Une Architecture à base d'Ontologies pour la Gestion Unifiées des Données Structurées et non Structurées

Gayo Diallo

▶ **To cite this version:**

Gayo Diallo. Une Architecture à base d'Ontologies pour la Gestion Unifiées des Données Structurées et non Structurées. Interface homme-machine [cs.HC]. Université Joseph-Fourier - Grenoble I, 2006. Français. NNT : . tel-00221392

HAL Id: tel-00221392

https://theses.hal.science/tel-00221392

Submitted on 28 Jan 2008

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Université Joseph Fourier – Grenoble I **École Doctorale MSTII**

Une Architecture à Base d'Ontologies pour la Gestion Unifiée des Données Structurées et non Structurées

THÈSE

présentée et soutenue publiquement le 11 Décembre 2006

pour obtenir le grade de

Docteur de l'Université Joseph Fourier – Grenoble I

(Spécialité : Informatique)

par

Gayo Diallo

Composition du jury

Président :	M. Yves Chiaramella	Professeur à l'Université Joseph Fourier Grenoble 1
Rapporteurs :	M. Chirstophe Roche	Professeur à l'Université de Savoie
	M. Djamel Zighed	Professeur à l'Université Lyon 2
Examinateurs :	Mme. Sylvie Calabretto	Maître de Conférences (HDR) à l'INSA de Lyon
	M. Michel Simonet	Chargé de recherche CNRS (Directeur de thèse)
	Mme. Ana Simonet	Maître de Conférences à l'UPMF de Grenoble (Co-encadrante)

Laboratoire des Techniques de l'Imagerie, de la Modélisation et de la Cognition
TIMC-IMAG — UMR 5525

Remerciements

Je tiens à remercier toutes les personnes qui ont contribué de manière directe ou indirecte à l'aboutissement de ce travail :

En premier lieu, je remercie très sincèrement Michel Simonet et Ana Simonet, Directeur et Directeur ajoint de l'équipe OSIRIS du laboratoire TIMC/IMAG de m'avoir accueilli au sein de l'équipe et encadré durant ma thèse. Leurs critiques constructives ont été d'un apport déterminant pour l'aboutissement de cette thèse.

Je remercie sincèrement Messieurs Christophe Roche, Professeur à l'Université de Savoie et Djamel Zighed, Professeur à l'Université Lyon 2, d'avoir accepté d'être les rapporteurs de ma thèse.

Je remercie également Monsieur Yves Chiaramella, Professeur à l'Université Joseph Fourier Grenoble 1 et Madame Sylvie Calabretto, Maître de Conférences (HDR) à l'INSA de Lyon, d'avoir accepté de participer au jury de ma thèse.

Je remercie mes sympathiques collègues de l'équipe OSIRIS pour leur présence et leur soutien cordial. Je remercie tout particulièrement Delphine, Lamis, Sylvain, Rémi, que j'ai régulièrement embêtés avec mes incéssantes questions ; Radja, Houda, Samer, Shokouh, Cyr ainsi que les anciens membres de OSIRIS que je n'aurais pas explicitement cités pour leur soutien et leurs encouragements...Mes remerciements vont également à l'endroit de Patrick Palmer, Maître de conférence à l'IUT 2 de Grenoble pour sa disponibilité à mon égard. Je n'oublie pas les partenaires du projet BC^3.

Je remercie enfin ma famille, mes proches et amis pour leur compréhension et leurs encouragements.

Gayo Diallo

A mes parents

Table des matières

Liste des tableaux

Table des figures

Introduction générale

Les données rendues disponibles par les systèmes informatiques sont de plus en plus nombreuses et variées, grâce notamment au développement des nouvelles technologies de l'information. Ces données sont souvent disséminées dans des sources d'information distinctes, gérées par les différents systèmes d'information des organisations.

