi-automatiquement à partir de sources différentes, à savoir (a) les interviews des experts de Renault, (b) les données structurées contenues dans les bases de données et (c) les données textuelles constituées par les discussions des concepteurs stockées dans des champs textuels de la base de données. Ces ontologies sont représentées en RDFS et servent à créer des annotations RDF sur la base des descriptions des problèmes, et à faciliter la recherche d'informations en utilisant le moteur de recherche sémantique Corese.

CoMMA :

CoMMA (Corporate Memory Management through Agents) est un projet européen qui a permis de construire une mémoire d'entreprise matérialisée dans une base documentaire annotée par des annotations sémantiques basées sur

l'ontologie O'CoMMA [Gandon, 2002]. Ce projet visait à développer une société d'agents coopérant et guidés par l'ontologie pour la recherche d'information dans cette mémoire d'entreprise, l'ajout d'annotations dans la base d'annotations et l'interaction avec les utilisateurs en intégrant un moteur de recherche sémantique.

Le projet CoMMA a traité aussi l'aspect distribué de la mémoire d'entreprise. Des agents dédiés aux connexions ont été proposés, ces agents proposaient des services de pages jaunes et de pages blanches pour fournir des informations sur un agent pouvant offrir un service particulier.

CoMMA se distingue par son approche basée sur l'intégration de plusieurs technologies émergentes (Langages du web sémantique : XML, RDF-S, Systèmes multi-agents, Apprentissage symbolique, Ingénierie des connaissances). Chacune de ces technologies apporte des éléments de solution pour la réalisation, la gestion et l'exploitation d'une mémoire organisationnelle distribuée et hétérogène. Les Intranets reposant sur la technologie Internet, bénéficient des progrès du web sémantique.

CORESE :

CORESE (COncceptual REsource Search Engine) est un moteur de recherche sémantique dédié au langage RDF(S) et qui peut être utilisé pour interroger les différentes ressources d'un web sémantique d'entreprise [Corby et al., 2004]. Ce moteur de recherche rend possible les inférences sur des annotations en RDF basées sur une ontologie. Il repose sur une correspondance entre RDF(S) et le formalisme des graphes conceptuels. Il dispose en outre d'une base de règles [Corby et al, 2006] et permet des requêtes SPARQL [Corby et Faron, 2007].

MEAT :

MEAT (Mémoire d'Expériences pour l'Analyse du Transcriptome) est un projet de construction d'une mémoire d'expériences, qui vise à faciliter la validation et l'interprétation des expériences poudres à ADN. L'objectif principal de cette mémoire est d'organiser les connaissances de ce domaine, qui proviennent de plusieurs sources hétérogènes (documents, base de données, connaissances humaines...) afin de faciliter leur partage et leur réutilisation. Ce projet vise donc à offrir aux biologistes un accès transparent et 'intelligent' à l'ensemble de ces connaissances [Khelif, 2006].

6.3. Les applications médicales :

La médecine est un des domaines d'applications privilégiés du Web sémantique comme elle l'a été, à une autre époque, des techniques de l'Intelligence Artificielle, en particulier les systèmes experts. C'est en effet un domaine complexe où les informations à partager sont nombreuses et où il n'y a pas ou

peu de solutions algorithmiques à ce partage comme à l'usage des connaissances, en particulier cliniques.

Ainsi, un des principaux mécanismes du Web sémantique qui est la description de ressources via des annotations est de la plus grande importance en bioinformatique, plus particulièrement autour des questions de partage des ressources génomiques.

Exemples :

1. Description de ressources Web en génomique fonctionnelle :

La recherche dans le domaine de la génomique fonctionnelle nécessite d'accéder à une multitude de bases de données et de connaissances accessibles via le Web, mais hétérogènes dans leur structure et leur terminologie [Froidevaux C., Cohen Boulakia S., 2002]. Aujourd'hui, le problème qui est posé est celui d'une collecte manuelle d'informations, qui ne peut être que partielle ou incroyablement fastidieuse, étant donné le nombre de sources disponibles et leur diversité, versus une collecte d'informations automatisée « intelligente ». L'intérêt du Web sémantique en génomique fonctionnelle est d'apporter suffisamment de renseignements sur les ressources au moyen des métadonnées, de décrire leur contenu de manière à la fois formelle et signifiante via des ontologies de domaine, de telle sorte que des agents intelligents de recherche sur le Web puissent sélectionner les informations pertinentes pour une question donnée.

La première phase se situe donc au niveau des choix du (ou des) langage(s) et du schéma de description. Les informations à décrire sont non seulement des ressources Web mais aussi des sites hébergeant les différentes bases de données. Pour chacun d'entre eux, le but est a priori de concevoir un modèle de description « type ». C'est la première étape qui nous permettra ensuite, à partir d'un environnement sémantique obtenu via l'ontologie de domaine, d'orienter intelligemment le biologiste vers telle ou telle base de données suivant la forme de sa requête.

6.4. E-learning:

6.4.1. Définition de E-Learning (Electronic -Learning) [Salemkour & kerbout, 2009]

Plusieurs termes sont utilisés pour traduire le terme Anglais E-learning. Les termes corrects sont apprentissage en ligne, apprentissage virtuel et apprentissage électronique. Parfois certains emploient le terme E-formation, un calque de l'anglais.

- "Le e-learning est un processus d'apprentissage à distance s'appuyant sur des ressources multimédias, qui permet à une ou plusieurs personnes de se former à partir de leur ordinateur.

Les supports multimédias utilisés peuvent combiner du texte, des graphismes en 2 ou 3 dimensions, du son, de l'image, de l'animation et même de la vidéo."

- "Mode d'apprentissage basé sur l'utilisation des nouvelles technologies, permettant l'accès à des formations en ligne, interactives et parfois personnalisées, diffusées par l'intermédiaire d'internet, d'un intranet ou autre média électronique, afin de développer les compétences, tout en rendant le processus d'apprentissage indépendant de l'heure et de l'endroit."

Aujourd'hui, le e-learning est lié spécialement à deux secteurs :

- Des universités qui offrent des alternatives de formation à distance.
- Des grandes entreprises qui l'utilisent avec le KM (**K**nowledge **M**anagement ou management des connaissances) pour former et actualiser les connaissances de ses employés de manière plus rapide.

6.4.2 Apports du web sémantique :

Le e-learning est considéré parmi les domaines auxquels le Web sémantique peut apporter une véritable amélioration que ce soit dans la recherche d'information, ou dans la réutilisation des ressources pédagogiques ou même dans la personnalisation des parcours d'apprentissage. Il est considéré aussi comme une plateforme adéquate pour implémenter un système e-learning, du moment qu'il fournit tous les moyens pour le développement d'ontologies permettant l'annotation du matériel d'apprentissage.

6.5. La recherche d'information :

Le web, grâce à sa simplicité d'édition et de consultation, a rendu l'internet convivial et accessible à tous, Cependant malgré cet engouement qui a conduit à un développement énorme de l'offre, rien n'a été fait pour garantir qu'un document publié sera retrouvé, visible et lisible. Depuis une quinzaine d'années des systèmes de recherche d'informations (SRI), nombreux et variés, destinés au grand public, ont été développés pour explorer le web.

De nos jours, les moteurs de recherche sont sans aucun doute la principale ressource à disposition des utilisateurs pour la recherche d'informations sur internet. Grâce aux moteurs de recherche, il suffit d'écrire un ou plusieurs mots clés concernant le sujet qui nous intéresse et en quelques secondes nous obtenons une liste de pages web qui contiennent les mots demandés. Des modèles de recherche d'information (booléen, vectoriel, probabiliste, etc.) ont été développés pour des documents textuels classiques depuis déjà plus de 30 ans. Ces modèles ont été très étudiés dans le contexte de documents classiques : atomiques, plats et indépendants.

Mais 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. Autrement dit, la recherche plein texte n'est pas toujours efficace, ne serait-ce parce qu'il existe des variantes lexicales et des synonymes considérés comme étant des termes différents.

Donc la problématique qui se pose est celle d'une recherche d'information intelligente où l'indexation devrait reposer sur la sémantique des ressources comme étant l'explication de structures et de concepts contenus dans les documents numériques qui leur sont associés.

L'intérêt du web sémantique est d'une part d'apporter suffisamment de renseignements sur les ressources, en ajoutant des annotations sous la forme de métadonnées et d'autre part, de décrire leur contenu de manière à la fois formelle et signifiante à l'aide d'une ontologie pour être interprétables aussi bien par les humains que par les machines.

De nombreux systèmes de recherche d'information utilisent aujourd'hui les technologies du web sémantique pour faciliter la recherche d'information tel que le langage RDF pour l'annotation de ressources, des ontologies pour normaliser la sémantique des annotations...etc.

Exemples :

- ✓ swoogle (moteur de recherche sémantique de Google).
- ✓ Yahoo SearchMonkey
- ✓ Google Rich Snippets
- ✓ Google Squared
- ✓ Sindice

Les applications de recherche d'informations dans le cadre du web sémantique peuvent être valorisées de plusieurs façons. On peut utiliser des ontologies de façon simple pour améliorer la pertinence des recherches sur le web : le moteur de recherche peut ne chercher que les pages faisant référence à un concept précis au lieu de celles qui utilisent des mots-clés ambigus. Mais des applications plus avancées utiliseront des ontologies plus complexes pour associer l'information d'une page web à des structures de connaissance et à des règles d'inférence. Elles peuvent aussi demander des garanties supérieures pour les traitements automatiques à base de connaissances qui se trouvent dans le web sémantique.

6.6. Les wikis sémantiques :

6.6.1. Introduction :

Imagés en 1995 par Ward Cunningham, les wikis sont des sites Web permettant la création et l'édition collaborative de contenus de manière simple [<http://fr.wikipedia.org/wiki/Wiki>]. La popularité de Wikipédia montre l'importance de ce type de systèmes. En parallèle, le développement du Web sémantique depuis le début des années 2000 a ouvert de nouvelles

perspectives (qu'ont a très bien détaillées dans les sections précédente); Ainsi sont nés les wikis sémantiques, dont la particularité consiste à formaliser le contenu des articles, notamment en caractérisant les relations entre ceux-ci. De nombreux moteurs de wiki sémantique ont vu le jour depuis 2003, avec plus ou moins de suivi et de succès.

6.6.2. Qu'est-ce qu'un wiki sémantique ?

a. Des wikis aux wikis sémantiques :

Les wikis sont des sites Web permettant la création et l'édition collaborative de contenus de manière simple [Leuf et Cunningham, 2001]. Ils reposent généralement sur un ensemble de pages éditables, organisées en catégories et reliées par des liens hypertextes. Ils sont devenus le symbole de l'interactivité promue à travers le Web 2.0. L'un des principes fondateurs des wikis, qui constitue également le principal vecteur de leur popularité, est leur simplicité d'utilisation.

Les wikis sont créés et maintenus grâce à des systèmes spécifiques de gestion de contenus, les moteurs de wiki. De nombreux langages, appelés les wikitexts, ont vu le jour afin de permettre la structuration, la mise en pages et les liens entre les articles. Chaque système dispose généralement de son propre wikitext, comme l'illustre l'exemple de la figure suivante :

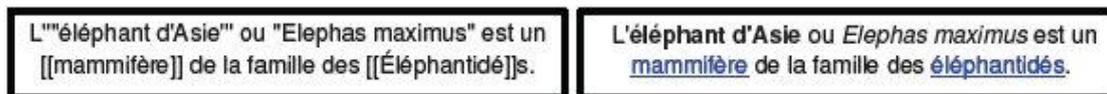


Figure 25 : Exemple du wikitext de MediaWiki (à gauche) et son affichage (à droite).

À ce point d'avancement, une idée a émergé : celle de pouvoir exploiter automatiquement les connaissances ainsi stockées. En effet, une limite de l'exploitation des wikis s'illustre par le questionnement des données contenues dans ces pages. La recherche se fait généralement grâce à la reconnaissance des mots par chaînes de caractères, sans prendre en compte leurs significations. Par exemple, le système ne peut pas répondre à une requête du type : « Donner la liste de tous les rois régnant actuellement ». La solution utilisée dans Wikipédia consiste en une génération manuelle de listes. On ne peut certes pas imaginer toutes les listes possibles correspondant aux requêtes qui seront posées par les utilisateurs. Pour ce besoin, l'ajout d'une couche sémantique s'est avérée nécessaire.

Les wikis sémantiques sont nés du rapprochement des wikis et du Web sémantique. [Berners-Lee et Fischetti, 1999] les définit comme des wikis améliorés par l'utilisation des technologies du Web sémantique. Plus particulièrement, un wiki sémantique est similaire au wiki traditionnel dans le sens où c'est un site Web dans lequel le contenu est ajouté par les utilisateurs. Ce contenu est organisé en pages éditables et indexables, accessibles à tous les utilisateurs. Cependant, contrairement au wiki traditionnel, le wiki sémantique

ne se limite pas au texte en langage naturel. Il caractérise les ressources et les liens entre celles-ci comme l'illustre la figure suivante :

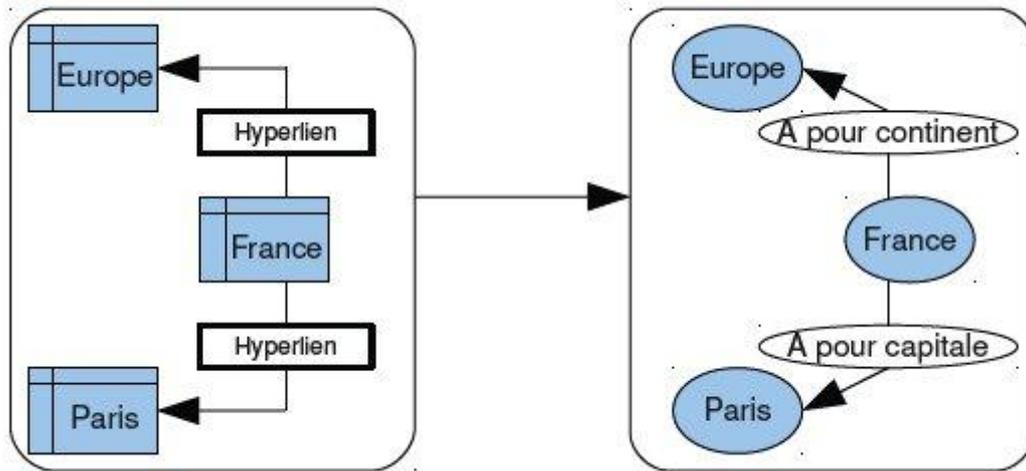


Figure 26 : Différences de représentation entre un wiki (à gauche) et un wiki sémantique (à droite).

Ces informations sont formalisées et deviennent donc exploitables par une machine, à travers des processus de raisonnements artificiels.

b. Deux approches :

Comme mentionné dans [Buffa et al., 2007] on peut distinguer deux approches dans la conception des moteurs. La première, appelée wikis for ontologies ou Wikitology, concerne le plus grand nombre des moteurs. Cette approche considère les pages comme des concepts et les liens typés comme des propriétés. Il se dessine ainsi une ontologie formalisée, dont la précision est le plus souvent renforcée par la catégorisation des concepts. Dans ce cas, la structure des données est généralement très souple et permet une grande liberté pour l'utilisateur mais ne garantit pas l'utilisabilité des ontologies résultantes. En effet, elle suppose que l'utilisateur se rappelle au moment de la saisie de toutes les propriétés et de tous les concepts existants afin de ne pas créer de doublons. Par exemple, rien n'empêche un utilisateur de créer deux relations « a pour roi » et « a pour monarque » ayant la même sémantique. Cela alourdit la base de connaissances et nuit à son homogénéité.

La deuxième approche, ontologies for wikis, suppose généralement une ontologie préexistante. Selon cette approche, le but du moteur de wiki consiste à fournir les outils permettant son peuplement par l'ajout d'instances et parfois de classes. Ces outils sont généralement des formulaires de choix multiples ou utilisant l'auto-complétion. S'ils restreignent la liberté des utilisateurs, en refusant par exemple la création de nouveaux types de lien, ils garantissent la cohérence de l'ontologie finale. Cette approche apparente le wiki à un éditeur de métadonnées, permettant de peupler l'ontologie. Ainsi, les moteurs de wiki de

cette catégorie sont le plus souvent destinés à des domaines spécifiques, plus facilement formalisables.

6.6.3. Exemples de moteurs wiki sémantique :

IkeWiki :

IkeWiki [Schaffert, 2006] est souvent pris comme exemple de l'approche ontologies for wikis. Il se présente comme un outil de gestion collaborative des connaissances. Il nécessite une ontologie pré-existante. La gestion des connaissances se fait grâce au framework Jena, qui sert de RDFstore et de moteur SPARQL. Le système propose une granularité avancée dans la gestion des utilisateurs, afin qu'il soit possible de confier à des experts des tâches plus complexes que celles confiées à des novices. Une autre fonction intéressante est l'import de pages depuis Wikipédia, dont Ikewiki gère les annotations postérieures. Son interface utilise les technologies Ajax, qui permettent une grande interaction avec l'utilisateur. Deux éditeurs sont proposés : le premier pour l'édition des métadonnées qui comporte un mécanisme d'auto-complétion, et le second WYSIWYG pour gérer le contenu. Récemment, le projet s'est agrandi et est devenu le projet KiWI (Knowledge In Wiki) [Schaffert et al., 2008]. Son but est désormais de mettre à l'épreuve cette vision des wikis dans le contexte industriel, comme dans la gestion de projet informatique (gestion des équipes, des compétences, des emplois du temps, etc.).

Semantic Mediawiki :

Semantic Mediawiki [Völkel et al., 2006] ,à l'opposé de **IkeWiki** , il est très représentatif de l'approche wikis for ontologies, et est celui qui se rapproche le plus de la philosophie Wiki. C'est une extension de MediaWiki, le moteur utilisé par Wikipédia. Afin de conserver une syntaxe simple, il intègre l'édition des triplets RDF dans son wikitext comme l'illustre la figure suivante :

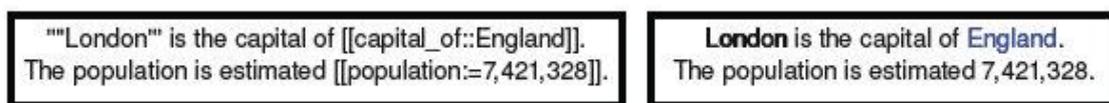


Figure 27 : Wikitext dans Semantic Mediawiki (à gauche) et son résultat (à droite).

Il permet ainsi aisément la création de liens typés qui servent également à indiquer les attributs de la page. Un autre intérêt de Semantic MediaWiki est la forte communauté de développeurs qui l'entoure et produit de nombreuses extensions, telles que les formulaires de saisie, l'ajout d'un moteur d'inférences, etc.

En particulier, l'extension Halo [<http://www.projecthalo.com/>] propose des formulaires, de l'auto-complétion, un éditeur WYSIWYG, l'intégration de fichiers audio ou vidéos mais aussi l'intégration d'un point d'accès SPARQL.

Cette communauté fait de Semantic MediaWiki un des projets les plus actifs et le mieux documenté.

Un dérivé particulièrement intéressant de Semantic MediaWiki se nomme **SWOOKI** [Rahhal et al., 2009]. Il propose de résoudre les problèmes des wikis liés à la centralisation des données en intégrant une architecture pair-à-pair et un algorithme de gestion des modifications concurrentes.

SMW+ [<http://wiki.ontoprise.de>] est une version commerciale et augmentée de Semantic Mediawiki. Il intègre de nombreuses extensions, faisant de SMW+ un système particulièrement complet. Il est à noter que toutes les extensions utilisées sont sous licence GPL et facilement intégrables à Semantic Mediawiki.

BOWiki :

BOWiki [Bacher et al., 2008] est un wiki sémantique spécialisé, son objectif est de proposer un outil pour la construction collaborative d'ontologies biomédicales. Lors de l'ajout d'une unité de connaissances, il contrôle la consistance de l'ontologie grâce à un moteur d'inférences et rejette cette unité de connaissances le cas échéant. Le moteur d'inférences est également utilisé pour répondre à des requêtes complexes. L'idée est de contrôler la connaissance avant de la placer dans l'ontologie, c'est-à-dire de compléter le travail de l'expert par une vérification de la consistance.

Knoodl et Wikidmart :

Knoodl et Wikidmart sont des exemples de systèmes commerciaux qui sont apparus dans la famille des wikis sémantiques.

Knoodl se présente comme un outil de développement communautaire d'ontologies OWL et de bases de connaissances. Il dispose de fonctionnalités d'import et d'export d'ontologies, propose la gestion de requêtes avancées et un point d'accès SPARQL. L'idée de communautés est mise en avant par un système dans lequel chacune d'entre elles dispose de son propre wiki, intégrant son vocabulaire spécifique.