Disposer d'une information particulière pour une bonne prise de décision peut nécessiter l'accès et le recoupement d'informations provenant de plusieurs de ces sources. Cependant, les systèmes d'information sont conçus en général pour un fonctionnement autonome et non pour un besoin d'intégration, et le travail de combinaison des informations issues des sources doit être fait de façon manuelle. Ce travail de combinaison est une activité qui peut se révéler très vite complexe, et coûteuse en temps et en argent.

1 Position du problème

Pour situer le problème que nous abordons dans cette thèse, nous allons tout d'abord présenter un exemple montrant la nécessité d'interagir avec plusieurs sources d'information.

1.1 Principale motivation

Prenons le cas du centre hospitalier de la Pitié Salpêtrière et plus particulièrement son service neuropsychologique[1]. Ce service est spécialisé dans le bilan des séquelles neuropsychologiques des lésions cérébrales et a donc en charge le suivi de patients cérébro-lésés. A cet effet, ce service collecte et traite des données médicales anatomiques, neuropsychologiques et comportementales de patients atteints de lésions cérébrales. L'interprétation de ces différentes données permet aux praticiens de mieux cerner d'une part les désordres produits par les lésions sur les patients et d'autre part de définir une stratégie optimale en termes de rééducation fonctionnelle.

Les patients qui arrivent au centre sont évalués sur le plan anatomique et fonctionnel afin d'élaborer un bilan anatomo-fonctionnel. Les patients subissent pour cela des examens :

1. neuropsychologiques, pour leur évaluation sur le plan cognitif, à travers des batteries de **tests cognitifs** ;
2. d'imagerie cérébrale, pour localiser les **zones anatomiques** touchées par les lésions ;
3. éventuellement un entretien d'évaluation comportementale.

Ces différents examens donnent lieu à plusieurs résultats, consignés dans des comptes rendus médicaux, des bases de données de gestion des patients, des banques d'images, etc. (voir l'illustration de la figure 1). Par ailleurs, l'établissement de certains diagnostics peut amener les praticiens à consulter la littérature spécialisée, disponible sous forme de publications scientifiques sur le domaine.

[1] Cas issu du projet Bases de Connaissances Coeur-Cerveau (BC^3) (2000-2003) de la Région Rhône-Alpes