Wikidmart entre dans le cadre d'une application Web sémantique plus lourde dont le but est de proposer des services pour la gestion commerciale et humaine des entreprises. Il propose le même type de solution que son concurrent. Dans les deux cas, il semble que les fonctions les plus intéressantes soient payantes, et les sociétés restent discrètes sur les moyens mis en œuvre.

6.6.4. Apports et emploi des technologies du web sémantique :

La figure suivante permet de juger de l'emploi des technologies du Web sémantique, en s'appuyant en premier lieu sur le langage de représentation des connaissances utilisé. Nous cherchons à savoir ensuite dans quelle mesure l'import et l'export d'ontologies sont supportés.

Enfin, nous nous intéressons aux méthodes de raisonnements et aux possibilités pour les requêtes internes (type de requête accepté) et externe (disponibilité d'un point d'accès SPARQL).

Remarque :

La première ligne de la figure concerne l'approche conceptuelle où nous cherchons à savoir si les concepteurs se fondent sur une approche wikis for ontologies ou une approche ontologies for wikis.

| | | AceWiki | BoWiki | KiWi | Knoodl | KnowWE | OntoWiki |
|--------------------------|---------------------------|--------------------------|--------------------------|----------------------------------|----------------------------------|----------------|----------------------------------|
| Approche | Wiki for ontology | • | | | | | • |
| | Ontology for wiki | | • | • | • | • | |
| Outils du Web sémantique | Langage de représentation | OWL, SWRL | OWL | RDF, OWL | NC | OWL, d3web XML | RDF(S), OWL |
| | Export d'ontologie | | • | | • | • | • |
| | Import d'ontologie | Format OBO | | | • | Format d3web | |
| | Système de requêtes | Texte, requêtes avancées | Texte, requêtes avancées | Texte, Sparql, requêtes avancées | Texte, Sparql, requêtes avancées | Texte, Sparql | Texte, Sparql, requêtes avancées |
| | Accès sparql | | | • | • | | • |
| | Moteur d'inférences | Pellet | Pellet | Jena | | Sesame | pOWL |
| | Stockage des données | Text | DB | RDFStore + DB | RDFstore | | RDFstore |

| Semantic MediaWiki | SMW+ | Subleme | SWIM | TaOPis | Wikidmart |
|---|---|---------------|---|--|---|
| • | • | | | • | • |
| | | • | • | | |
| RDF(S), OWL | RDFS, OWL | RDF | RDF(S), OWL | F-Logic | Inconnu |
| • | • | | • | • | |
| Format OMDoc | | | | | |
| Texte, Sparql, requêtes avancées | Texte, Sparql, requêtes avancées | Texte | Texte, Sparql, requêtes avancées | Texte, requêtes Flora-2, requêtes avancées | Texte, Sparql, requêtes avancées |
| | • | | | | • |
| | KAON 2 | | | Flora-2 | • |
| DB | DB | Pages HTML | DB | DB | DB |

Figure 28 : utilisation des technologies du WS dans les moteurs wiki sémantique [T. Meilender et al. 2010]

6.7. Conclusion :

Nous avons résumé dans cette section quelques classes d'applications archétypes de l'usage immédiat et tangible des technologies du web sémantique. Il ne fait aucun doute que cette technologie du sens doit apporter un saut qualitatif indiscutable si ce n'est une réelle rupture technologique.

Toutefois pour réaliser cette vision du Web sémantique, les recherches actuelles qui s'appuient sur un existant riche (venant des domaines de l'ingénierie des connaissances, des bases de données, de l'apprentissage automatique, des hypermédias ou encore des interfaces homme-machine) doivent maintenant résoudre les problèmes posés par :

- Le passage à l'échelle du web,
- La tenue en contexte de forte hétérogénéité (modélisations et langages),
- La tenue en milieu fortement évolutif.

VI

Conclusion Générale

IV. Conclusion et perspectives :

Nous avons présenté dans ce rapport ce que tout le monde définit comme étant le Web du futur, qui est nommé le *Web Sémantique*. Nous avons expliqué ces principes de base et les changements qu'il apporte à notre vision du Web vers un Web de données (dont les documents ne sont qu'un type parmi d'autres) à fort niveau de connexion. Nous avons cherché à faire comprendre la nature des représentations sémantiques, le rôle majeur des métadonnées/annotations et le rôle central des ontologies dans un contexte de recherche d'interopérabilité généralisée. Nous avons vu aussi les différentes technologies proposées dans le cadre du Web sémantique dont les langages, certes centraux, ne sont pourtant qu'une partie.

Ensuite, bien qu'il soit impossible de rendre compte de l'ensemble des applications qui utilisent les technologies du WS qui diffusent dans des contextes différents (Web ouvert mais aussi Web d'entreprise), nous avons présenté différentes réalisations et la en va conclure par les différentes perspectives de recherche pour le web sémantique qui sont axées sur :

- L'indexation et évaluation des ontologies,
- La création, documentation et maintenance des ontologies,
- l'interopérabilité entre ontologies (ce qui sera fort utile si l'on n'arrive pas à se mettre d'accord sur un langage universel de représentation d'ontologies) ou sur l'intégration de multiples ontologies,
- L'automatisation (semi) du processus d'annotation et de construction des ontologies,
- Les langages de requêtes pour le Web sémantique,
- La sécurité, confiance et provenance,
- Le développement des agents intelligents,
- Le développement d'interfaces utilisateurs adaptées au Web sémantique.

Bibliographie

❧ Références bibliographiques ❧

[A. Napoli et al, 2004] A. Napoli, B. Carre, R. Ducournau, J. Euzenat, and F. Rechenmann. Objets et représentation, un couple en devenir. *RSTI-L'objet*, pages 61–81.

[ARPIREZ J., CORCHO O., FERNÁNDEZ-LÓPEZ M. & GÓMEZ-PÉREZ A. 2001]. WebODE : a Workbench for Ontological Engineering. In *First international Conference on Knowledge Capture (K-CAP'01)*, p. 6–13, Victoria, Canada: ACM.

[Bacher et al., 2008] Bacher, J., R. Hoehndorf, et J. Kelso (2008). Bowiki : Ontology-based semantic wiki with abox reasoning. In 3rd Semantic Wiki Workshop.

[BACHIMONT B., ISAAC A. & TRONCY R. 2002]. Semantic Commitment for Designing Ontologies: A Proposal. In A. GOMEZ-PÉREZ & V. BENJAMINS, Eds., *13th International Conference on Knowledge Engineering and Knowledge Management (EKAW'02)*, volume (2473) of *Lecture Notes in Artificial Intelligence*, p. 114–121, Sigüenza, Espagne: Springer Verlag.

[B. Motik et al, 2007]. B. Motik, Peter F. Patel-Schneider, and I. Horrocks. Owl 1.1 web ontology language structural specification and functional-style syntax, May 2007.

[B. Motik et al, 2007]. B. Motik, Peter F. Patel-Schneider, and I. Horrocks. Owl 1.1 web ontology language structural specification and functional-style syntax, May 2007.

[B. Motik et al, 2009]. B. Motik, P. F. Patel-Schneider, B. Parsia, C. Bock, A. Fokoue, P. Haase, R. Hoekstra, I. Horrocks, A. Ruttenberg, U. Sattler, and M. Smith. Owl 2 web ontology language structural specification and functional-style syntax.

[B. C. Grau et al, 2008] B. C. Grau, I. Horrocks, B. Motik, B. Parsia, P. Patel-Schneider, and U. Sattler. Owl2 : The next step for owl. Elsevier, 2008.

[BECHHOFFER S., HORROCKS I., GOBLE C. & STEVENS R. 2001]. OilEd: a Reason-able Ontology Editor for the Semantic Web. In *Joint German/Austrian conference on Artificial Intelligence (KI'01)*, volume (2174) of *Lecture Notes in Artificial Intelligence*, p. 396–408, Vienne,

Bibliographie

Autriche: Springer Verlag.

[Bechhofer et al, 2004]. Sean Bechhofer, Frank van Harmelen, Jim Hendler, Ian Horrocks, Deborah L. McGuinness, Peter F. Patel-Schneider, et Lynn Andrea Stein. "OWL Web Ontology Language reference". W3C Recommendation, 10 February 2004. Available at <http://www.w3.org/TR/owl-ref/>

[Benjamins et al, 1999] V.R. Benjamins, D. Fenel, S. Decker et A. GÓMEZ-PÉREZ, (KA) 2: "building ontologies for the internet: a mid term report", in International Journal of Human Computer Studies, Volume 51, pp.687-712.

[Berners-Lee, 1989]. Tim Berners-Lee (1989). Information Management : A Proposal. Rapport technique, CERN.
<http://www.w3.org/History/1989/proposal.html>.

[Berners-Lee et Fischetti, 1999]. Berners-Lee, T. et M. Fischetti (1999). Weaving the Web : The Original Design and Ultimate Destiny of the World Wide Web by Its Inventor (1st ed.). Harper San Francisco.

[Bernstein, 2001] Philip Bernstein, et Erhard Rahm, *A survey of approaches to automatic schema matching*, The VLDB Journal, Volume 10 , pages 334-350, 2001.

[Blazquez , 1998] M. Blazquez, M. Fernandez, J. Garcia-Pinar. et A. Gomez-Perez, "Building Ontologies at the Knowledge Level using the Ontology Design Environment", in Proceedings of the Banff Workshop on Knowledge Acquisition for Knowledge-based System.

[Borst, 1997] Borst W. N. (1997). "Construction of Engineering Ontologies for knowledge sharing and reuse". CTIT Ph. D-thesis series No. 97-14 .Center for Telematica and Information.

[BOURIGAULT D. & LAME G. 2002]. Analyse distributionnelle et structuration de terminologie. Application à la construction d'une ontologie documentaire du droit. *Traitement automatique des langues*, **43**(1).

[Box et al, 2000]. D. Box, D. Ehnebuske, G. Kakivaya, A. Layman, N. Mendelsohn, H. Nielsen, S. Thatte, and D. Winer, "Simple Object Access Protocol (SOAP) 1.1",

Bibliographie

[Brachman et Levesque, 1984]. Brachman, R.J. et Levesque, H.J. “The tractability of subsumption in frame-based description languages”. Proceedings of AAAI-84. Pages.34-37. (1984).

[Bruijn et al, 2008] Jos de Bruijn, Dieter Fensel, Mick Kerrigan, Uwe Keller, Holger Lausen, et James Scicluna, *Modeling Semantic Web Services, The Web Service Modeling Language*, édition Springer-Verlag Berlin Heidelberg.

[Burstein et al., 2004] M. Burstein, J. Hobbs, Ora Lassila, D. Martin, D. McDermott, S. McIlraith, S. Narayanan, M. Paolucci B. Parsia, T. Payne, E. Sirin, N. Srinivasan, and K. Sycara, *OWL-S : Semantic markup for Web services*, pages 63-100, 2004.

[Buffa et al., 2007]. Buffa, M., G. Erétéo, et F. L. Gandon (2007). Wiki et web sémantique. In F. Trichet (Ed.), IC 2007 : 18e Journées Francophones d’Ingénierie des connaissances 2007, Grenoble, France, July 4-6, 2007, pp. 49–61. Cepadues.

[Camille Roux, 2008] www.lpmagazine.org

[Casati et al., 2000] F. Casati, S. Ilnicki, L. Jin, V. Krishnamoorthy, et M.-C Shan, *Adaptive and dynamic service composition in E-Flow*, In Proceeding of the 12th Conference on Advanced Information Systems Engineering (CAISE), S. Verlag, pages 13-31, Stockholm, Sweden.

[Cerami 2002] Ethan Cerami, *Web Services Essentials*, édition O’Reilly, Février 2002.

[Chappell 2002] David Chappell, et Tyler JEWELL, *Java Web Service*, édition O’Reilly, Mars 2002.

[CHARLET J. 2002]. *L’Ingénierie des Connaissances : développements, résultats et perspectives pour la gestion des connaissances médicales*. Mémoire d’habilitation à diriger des recherches en Informatique de l’université de Pierre et Marie Curie.

[Chauvet, J.M., 2002] Chauvet, J.M. (2002). *SW avec SOAP, WSDL, UDDI, ebXML...* Paris: Eyrolles, pp.47-52.

[Cimiano et al, 2004] P. Cimiano, S. Handschuh, et S. Staab, “Towards the Self-Annotating Web”, in Proceedings of the 13th International World Wide Web Conference (WWW’04), ACM Press, New-York, USA, pp. 462-471.

Bibliographie

- [Corby et al., 2004] Corby, O., Dieng-Kuntz, R. et Faron-Zucker, C. (2004). Querying the Semantic Web with the CORESE search engine. In R. Lopez de Mantaras and L. Saitta eds, Proc. of the 16th European Conference on Artificial Intelligence (ECAI'2004), Valencia, 22-27 August 2004, IOS Press, p.705-709.
- [Corby et al., 2004] Olivier Corby, Rose Dieng-Kuntz et Catherine Faron-Zucker (2004). Querying the Semantic Web with Corese Search Engine. pages 705–709. IOS Press.
- [Corby et al, 2006] Corby, O., Dieng-Kuntz, R., Faron-Zucker, C. et Gandon, F. (2006). Searching the Semantic Web: Approximate Query Processing based on Ontologies IEEE Intelligent Systems Journal, January/February 2006 (Vol. 21, No. 1).
- [Corby et Faron, 2007] Corby, O. et Faron, C. (2007). Implementation of SPARQL Query Language based on Graph Homomorphism. In Proc. of the 15th International Conference on Conceptual Structures (ICCS'2007), Sheffield, UK.
- [Crubezy-Met al., 2003] Crubezy-M. Fensel D. Benjamins R. Wielinga B. Motta E. Musen M. Omelayenko, B. Upml : The language and tool support for making the semantic web alive. Technical report, MIT, 2003.
- [Dallons, 2004] Gautier Dallons, *DAML-S : interactions, critique et évaluation*, Institut d'informatique des FUNDP, Namur, Belgique.
- [DAML Services Coalition , 2002]. DAML-S: Web Service Description for the Semantic Web. In *The First International Semantic Web Conference (ISWC)*, pages 348–363.
- [Day 2005] M. DAY, Installment on « metadata », Digital Curation Manual, version 1.1, Disponible sur : [http://www.dcc.ac.uk/resource/curation manual/chapters/metadata/metadata.pdf](http://www.dcc.ac.uk/resource/curation_manual/chapters/metadata/metadata.pdf), novembre 2005
- [Dieng et al., 2004] Dieng, R., Corby, O., Gandon, F. et Golebiowska, J. Ontologies pour un Web sémantique d'entreprise. Chapitre 1 de Gestion Dynamique des Connaissances Industrielles, Benoît Meynard, Muriel Lombard, Nada Matta, Jean Renaud, édés, Hermès.
- [Djaghoul Younes.2007] « Intégration des ressources Web dans un environnement P2P, basée sur les ontologies et la gestion de la confiance »,

Bibliographie

Université Mentouri Constantine, thèse Présentée pour obtenir le grade de doctorat en sciences en informatique.

[Enrico Motta et al., 2003] Enrico Motta, John Domingue, Liliana Cabral, and Mauro Gaspari. Irs-2 : A framework and infrastructure for semantic web services. In *International Semantic Web Conference*, pages 306–318.

[Eric Newcomer, 2002]. Eric Newcomer. *Understanding Web Services- XML, WSDL, SOAP and UDDI*, chapter 3, Describing Web Services : WSDL. Addison Wesley Professional, May 2002.

[Euzenat1996]. Jérôme Euzenat, Corporate memory through cooperative creation of knowledge bases and hyper-documents, Proc. 10th workshop on knowledge acquisition (KAW), Banff (CA), pp(36)1-18, 1996

[<http://www.archivesdefrance.culture.gouv.fr/thesaurus/historique-web-semantique.html>]

[Euzenat, 2005] J. Euzenat, et P. Shvaiko, *A survey of schema-based matching approaches*, Journal on Data Semantics IV, pages 146-171, 2005.

[Fabrice Molinaro 2004] <http://www.les-infostrategies.com/tag/world-wide-web>

[Francis Lapique, 2006] Francis Lapique:Langages de requêtes pour base de connaissances.

[Froidevaux C., Cohen Boulakia S., 2002] Intégration de Sources de Données Génomiques du Web, aux « Journées scientifiques du Web Sémantique », Paris

[Furst, 2002] F. Furst, “L’ingénierie ontologique”. Rapport de recherche N°02-07.

[Gandon, 2002] F.Gandon. Distributed Artificial Intelligence and Knowledge Management: Ontologies and Multi-Agent System for a Corporate Semantic Web. PhD Thesis at INRIA Sophia Antipolis, ACACIA team.

[Gardien, 2002] Georges Gardien, *XML des bases de données aux Services Web*, édition Dunod.

[Garey et Johnson, 1979] Michael R. Garey et David S. Johnson (1979). *Computers and Intractability – A Guide to the Theory of NP-Completeness*. W. H. Freeman And Company.

Bibliographie

- [Golebiowska et al., 2001] Golebiowska, J., Dieng-Kuntz, R., Corby, O. et Mousseau, D. (2001). Building and Exploiting Ontologies for an Automobile Project Memory. First International Conference on Knowledge Capture (K-CAP), Victoria, October 23–24.
- [Gomez-Perez, 1999], Gomèz-Pérez A. “Ontological Engineering: A state of the art”. Expert Update, Volume 2 (3), Pages 33-43. 1999.
- [Gruber, 1993] Tomas Gruber. “A Translation Approach to Portable Ontology Specifications”. Knowledge Acquisition, Volume 5, No 2. Pages 199-220. 1993.
- [Guarino, 1997] Guarino N. “Some organizing principles for a unified top-level ontology”. AAAI Spring Symposium on Ontological Engineering, 57-63.
- [Haarslev, 2001] V. Haarslev, R. Möller, “ Racer user’s guide and reference manual version 1.6”. Technical report, University of Hamburg, Computer Science Department.
- [Handschuh, 2001] S. Handschuh, “Ontomat. ”. <http://km.aifb.uni-karlsruhe.de/annotation/index.html>.
- [Handschuh et al 2002] S. Handschuh, S. Staab et F. Ciravegna, “S-cream-semi-automatic creation of metadata”. Proc. of the European Conference on Knowledge Acquisition and Management. LNCS 2473, Springer-Verlag, Madrid, Espagne, pp. 379-387.
- [HANDSCHUH S. 2005] M. HANDSCHUH S, Creating Ontology-based Metadata by Annotation for the Semantic Web, Thèse de doctorat, Université de Karlsruhe, 2005, 225 p.
- [Heflin, 2001] J. Heflin et J. A. Hendler, “A Portrait of the Semantic Web in Action”, in IEEE Intelligent Systems, 16(2), IEEE, pp. 54-59.
- [Heflin, 2002] J. Heflin, J. Hendler, S. Luke, C. Gasarch, Q. Zhendong, L. Spector et D. Rager, “The shoe knowledge annotator”. (2002). <http://www.cs.umd.edu/projects/plus/SHOE/KnowledgeAnnotator.html>.
IMARKUP SOLUTIONS. Imarkup. <http://www.imarkup.com/>.
- [Hollunder et al, 1990]. Bernhard Hollunder, Werner Nutt, and Manfred Schmidt-Schau. Subsumption algorithms for concept description languages. Proc. of the 9th Eur. Conf. on Artificial Intelligence (ECAI'90), pages 348--353, 1990.

Bibliographie

[Horrocks,1997]. Ian Horrocks. “Optimising Tableaux Decision Procedures for Description Logics”. PhD thesis, University of Manchester, 1997.

[Horrocks et al, 1999a]. I. Horrocks, U. Sattler, and S. Tobies. “ Practical reasoning for expressive description logics”. H. Ganzinger, D. McAllester, and A. Voronkov, editors, Proc. of the 6th Int. Conf. on Logic for Programming and Automated Reasoning (LPAR'99), number 1705 in Lecture Notes in Artificial Intelligence, Pages 161-180. Springer , 1999.

[Horrocks, et al, 2000a]. Ian Horrocks, Ulrike Sattler, and Stephan Tobies. “ Practical reasoning for very expressive description logics”. J. of the Interest Group in Pure and Applied Logic, Volume 8(3):Pages 239-264, 2000.

[Horrocks, et al, 2000b]. Ian Horrocks, Ulrike Sattler, and Stephan Tobies. “ Reasoning with individuals for the description logic SHIQ”. David McAllester, editor, Proc. of the 17th Int. Conf. on Automated Deduction (CADE 2000), volume 1831 of Lecture Notes in Computer Science, pages 482-496. Springer, 2000.

[Horrocks et al, 2003]. Ian Horrocks, Peter F. Patel-Schneider, and Frank van Harmelen. “From SHIQ and RDF to OWL: The making of a web ontology language”. J. of Web Semantics, Volume 1 (1):page 7-36, 2003.