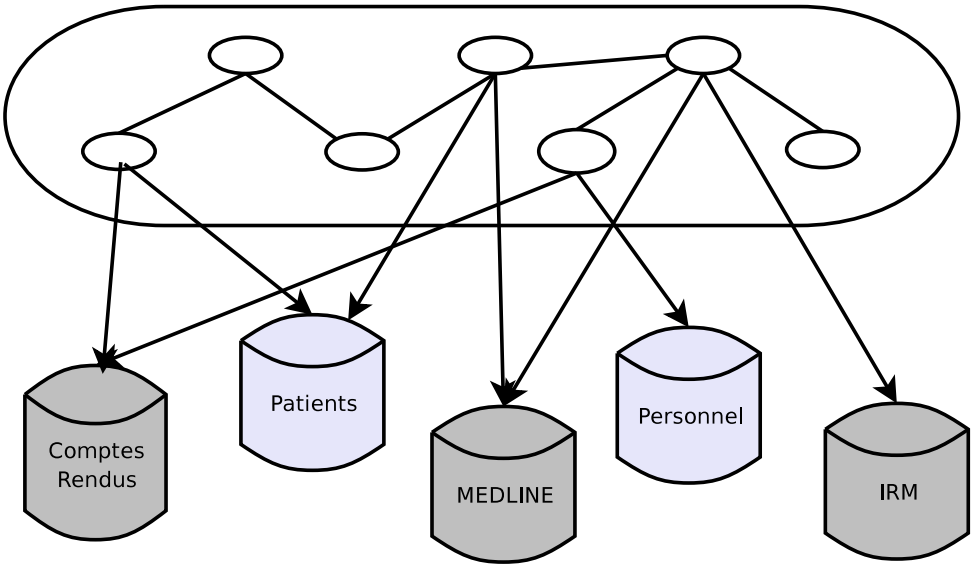


FIG. 1 – Scénario d'intégration

Les données issues des différents examens que nous avons évoqués ci-dessus sont de natures diverses et sont stockées sur des supports différents : serveurs de fichiers et station de traitement d'images pour les unes, bases de données relationnelles ou pages Web pour les autres. Elles ont pourtant entre elles des liens que l'homme peut identifier relativement facilement. Un compte-rendu médical (contenant des résultats d'évaluations cognitives par exemple) concerne un patient donné. La lésion cérébrale identifiée (à partir de l'imagerie IRM) chez tel patient est localisée dans tel hémisphère de son cerveau. Les résultats d'une évaluation cognitive donnée sont obtenus par tels praticiens, etc.

Il est communément admis que l'accès individualisé à ces différentes sources d'information reste possible. Les comptes rendus médicaux, par exemple, disponibles sous forme textuelle, peuvent être retrouvés à l'aide des méthodes classiques de recherche d'information, tandis que le langage d'interrogation propre aux bases de données est utilisé pour accéder aux données des patients. Il appartient ensuite aux praticiens de combiner manuellement les données fournies par les différentes sources s'ils veulent avoir une vue globale de l'information souhaitée. Ce travail peut rapidement devenir complexe, même si ces praticiens disposent d'une **expertise** qui leur permet d'établir des liens entre les données.

Proposer des mécanismes pour un accès unifié à des données dispersées dans plusieurs sources est l'objectif principal de la gestion unifiée de données. Notre travail de thèse se situe dans ce cadre.

L'illustration que nous avons présentée est issue du domaine médical. Le problème de la gestion unifiée (ou encore de l'intégration de données) est cependant beaucoup plus général. Il se rencontre de nos jours fréquemment dans le domaine des entreprises commerciales, où le besoin d'intégration de données issues de différents systèmes d'information devient de plus en plus crucial.

Tab. 1 – Comparaison entre recherche de données et recherche d'information

	Recherche de Données	**Recherche d'information**
Raisonnement	Déductif	Inductif
Modèle	Déterministe	Probabiliste
Language de Requête	Formule	Naturel
Spécification de la requête	Totale	Partielle
Appariement	Exact	Meilleur appariement
Objet sélectionné	Appariement de la requête	Pertinence

1.2 Le problème général de la gestion unifiée des données

Intégrer plusieurs sources d'information a généralement pour objectif de combiner ces sources, de telle sorte qu'elles apparaissent comme une **source unique** et donnent aux utilisateurs l'illusion de n'interagir qu'avec cette seule source. Les données, distribuées sur plusieurs sources hétérogènes, sont présentées aux utilisateurs à travers une vue logique unique. Elles doivent donc être représentées en utilisant les mêmes principes et le même niveau d'abstraction (avec un schéma global et une sémantique unifiés). La représentation unifiée des données nécessite la détection et la résolution d'éventuels conflits de schémas et d'éventuels conflits de données, tant du point de vue structurel que sémantique [Sheth et Kashyap, 1992][Rahm et Bernstein, 2001].

1.3 Recherche de données Vs Recherche d'information

Manipuler à la fois des données textuelles (contenant parfois des items pouvant être clairement identifiés, tels que les résultats d'une évaluation cognitive) et des données de nature structurée, nous amène à faire une distinction entre la recherche de données et la recherche d'information.

Pour Lancaster, "*La recherche d'information est le terme conventionnellement, parfois de façon inappropriée, appliqué au type d'activité abordé dans ce volume (son ouvrage de RI). Un système d'information n'informe pas (ne change pas la connaissance de) l'utilisateur sur le sujet de sa demande. Il l'informe simplement sur l'existence (ou la non existence) de documents concernant sa demande*" [Lancaster, 1968]. Lancaster exclut donc les systèmes Question/Réponse tels qu'ils sont définis par Winograd [Winograd, 1972] ou décrits par Minsky [Minsky, 1968]. Il exclut aussi les Systèmes de Recherche de Données tels que ceux utilisés par les systèmes de cotation en ligne des bourses d'échange [van Rijsbergen, 1979]. Van Rijsbergen établit une distinction claire entre ces deux notions : la recherche de données (RD) procède par recherche exacte tandis que la recherche d'information utilise des principes de recherche dits approchés. Le tableau 1, tiré de [van Rijsbergen, 1979], résume les principales caractéristiques des deux modes de recherche. Van Rijsbergen note que cette dichotomie peut être discutée car la frontière entre les deux modes de recherche est vague. Elle permet néanmoins d'illustrer le niveau de complexité associé à chaque mode. D'autres différences entre les deux modes de recherche, inhérentes aux caractérisques données dans le tableau 1, sont à considérer :