[Horrocks et al, 2005]. Ian Horrocks, Oliver Kutz, and Ulrike Sattler. “The irresistible SRIQ”. In Proc. of the First OWL Experiences and Directions Workshop, 2005.

[Horrocks et al, 2006]. Ian Horrocks, Oliver Kutz, and Ulrike Sattler. “The even more irresistible SROIQ”. In Proc. of the 10th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR 2006), Pages 57-67. AAAI Press, 2006.

[Horrocks et Patel-Schneider, 2003]. Ian Horrocks and Peter F. Patel-Schneider. “Reducing OWL entailment to description logic satisfiability”. In Dieter Fensel, Djaghloul Younes. Université Mentouri Constantine. 2007 130 Katia Sycara, and John Mylopoulos, editors, Proc. of the 2003 International Semantic Web Conference (ISWC 2003), number 2870 in Lecture Notes in Computer Science, Pages 17-29. Springer, 2003.

Bibliographie

[Horrocks et Sattler, 1999]. Ian Horrocks et Ulrike Sattler. “A Description Logic with Transitive and Inverse roles and Role Hierarchies”. *J. of Logic and Computation*, Volume 9(3): Pages 385-410, 1999.

[Horrocks et Sattler, 2001]. Ian Horrocks and Ulrike Sattler. “Ontology reasoning in the SHOQ(D) description logic”. *Proc. of the 17th Int. Joint Conf. on Artificial Intelligence (IJCAI 2001)*, Pages 199-204. Morgan Kaufmann, Los Altos, 2001.

[Horrocks et Sattler, 2004]. Ian Horrocks and Ulrike Sattler. “Decidability of SHIQ with complex role inclusion axioms”. *Artificial Intelligence*, 160(1-2):79-104, 2004.

[Horrocks et Sattler, 2005]. Ian Horrocks and Ulrike Sattler. “A tableaux decision procedure for SHOIQ”. In *Proc. of the 19th Int. Joint Conf. on Artificial Intelligence (IJCAI 2005)*, Pages 448-453, 2005.

[Hyoil 2006] Han Hyoil, Song il-yeol, et Choi Namyoun, *A survey on ontology mapping*, SIG-MOD Rec, Volume 35, pages 34-41, September 2006.

[Inaba et al, 2000]. A. Inaba, Thepchai Supnithi, Mitsuru Ikeda, Riichiro Mizoguchi, Jun'ichi Toyoda. “An overview of Learning Goal Ontology”. *Proc. of ECAI2000 Workshop on Analysis and Modelling of Collaborative Learning Interactions*, Pages.23-30, Berlin, Germany, 2000.

[J.Miller et al, 2005] J.Miller M. Nagarajan M. Schmidt A. Sheth K. Verma R. Akkiraju, J. Farrell. *Web service semantics : Wsdl-s*. Technical report, IBM.

[J. ANGELE & Y. SURE , Eds. 2002]. *First International Workshop Evaluation of Ontology-based Tools (EON'02)* , volume (62) of *CEUR-WS*, Sigüenza, Espagne. <http://CEURWS.org/Vol-62/>.

[Jean Charlet, Bruno Bachimont, Raphaël Troncy.2003] « Ontologies pour le Web sémantique »** Mission de recherche STIM, AP-HP & INSERM ERM 202 jc@biomath.jussieu.fr **Institut National de l'Audiovisuel {bbachimont,Raphael.Troncy}@ina.fr **Université Technologique de Compiègne bruno.bachimont@utc.fr **INRIA Rhône-Alpes, Équipe EXMO raphael.troncy@inrialpes.fr

[Jim Hendler et al, 2004]. Jim Hendler Ian Horrocks Deborah L. McGuinness Peter F. Patel-Schneider Lynn Andrea Stein Sean Bechhofer, Frank van Harmelen. *Owl web ontology language*. Technical report, W3C, 2004.

Bibliographie

[Julien Plu, 2011]. Introduction au Web sémantique avril 2011
<http://jplu.developpez.com/>

[Kahan et al, 2001] J. Kahan, M.R. Koivunen, E. Prud'hommeaux et R. Swick, "Annotea: An Open RDF Infrastructure for Shared Web Annotations", in Proceedings of the 10th International World Wide Web Conference (WWW'01), ACM Press, Hong-Kong, pp. 623-632.

[Kiefer et al., 2007] Christoph Kiefer, Abraham Bernstein, Hong Joo Lee, Mark Klein et Markus Stocker (2007). Semantic Process Retrieval with iSPARQL. In Proceedings of the 4th European Semantic Web Conference (ESWC 2007), volume 4519 de Lecture Notes in Computer Science, pages 609–623. Springer.

[KLYNE Graham & CARROLL Jeremy, 2003]. Resource Description Framework (RDF): Concepts and Abstract Syntax. W3C Working Draft, <http://www.w3.org/TR/rdf-concepts/>

[Kochut et Janik, 2007] Krys Kochut et Maciej Janik (2007). SPARQLeR : Extended Sparql for Semantic Association Discovery. In Proceedings of the 4th European SemanticWeb Conference (ESWC 2007), volume 4519 de Lecture Notes in Computer Science, pages 145–159. Springer.

[Kraft et al., 2003] Tobias Kraft, Holger Schwarz, Ralf Rantzau et Bernhard Mitschang (2003). Coarse-Grained Optimization : Techniques for Rewriting SQL Statement Sequences. In Proceedings of the 29th international conference on Very large data bases, pages 488–499. Morgan Kaufmann.

[Lassila et Swick, 1999]. O. Lassila et R. Swick, Editors. 'Resource Description Framework (RDF) Model and Syntax Specification'. World Wide Web Consortium. 22 February 1999. <http://www.w3.org/TR/1999/REC-rdf-syntax-19990222>.

[Leuf et Cunningham, 2001]. Leuf, B. et W. Cunningham (2001). The Wiki way: quick collaboration on the Web. Addison- Wesley.

[Maedche et al, 2003]. Maedche A., Motik B., Stojanovic L., Studer R. et Volz R. "Ontologies for Enterprise Knowledge Management". IEEE Intelligent Systems, Volume 18 (2), Pages 26-33. 2003.

Bibliographie

[McCarthy, 1980]. John McCarthy. "Circumscription---A Form of Non-Monotonic Reasoning". *Artificial Intelligence Journal*. Volume 13, pages = 27-39. 1980.

[Melliti 2004] Tarek Melliti, Interopérabilité des Services Web complexes, Thèse de Doctorat, Université ;Paris IX Dauphine, le 8 Décembre 2004.

[Michard, 1999] A. Michard, "XML : langage et applications" Eyrolles ISBN 2-212- 09052-8.

[Michel Leblanc, 2003]. Michel Leblanc. Les web services et leur impact sur le commerce b2b. Master's thesis, CIRANO : Centre Interuniversitaire de Recherche en Analyse des Organisations, Août 2003.

[Minsky, 1975]. Marvin Minsky. "A framework for representing knowledge". P.H. Winston (Ed.), *The psychology of computer vision* (211-277). New York: McGraw-Hill.

[Morel-Pair 2001] C. MOREL-PAIR, « Panorama : des métadonnées pour les ressources électroniques »

Disponible sur :

http://hal.ccsd.cnrs.fr/docs/00/04/04/73/PDF/Metas_panorama_CMO.pdf, 2005.

[Morel-Pair 2007a] C. MOREL-PAIR, « Métadonnées, pour quoi faire ? », Disponible sur : http://artist.inist.fr/article.php3?id_article=384 ; mars 2007.

[Mougin F.,et al., 2003]Mougin F., Cuggia M., Le Beux P., Development of an indexing search engine for the UMVF : Proposal for an indexing method based on Dublin Core and XML, accepté à Medical Informatics Europe 2003.

[SALEMKOUR & KERBOUT, 2009] SALEMKOUR & KERBOUT, Etude des normes et standards en e-Learning et leurs comparaisons à un Méta-Modèle conçu pour un domaine d'enseignement, université de Tizi Ouzou ingénieur 2009.

[Mizoguchi et al, 1995]. Mizoguchi Riichiro, Vanwelkenhuysen Johan, Ikeda Mitsuru. "Task Ontology for Reuse of Problem Solving Knowledge". *Knowledge Building & Knowledge Sharing* (2nd International Conference on Very Large- Scale Knowledge Bases), Enschede, The Netherlands, Pages.46-59. 1995.

[Mizoguchi et al, 2000]. Riichiro Mizoguchi, Kouji Kozaki, Toshinobu Sano and Yoshinobu Kitamura. "Construction and Deployment of a Plant Ontology".

Bibliographie

Knowledge Engineering and Knowledge Management - Methods, Models and Tools, The 12th International Conference, EKAW2000, Lecture Notes in Artificial Intelligence 1937, Springer-Verlag, Pages.113-128, Juan-les-Pins, France, 2000.

[Neches et al, 1991] Robert Neches, Richard Fikes, Tim Finin, Thomas Gruber, Ramesh Patil, Ted Senator, and William R. Swartout. “Enabling Technology For Knowledge Sharing”. AI Magazine, Volume 12, No. 3, Pages 37-56. Fall 1991.

[Nelson, 1965]. Theodor H. Nelson (1965). Complex information processing: a file structure for the complex, the changing and the indeterminate. In Proceedings of the 1965 20th ACM national conference, pages 84–100. ACM Press.

[Newcomere, 2004] Eric Newcomere, *Understanding Web Services Xml WSDL SOAP And UDDI*, edition O'Reilly.

[Noy 2007] Natalya Noy, *Semantic integration : a survey of ontology-based approaches*, SIGMOD Rec, Volume 33, pages 65-70, 2007.

[NOY N., FERGERSON R. & MUSEN M. 2000]. The knowledge model of Protégé2000 : Combining interoperability and flexibility. In R. D IENG & O.CORBY, Eds., *12th International Conference on Knowledge Engineering and Knowledge Management (EKAW'00)*, volume (1937) of *Lecture Notes in Computer Science*, p. 17–32, Juan-les-Pins, France: Springer Verlag.

[P. F. Patel-Schneider and I. Horrocks, 2007]. Owl 1.1 web ontology language overview , 23 May 2007.

[Pan et Horrocks, 2003]. Jeff Pan and Ian Horrocks. “ Web ontology reasoning with datatype groups”. Dieter Fensel, Katia Sycara, and John Mylopoulos, editors, Proc. of the 2003 International Semantic Web Conference (ISWC 2003), number 2870 in *Lecture Notes in Computer Science*, Pages 47-63. Springer, 2003.

[Patrick Kellert et Farouk Toumani, 2003]. Laboratoire LIMOS - UMR (6158) du CNRS ISIMA - Campus des Cezeaux - B.P. 125 ; 63173 AUBIERE Cedex .Email : patrick.kellert@isima.fr et ftoumani@isima.fr

[Patrick Kellert et Farouk Toumani ; 2004] Patrick Kellert et Farouk Toumani « Les Web services sémantiques », Article 2004. Laboratoire LIMOS - UMR (6158) du CNRS ISIMA - Campus des Cezeaux - B.P. 125 63173 AUBIERE Cedex .

Bibliographie

[Patrick Kellert1 et Farouk Toumani, 2010] Les web services sémantiques. Patrick Kellert1 et Farouk Toumani2 [en ligne] [Accès 12 Mai 2010] www.isima.fr/limos/publi/RR-03-15.doc

[Pérez et al., 2006] Jorge Pérez, Marcelo Arenas et Claudio Gutierrez (2006). Semantics and Complexity of SPARQL. In Proceedings of the 5th International Semantic Web Conference (ISWC 2006), volume 4273 de Lecture Notes in Computer Science, pages 30–43. Springer.

[Ponge 2008] Julien Ponge. *Model Based Analysis of Time-aware Web Services Interactions*. Thèse de Doctorat de l'Université Blaise Pascal - Clermont-Ferrand II, dans le cadre de l'Ecole Doctorale des Sciences pour l'Ingénieur, France, 2008.

[Popov et al, 2003] B. Popov, A. Kiryakov, D. Manov, A. Kirilov, D. Ognyanoff et M. Goranov, “Towards Semantic Web Information Extraction”, in Proceedings of the Human Language Technologies Workshop (ISWC'03), Sanibel, Floride, pp. 1-22.

[PRIE Y. & GARLATTI S 2004]. M PRIE Y. & GARLATTI S., Méta-données et annotations dans le Web sémantique, in Le Web sémantique, CHARLET J., LAUBLET P. & REYNAUD C. (Ed.), Hors série de la Revue Information - Interaction - Intelligence (I3), 4(1), Cépaduès, Toulouse, 2004, pp. 45-68.

[Prud'hommeaux et Seaborne, 2008] Eric Prud'hommeaux et Andy Seaborne, éditeurs. SPARQL query language for RDF. W3C Recommendation 15 January 2008, World Wide Web Consortium. <http://www.w3.org/TR/rdf-sparql-query/>.

[Rahhal et al., 2009]. Rahhal, C., H. Skaf-Molli, et P. Molli (2009). Swooki : Un Wiki Sémantique sur réseau Pair-à-Pair. Ingénierie des Systèmes d'Information 14(1).

[RECTOR A. L. 1998]. Thesauri and formal classifications: Terminologies for people and machines. *Methods of Information in Medicine*, **37**(4–5), 501–509.

[ROUSSEY C, CALABRETTO S. & PINON J.-M. 2002]. Le thésaurus sémantique : contribution à l'ingénierie des connaissances documentaires. In B. BACHIMONT, Ed., *Actes des 6es Journées Ingénierie des Connaissances*, p. 209–20, Rouen, France.

[Schaffert, 2006]. Schaffert, S. (2006). Ikewiki : A semantic wiki for collaborative knowledge management. In 15th IEEE International Workshops

Bibliographie

on Enabling Technologies : Infrastructures for Collaborative Enterprises (WETICE 2006), 26-28 June 2006, Manchester, United Kingdom, pp. 388–396. IEEE Computer Society.

[Schaffert et al., 2008]. Schaffert, S., J. Eder, M. Samwald, et A. Blumauer (2008). Kiwi - knowledge in a wiki. In European Semantic Web Conference 2008.

[Schmidt-Schaubß et Smolka, 1991]. Manfred Schmidt-Schaubß, Gert Smolka. “Attributive concept descriptions with complements”. Artificial Intelligence Volume 48 , Issue 1. Pages 1-26. (1991).

[Seaborne, 2004] Andy Seaborne (2004). RDQL – A Query Language for RDF. W3c member submission 9 january 2004, World Wide Web Consortium. <http://www.w3.org/ Submission/2004/SUBM-RDQL-20040109/>.

[Sowa, 1996]. "Top-level ontological categories," International J. of Human-Computer Studies, Volume 43:5/6, Pages 669-686. 1996.

[Stocker et al., 2008] Markus Stocker, Christoph Kiefer Andy Seaborne, Abraham Bernstein et Dave Reynolds (2008). SPARQL Basic Graph Pattern Optimization Using Selectivity Estimation. In Proceedings of the 17th InternationalWorldWideWeb Conference (WWW 2008), pages 595–604.

[Studer et al, 1998]. Studer R., Benjamins R. et Fensel D. “Knowledge Engineering: Principles and Methods”. Data Knowledge Engineering. 1998, volume. 25, no1- 2, Pages. 161-197. 1998.

[SURE Y., ERDMANN M., ANGELE J., STAAB S., STUDER R. & WENKE D. 2002]. OntoEdit: Collaborative Ontology Engineering for the Semantic Web. In I. HORROCKS & J. HENDLER, Eds., *First International Semantic Web Conference (ISWC'02)*, volume (2342) of *Lecture Notes in Computer Science*,p. 221–235, Chia, Sardaigne, Italie: Springer Verlag.

[T. Meilender et al. 2010]. Thomas Meilender_;Nicolas Jay, Jean Lieber, Fabien Palomares. Les moteurs de wikis sémantiques : un état de l’art

[Tim Berners-Lee *et al.*, 2001]. Tim Berners-Lee, James Hendler, Ora Lassila The Semantic Web, Scientific american, May 2001 <http://www.sciencificamerican.com>

Bibliographie

[Uschold et Gruninger, 1996]. Uschold, M. et Gruninger, M, "Ontologies: principles, methods and applications", Knowledge Engineering Review, vol. 11, no. 2 1996

[VanHeijst et al, 1997]. Van Heijst G., Schreiber A. et Wielinga B. J. "Using Explicit Ontologies in KBS Development". International Journal of Human and Computer Studies /Knowledge Acquisition, Volume 46, No (2/3), Pages 183-292. 1997.

[Vargaz-vera et al, 2002 a] M. Vargaz-vera, E. Motta, J. Domingue, M. Lanzoni, A. Stutt et F. Ciravegna, "MnM: Ontology Driven Tool for Semantic Markup", in Proceedings of the Workshop on Semantic Authoring, Annotation et Knowledge Markup (SAAKM'02), Lyon, France.

[Vargaz-vera et al, 2002 b] M. Vargaz-vera, E. Motta, J. Domingue, M. Lanzoni, A. Stutt et F. Ciravegna, "MnM: Ontology Driven Semi-Automatic and Automatic Support for Semantic Markup", in Proceedings of the 13th International Conference on Knowledge Engineering and Management (EKAW'02), LNCS 2473, Springer-Verlag, Madrid, Espagne, , pp. 379-391.

[Völkel et al., 2006]. Völkel, M., M. Krötzsch, D. Vrandečić, H. Haller, et R. Studer (2006). Semantic wikipedia. In 15th international conference on World Wide Web, WWW 2006, Edinburgh, Scotland, May 23-26, 2006.

[West, 2000] Douglas B. West (2000). Introduction to Graph Theory (Second Edition). Prentice Hall.

[Wolff, 1729]. Wolff C. (1729). *Philosophia Prima sive Ontologia*.

[Yannick Prié¹, Serge Garlatti. 2003] Méta-données et annotations dans le Web sémantique. **LIRIS FRE 2672 CNRS : Université Claude Bernard Lyon 1
yprie@liris.univ-lyon1.fr ; Département IASC GET - ENST Bretagne
Serge.Garlatti@enst-bretagne.f

Annexe

A

***Les Services
Web***

Les services Web

I. Introduction :

Le web contient des vastes bases de données avec plusieurs buts et de multiples sources non homogènes. Ceci prouve qu'il est nécessaire d'améliorer l'accessibilité à cette importante masse d'informations, et de disposer d'outils plus sophistiqués pour une meilleure recherche et organisation au sein du web. Les web services fournissent une nouvelle manière de développer des applications conformes aux besoins de l'Internet, et ils semblent être la solution la plus adaptée pour assurer l'interopérabilité, qui permet de transmettre les données entre les différentes applications d'une organisation comme l'entreprise, la société ou l'individu ; ainsi la technologie des web services permet de réaliser le traitement de ces données, et gérer les liaisons entre les différentes applications.

II. Définition des services Web :

Le terme service Web est souvent utilisé de nos jours, mais pas toujours avec la même signification, car la signification est tributaire de l'application de cette technologie. On peut dire qu'un service Web est souvent vu comme une application accessible à d'autres applications sur le Web, mais il existe plusieurs définitions pour les services Web :

Définition 1 : Le consortium W3C définit un service Web comme étant : une application, ou un composant logiciel qui vérifie les propriétés suivantes : [Cerami 2002] :

- Il est identifié par un URI ;
- Ses interfaces et ses liens peuvent être décrits en XML;
- Sa définition peut être découverte par d'autres services Web ;
- Il peut interagir directement avec d'autres services Web à travers le langage XML en utilisant des protocoles Internet standards.

Définition 2 : « Un service Web est une application accessible à partir du Web. Il utilise les protocoles Internet pour communiquer, et utilise un langage standard pour décrire son interface. »[Melliti 2004].

Définition 3 : « Les services Web sont la nouvelle vague des applications Web. Ce sont des applications modulaires, auto-contenues et auto-descriptives qui peuvent être publiées, localisées et invoquées depuis le Web. Les services Web effectuent des actions allant de simples requêtes à des processus métiers complexes. Une fois qu'un service Web est déployé, d'autres applications (y compris des services Web) peuvent le découvrir et l'invoquer. » [Ponge 2008].

Exemple :

Quand nous voulons faire un voyage, il nous suffit d'interagir avec un seul SW ; il va arranger tout- ce dont nous avons besoin, comme la réservation de l'hôtel, l'achat de billet de compagnie aérienne, la location de voiture, etc. (cf. Figure 29).