1. le langage de requête pour la RD est généralement de nature artificielle, avec une syntaxe et un vocabulaire restreints, tandis qu'en RI on utilise plutôt le langage naturel. En RD, la requête est généralement une spécification complète de ce qui est voulu tandis qu'en RI elle est invariablement incomplète ;
2. en RD on est généralement intéressé par une classification monothétique, où les éléments à

classer possèdent tous les attributs nécessaires et suffisants pour appartenir à une classe. En RI c'est plutôt une classification polythétique qui est desirée, un objet pouvant être classé dans une classe même s'il ne possède pas tous les attributs nécessaires pour appartenir à la classe.

Enfin, comme nous l'avons indiqué, la RD utilise les principes de la recherche exacte. Par conséquent, aucune erreur n'est tolérée dans les résultats. En revanche, les "erreurs" en RI sont acceptées et sont caractérisées par les notions de précision et de rappel.

Unifier la gestion des données de nature structurée et des données de nature non structurée nécessite de tenir compte des considérations que nous venons de faire.

1.4 Problème abordé par le travail de thèse

Le problème de l'intégration classiquement abordé par la communauté des bases de données (intégration de schémas, résolution de conflits, réécriture de requêtes, etc.) est essentiellement celui de l'hétérogénéité entre différentes représentations des mêmes entités du monde réel. Par ailleurs, le problème de l'intégration des données textuelles et des données structurées n'est pas explicitement traité. En effet, même si des travaux relatent des cas d'intégration dans lesquels certaines sources contiennent des informations de nature textuelle (cas des publications scientifiques, souvent cité en exemple), leur processus d'intégration tient compte uniquement du schéma servant à représenter ces sources textuelles et non du contenu des sources elles-mêmes. Dans le cas des publications scientifiques, par exemple, on utilise une représentation basée sur la modélisation du domaine (les auteurs, les types de publications, les éditeurs, etc.). Pour notre part, nous considérons que traiter le problème d'intégration en considérant cette représentation des sources revient à traiter un cas classique d'intégration de données structurées.

Le problème général abordé par notre travail de thèse concerne l'intégration de données structurées (bases de données relationnelles) et de données non structurées (sources textuelles, pouvant être multilingues), avec l'objectif d'offrir aux utilisateurs une vue homogène de l'information véhiculée par ces deux médias. Nous traitons en particulier le problème de l'intégration des sources pouvant contenir des informations différentes, mais qui peuvent avoir des liens au travers de sources externes de connaissances – dans notre cas des ontologies ou des classifications. Nous nous intéressons notamment à la prise en compte de sources textuelles dans une infrastructure de gestion unifiée.

2 Principales contributions

2.1 Première contribution : Une architecture d'intégration à base d'ontologies

Plusieurs travaux sur l'intégration de données utilisant des ontologies ont été reportés dans la littérature [Mena *et al.*, 2000][Rousset et Reynaud, 2003][Arens *et al.*, 1997] et particulièrement dans le domaine biomédical [P ?rez-Rey *et al.*, 2006][Stevens *et al.*, 2000]. Les ontologies sont utilisées dans les systèmes issus de ces travaux, exclusivement pour représenter le schéma global de médiation et/ou les schémas des sources locales. Eventuellement, plusieurs ontologies peuvent être utilisées à cet effet [Mena *et al.*, 2000]. L'approche à base de médiateurs que nous proposons repose sur l'utilisation des technologies du Web Sémantique et de plusieurs types d'ontologies [Diallo *et al.*, 2006b], les ontologies étant également utilisées pour la caractérisation sémantique des sources non structurées (textuelles). Les ontologies servent d'une part à définir le schéma global d'intégration (ontologie globale) et les différentes sources à intégrer : les ontologies qui

représentent les sources à intégrer sont appelées schémas virtuels de sources ou ontologies locales, et les correspondances sont établies entre l'ontologie globale et les différentes ontologies locales. D'autre part, les ontologies permettent d'obtenir une représentation hybride de chaque source textuelle de façon automatique [Diallo *et al.*, 2006a]. Par ailleurs, à partir de sources d'informations identiques, plusieurs besoins d'intégration peuvent se faire sentir. Ces besoins nécessitent différentes vues sur le schéma global de l'organisation. L'architecture prend en compte ces différentes vues (ou points de vue) et ne s'appuie pas sur un schéma figé unique.

2.2 Seconde contribution : Représentation hybride et interrogation de sources textuelles

Parent et Spaccapietra [Parent et Spaccapietra, 1996] notent que la plupart des recherches menées dans le domaine de l'intégration supposent que les schémas initiaux sont exprimés avec un modèle commun, sur lequel le système d'intégration est construit. L'étape de traduction vers ce modèle commun est un prérequis et est traité comme un problème à part. Une des originalités de notre travail est l'étape préalable de traduction de toute source non structurée vers une représentation pivot, appelée représentation hybride de sources textuelles. Pour arriver à cette représentation, nous utilisons des sources externes de connaissances, dans notre cas des ontologies ou des classifications, qui encodent la connaissance du domaine concerné. La première étape de la représentation hybride est l'identification dans les documents de concepts issus des sources externes de connaissances. La seconde étape est l'identification d'entités nommées (e.g., des valeurs de tests cognitifs) dans les documents textuels. En plus de la représentation classique des documents textuels (basée sur les mots ou termes simples), nous avons proposé et implémenté une méthode d'identification automatique de concepts dans des sources textuelles et une méthode de recherche d'information qui combine les termes simples et les concepts issus d'ontologies. Garder les termes simples dans la représentation des sources est motivé par le fait que les sources externes de connaissances couvrent rarement la totalité du vocabulaire de la source à représenter. Les termes simples constituent donc un complément à la recherche (uniquement) conceptuelle. Dans la suite, nous utiliserons indifféremment les termes représentation hybride et représentation sémantique.

2.3 Troisième contribution : Construction semi-automatique des schémas virtuels des sources

Après la phase de représentation hybride des sources textuelles, nous avons obtenu un modèle pivot unique (schéma relationnel) de chaque source à intégrer. En vue d'établir les correspondances avec l'ontologie qui sert de schéma global pour l'interrogation, nous avons élaboré une méthode de construction semi-automatique des schémas virtuels de sources locales, avec pour objectif de les décrire dans les langages OWL/RDF, recommandés par le W3C pour décrire des ontologies. Cette méthode utilise des techniques de ré-ingénierie de bases de données [Diallo *et al.*, 2006b]. Le résultat obtenu pour chaque source est un schéma local contenant un ensemble de vues (nécessaires pour le besoin d'intégration) et les paramètres d'accès à la source. L'implémentation de traducteurs de requêtes posées dans les langages d'interrogation du Web sémantique vers SQL permet d'interroger le schéma virtuel et de récupérer les données de la base. C'est une étape décisive pour s'affranchir de SQL pour l'interrogation de bases relationnelles ou de l'exportation des données vers les infrastructures Web Sémantique, ainsi que le préconisent la plupart des travaux actuels dans ce domaine [de Laborda et Conrad, 2005].