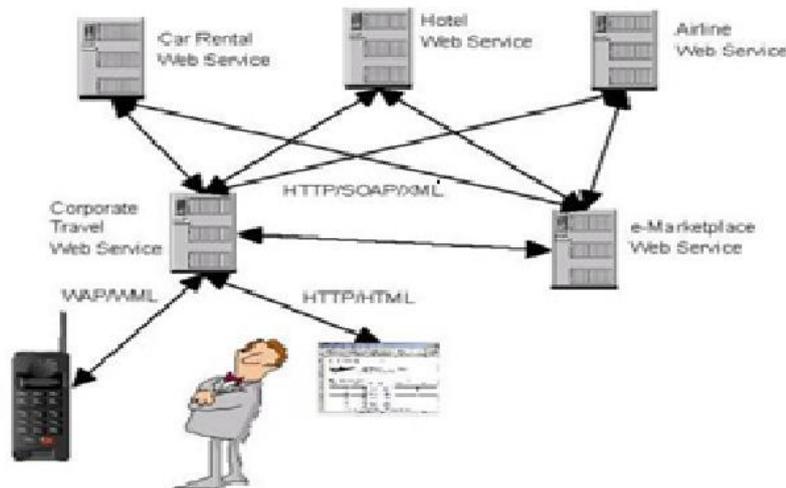


Figure 29 : Application de voyage de business qui interagit avec les SW

III. Caractéristiques des web services :

Selon [Cerami 2002], plusieurs acteurs définissent les web services par des caractéristiques technologiques distinctives, les plus importantes sont :

- **Un web service est une application logicielle qui est reconnue par un URI :** URI est la façon d'identifier un point de contenu sur le web comme un document tel qu'un texte, audio ou vidéo. L'URI la plus connue est l'adresse d'une page web, le web service est donc accessible en spécifiant son URI, c'est-à-dire que le web service est caractérisé par un seul objet et une seule fonctionnalité, c'est un prolongement de la programmation orientée objet et à partir de cela, on peut faire la construction d'une application logicielle très large comportant plusieurs fonctionnalités, afin de sélectionner les fonctionnalités qui sont recherchées par les URI spécifiques.
- **Capacité des interfaces et liaisons (bindings) d'être publiées, localisées et invoquées via le langage XML :** les principales tâches d'un web service sont : la publication dans un registre, la localisation en interrogeant ce registre qui l'héberge et l'invocation par un ou plusieurs web services après sa localisation. Ces tâches sont réalisées en utilisant le langage XML.

- **Capacité d'interagir avec les composantes des logiciels via des éléments XML avec l'utilisation des protocoles Internet standards :** un web service est créé pour être interrogé par d'autres logiciels contrairement à une page web, ou à une autre application qui n'utilise pas les web services. L'interopérabilité est basée sur l'utilisation du XML et des protocoles Internet standards, tels que, le HTTP qui est le protocole du web, le SMTP qui est le protocole du courrier électronique,... etc.
- **Composante logicielle légèrement couplée à interaction dynamique :** un web service avec un programme qui l'invoque est appelé le consommateur de web service, et qui sont indépendants l'un de l'autre. Si une modification est à faire sur le consommateur, on n'a pas besoin de connaître la machine, le langage de programmation, le système d'exploitation ou autres paramètres, afin d'établir à nouveau une communication entre le web service et son consommateur. Le consommateur possède une fonctionnalité qui consiste à faire une localisation et une invocation du web service, au moment de l'exécution du programme de web service, de manière automatique.

VI. Avantages des services Web

L'idée essentielle derrière les services Web est de partager les applications et les programmes en un ensemble d'éléments réutilisables appelés service, de sorte que, chacun de ces éléments effectuent une tâche principale et efficace, afin de faciliter l'interopérabilité entre tous ces services Web. D'autre part les services Web [Chappell 2002] :

- ✓ Permettent l'interopérabilité dans des environnements applicatifs, cela veut dire, que les logiciels et les applications écrits dans différents langages de programmation, et évoluant sur différents systèmes d'exploitation peuvent communiquer et/ou échanger des données entre eux facilement ;
- ✓ Permettent de profiter de différents environnements et langages de développement par une publication, localisation, description et une invocation via XML. Les services Web sont très flexibles, indépendants des langages de programmation et des systèmes d'exploitation ;
- ✓ Permettent d'accéder aux applications à travers les pare-feux à l'aide d'utilisation via le langage XML et les protocoles Internet standards comme HTTP sur le port 80, qui est généralement ouvert. Cela permet d'assurer une transmission des données transactionnelles et sécurisé ;
- ✓ Utilisables à distance via n'importe quel type de plateforme, et sont accessibles depuis n'importe quel type de clients ;

- ✓ Peuvent servir au développement d'applications distribuées. Ils appartiennent à des applications capables de collaborer entre elles de manière transparente pour l'utilisateur, et permettent d'avoir un partage des fonctionnalités et facilitent grandement le développement.

Pour résumer, les services Web ont plusieurs avantages, comme l'utilisation de standards universels, l'indépendance de plateforme, un environnement universel pour les systèmes d'information distribués, l'utilisation de plusieurs protocoles de transfert (par exemple HTTP, SMTP, FTP, . . . etc.), le codage des messages en utilisant le langage XML, un comportement compatible aux pare-feu, et la localisation par l'URI. Ils sont le résultat du développement exceptionnel des technologies de l'information, qui se sont entendus sur un certain nombre de protocoles et d'approches qui favoriseront l'interopérabilité entre les plateformes, les systèmes d'exploitation et les langages de programmation.

V. Architecture des services Web :

Deux types d'architecture existent pour les services Web : La première dite de référence, elle contient trois couches principales. La seconde architecture est plus complète, elle utilise les couches standards de la première architecture en ajoutant au-dessus d'autres couches plus spécifiques. Elle est appelé architecture étendue ou encore en Pile.

1. Architecture de Référence :

L'architecture de référence comporte les trois éléments suivants [Patrick Kellert et Farouk Toumani ; 2004] :

- **Le fournisseur de service** : c'est le propriétaire du service. D'un point de vue technique, il est constitué par la plateforme d'hébergement du service.
- **Le client** : c'est le demandeur de service. Techniquement, il est constitué par l'application qui va rechercher et invoquer un service. Une application cliente peut être elle-même un web service.
- **L'annuaire des services** : c'est un registre de descriptions de services offrant des facilités de publication de services pour les fournisseurs de services ainsi que des facilités de recherche de services pour les clients.

Ces trois éléments de l'architecture interagissent entre eux selon trois types d'opérations : les opérations de publication, de recherche et de liens.

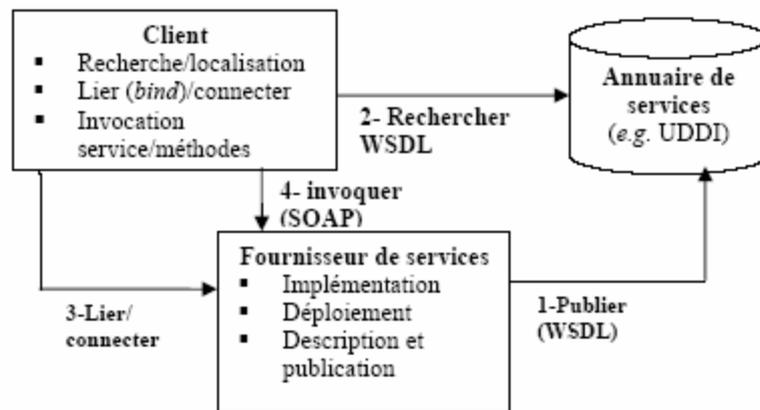


Figure 30 : Architecture des web services.

Le fournisseur de services définit la description de son service et la publie dans un annuaire de service UDDI qui peut être public ou privé. Le client utilise les facilités de recherche disponibles au niveau de l'annuaire pour retrouver et sélectionner un service donné. Il récupère ensuite les informations nécessaires sous format WSDL, à partir de la description du service sélectionné, lui permettant de se connecter au fournisseur du service et d'interagir avec l'implémentation du service considéré. La communication entre le demandeur de service et le fournisseur est assurée par le SOAP et les autres protocoles de communication. Le demandeur de service envoie une requête SOAP vers le fournisseur de service, cette requête est véhiculée par le HTTP jusqu'au fournisseur. Ensuite le web service du fournisseur de service renvoie sa réponse au demandeur sous la forme d'un document XML via SOAP et HTTP.

2. Architecture étendue :

Une architecture étendue est constituée de plusieurs couches se superposant les unes sur les autres, d'où le nom de pile des services Web. La pile est constituée de plusieurs couches, chaque couche s'appuyant sur un standard particulier. On retrouve, au-dessus de la couche de transport, les trois couches formant l'infrastructure de base décrite précédemment.

Nous apportons une explication de la mise en relief des trois types de couches :

- **L'infrastructure de base (Discovery, Discription, Exchange) :** Ce sont les fondements techniques établis par l'architecture de référence. Nous distinguons les échanges des messages établis par SOAP, la description de service par WSDL et la recherche de services Web que les organisations souhaitent utiliser via le registre UDDI.

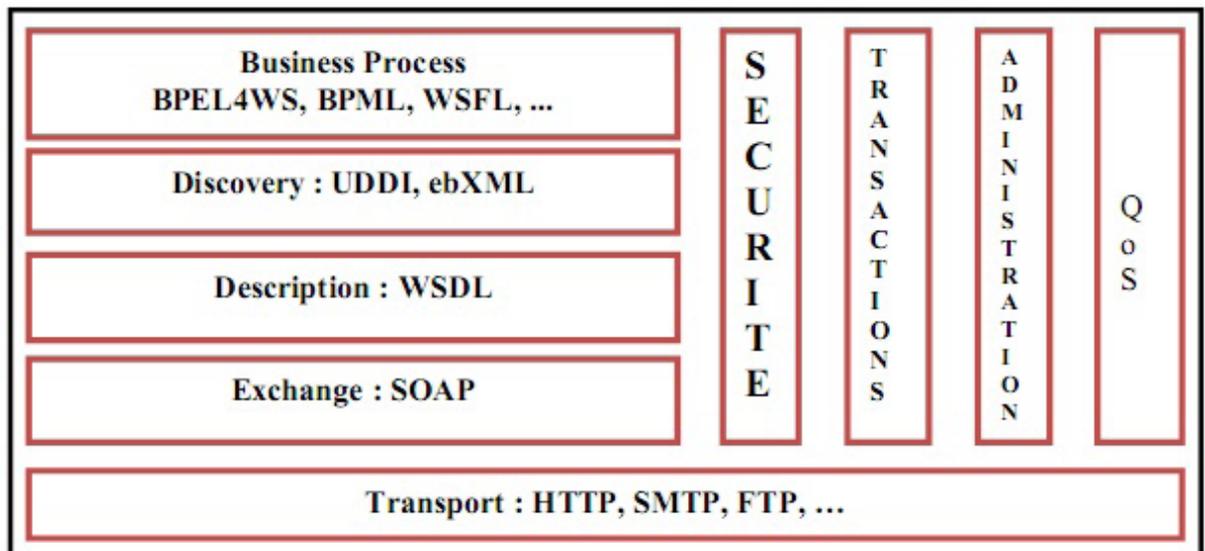


Figure 31 : Architecture en Pile des services Web.

- **Couches transversales (Security, Transactions, Administration, QoS)** : Ce sont ces couches qui rendent viable l'utilisation effective des services Web dans le monde industriel ;
- **La couche Business Processus (BusinessProcess)** : Cette couche supérieure permet l'intégration de services Web, elle établit la représentation d'un « **BusinessProcess** » comme un ensemble de service Web. De plus, la description de l'utilisation de différents services composant ce service est disponible par l'intermédiaire de cette couche.

VI. Standards de Service Web :

Plusieurs standards ont été proposés pour assurer l'interaction entre les trois opérations précédentes (publication, recherche et lien), entre autre nous citons les standards suivants :

1. SOAP (Simple Object Access Protocol) :

SOAP est une spécification de communication entre SW par échange de messages en XML à travers le Web. SOAP est simple et facile à implémenter dans les serveurs Web ou dans les serveurs d'applications. Elle est indépendante des langages de programmation ou des systèmes d'exploitation employés pour l'implémentation des SW. Les concepteurs de SOAP ont, en effet, réussi à préserver la plus grande généralité dans la représentation en XML des principes des protocoles de communication.

SOAP est défini comme un protocole léger d'échange de données dans un réseau de pair à pair, c'est-à-dire décentralisé. S'appuyant sur XML, SOAP propose un mécanisme simple de représentation des différents aspects d'un message entre applications. N'imposant aucun modèle de programmation spécifique, SOAP peut donc être utilisé dans tous les styles de communication : synchrone ou asynchrone, point à point ou multipoint, intranet ou Internet.

La spécification SOAP se divise en quatre parties [Chauvet, J.M., 2002]:

- L'enveloppe SOAP, qui définit le contexte d'un message, son destinataire, son contenu et différentes options ;
- Les règles de codage SOAP, définissant la représentation des données d'une application dans le corps d'un message SOAP (en particulier leur structure) ;
- Un protocole de RPC définissant la succession des requêtes et des réponses ;
- La définition de l'utilisation de HTTP comme couche de transport des messages SOAP.

Les règles de codage utilisent abondamment XML Schema pour décrire la structure des données constitutives des messages SOAP. Les Figure 32 et Figure 33 nous permettent de voir le message de requête et de réponse de SOAP.

```
POST /StockQuote HTTP/1.1
Host: www.stockquotesever.com
Content-Type: text/xml; charset="utf-8"
Content-Length: nnnn
SOAPAction: "Some-URI"

<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <m:GetLastTradePrice xmlns:m="Some-URI">
      <m:tickerSymbol>DIS</m:tickerSymbol>
    </m:GetLastTradePrice>
  </soapenv:Body>
</soapenv:Envelope>
```

Figure 32 : Exemple de Requête SOAP

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset="utf-8"
Content-Length: nnnn

<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <m:GetLastTradePriceResponse xmlns:m="Some-URI">
      <m:price>34.5</m:price>
    </m:GetLastTradePriceResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Figure 33 : Exemple de Réponse SOAP

2. WSDL (Web Services Description Language) :

Comme les protocoles de communication et de format de messages sont normalisés dans la communauté web, il devient de plus en plus possible et important d'être capable de décrire les communications d'une façon structurée.

WSDL répond à ce besoin en définissant une grammaire XML pour décrire des SW réseau comme les collections de point de terminaison de communication qui sont capable d'échanger des messages. Les définitions de SW WSDL fournissent de la documentation pour les systèmes distribués et servent comme une recette pour automatiser les détails concernant avec les applications de communication. Un document WSDL définit les **SW** comme des collections de points de terminaison du réseau, ou des **ports**. Dans WSDL, la définition abstraite des points de terminaison et des messages est séparé de leur déploiement de réseau concret ou de fixations de format de données. Cela permet la réutilisation des définitions abstraites: **messages**, qui sont des descriptions abstraites des données échangées, et les **types de ports** qui sont des collections abstraites des **opérations**. Le protocole concret et les spécifications de format de données pour un type de port particulier, constitue une **liaison** réutilisable. Un port est défini par l'association d'une adresse réseau avec une fixation réutilisable, et une collection de ports définit un SW. Ainsi, un document WSDL utilise les éléments suivants dans la définition des SW de réseau [<http://www.w3.org/TR/wsdl>]:

- **Types**, fournit des définitions de type de données utilisé pour décrire les messages échangés (cf. Figure 34). Il existe deux sortes de types, l'une est type simple, et l'autre est type complexe qui est composé des types simples.
- **Message**, représente une définition abstraite des données transmises (cf. Figure 35).
- **Operation**, est une description abstraite d'une fonction supporté par le SW.
- **PortType**, est un ensemble d'opérations abstraites. Chaque opération fait référence à un message d'entrée et de sortie (cf. Figure 36).
- **Binding**, précise le protocole concret et les spécifications pour les opérations et les messages définis par un Port Type particulier (cf. Figure 37).
- **Port**, associe une adresse unique à un élément Binding.
- **Service**, est utilisé pour regrouper un ensemble de ports associés (cf. Figure 38).

```

<types>
  <schema targetNamespace="http://example.com/stockquote.xsd"
    xmlns="http://www.w3.org/2000/10/XMLSchema">
    <element name="TradePriceRequest">
      <complexType>
        <all>
          <element name="tickerSymbol" type="string"/>
        </all>
      </complexType>
    </element>
    <element name="TradePrice">
      ...
    </element>
  </schema>
</types>

```

Figure 34 : Exemple de Type

```

<message name="GetLastTradePriceInput">
  <part name="body" element="xsd:TradePriceRequest"/>
</message>

<message name="GetLastTradePriceOutput">
  <part name="body" element="xsd:TradePrice"/>
</message>

```

Figure 35 : Exemple de Message

```

<portType name="StockQuotePortType">
  <operation name="GetLastTradePrice">
    <input message="tns:GetLastTradePriceInput"/>
    <output message="tns:GetLastTradePriceOutput"/>
  </operation>
</portType>

```

Figure 36 : Exemple de Port Type

```

<binding name="StockQuoteSoapBinding"
  type="tns:StockQuotePortType">
  <soap:binding style="document"
  transport="http://schemas.xmlsoap.org/soap/http"/>
  <operation name="GetLastTradePrice">
    <soap:operation
  soapAction="http://example.com/GetLastTradePrice"/>
    <input>
      <soap:body use="literal"/>
    </input>
    <output>
      <soap:body use="literal"/>
    </output>
  </operation>
</binding>

```

Figure 37 : Exemple de Binding

```

<service name="StockQuoteService">
  <documentation>My first service</documentation>
  <port name="StockQuotePort" binding="tns:StockQuoteBinding">
    <soap:address location="http://example.com/stockquote"/>
  </port>
</service>

```

Figure 38 : Exemple de Service

3. UDDI (Universal Description, Discovery and Integration):

Il faut connaître les fournisseurs de SW et les SW ce qu'ils fournissent pour pouvoir utiliser un SW. UDDI permet aux fournisseurs de publier leurs SW en les décrivant, et il permet de les scanner et les trouver par des demandeurs. UDDI Registres d'Affaire (Business Registry) sont des serveurs qui tiennent les informations des entreprises et leur SW. Ces serveurs enregistrent les informations dans la base de données ce qui viennent des fournisseurs de service, et permet de les accéder aux autres. Il existe deux registres d'affaire actuellement, l'une est uddi.microsoft.com et l'autre est uddi.ibm.com. Ces serveurs partagent les informations pris avec les autres serveurs pour augmenter la rapidité d'accès (cf. Figure 39). Les serveurs d'UDDI réalisent les enregistrements, les modifications et les balayages en utilisant les SW (avec les messages de SOAP) [<http://xml.coverpages.org/UDDI-TechnicalWhitePaperOct28.pdf>].

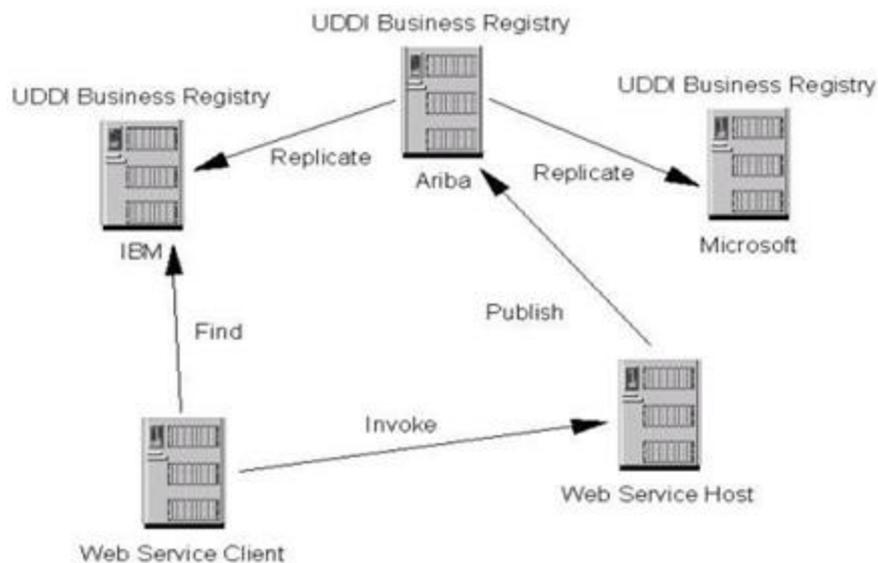


Figure 39 : Serveurs de Registre des Affaires

VII. Conclusion :

Les Web Services améliorent l'intégration et l'interopérabilité entre services à travers l'infrastructure Web en utilisant différents standards XML : SOAP pour l'échange de messages, WSDL pour la description de services et UDDI pour la publication et la découverte de services. Nous avons expliqué dans cette annexe ce qui le web service, ces caractéristique, les avantages qu'il apporte, nous avons aussi présenté brièvement ces déférente architecture et ces standard de base et tout cela en vu de mieux assimilé la section sur le web service sémantique dans le présent rapport.

UMMTO 2010/ 2011