2.4 Quatrième contribution : Gestion conjointe de plusieurs ontologies

L'interopérabilité des différentes sources au niveau du vocabulaire peut s'opérer au travers d'une capture du vocabulaire contenu dans ces sources. Plusieurs sources externes – ontologies, classifications – peuvent être utilisées comme source de vocabulaire. Nous avons élaboré une approche de gestion conjointe de multiples sources de connaissances à travers un serveur qui implémente deux algorithmes utilisant des techniques issues de la recherche d'information : l'un pour l'indexation d'ontologies et le second pour la recherche sémantique. Le serveur d'ontologies est capable d'intégrer toute ontologie/classification décrite dans les langages du Web sémantique RDF/OWL [Diallo *et al.*, 2006b]. L'approche que nous avons implémentée permet aussi d'autres utilisations, comme la recherche de similarités entre ontologies.

Nous avons, compte tenu de notre premier domaine d'application qu'est le cerveau, construit ou enrichi des ontologies servant en premier lieu à la représentation hybride de sources textuelles dans ce domaine. Ces ontologies sont réutilisables dans d'autres contextes, par exemple l'enseignement et l'indexation de ressources éducatives sur le cerveau.

3 Organisation de la thèse

La thèse est constituée de deux parties principales. La première partie comporte trois chapitres sur l'état de l'art en rapport avec les thèmes qu'aborde notre travail. La seconde partie décrit notre apport concernant la gestion unifiée.

3.1 Première partie

Le **premier chapitre** est consacré aux ontologies et au Web sémantique. Il introduit les notions essentielles sur les ontologies et leur ingénierie. L'utilisation des ontologies pour la recherche d'information conceptuelle y est également présentée.

Le **second chapitre** présente l'intégration de sources hétérogènes et les différentes approches utilisées. Nous décrivons les principaux problèmes d'hétérogénéité rencontrés et traités en général par la communauté des bases de données. Après avoir précisé la place des ontologies dans l'intégration, nous décrivons quelques systèmes d'intégration représentatifs.

Le **troisième chapitre** présente les notions et concepts de base de la recherche d'information. Nous présentons les principaux modèles utilisés pour implémenter les systèmes de recherche d'information (SRI) et nous abordons la notion de recherche d'information à base d'ontologies. Nous abordons également dans ce chapitre la notion d'annotation de documents. L'activité d'annotation permet d'acquérir des métadonnées sur les documents. Ces métadonnées sont par la suite utilisées par un SRI.

3.2 Deuxième partie

La seconde partie, composée de quatre chapitres, concerne notre contribution pour la gestion unifiée des données structurées et non structurées.

Le **quatrième chapitre** est consacré à la présentation de l'approche que nous proposons pour la gestion unifiée. Nous présentons son architecture globale et nous détaillons ses différentes couches. Nous abordons également notre approche de représentation d'une source structurée à l'aide de techniques de ré-ingénierie de bases de données et de technologies du Web sémantique. Le schéma virtuel permettant de décrire une base relationnelle est présenté.

Dans le **chapitre 5** le modèle des ontologies utilisé pour la représentation hybride de sources non structurées est présenté. Ensuite, le principe de construction de ce type d'ontologies à partir de ressources existantes est décrit. Une application de ce principe à la construction d'ontologies dans le domaine du cerveau est décrite.

Le **chapitre 6** est consacré à l'approche de représentation hybride de sources non structurées à partir de sources externes de connaissances – ontologies ou classifications. La représentation hybride constitue la première phase de notre approche d'intégration et permet d'apporter une certaine structure à chaque source textuelle.

Le **chapitre 7** le serveur d'ontologies, qui permet la gestion conjointe de multiples ontologies, est décrit. Trois modes de gestions de l'index des ontologies sont proposés (gestion de l'index en mémoire, gestion de l'index à travers un fichier temporaire et enfin une gestion de l'index de façon pérenne). Nous détaillons le processus d'indexation ainsi que le processus de recherche d'ontologies et nous présentons le module implémenté à cet effet. Nous présentons également le prototype qui permet un accès Web au serveur d'ontologies et d'effectuer des interrogations sur des sources textuelles.

En **conclusion**, nous dressons le bilan des travaux réalisés dans le cadre de la gestion unifiée, en tenant compte de l'étape préalable de représentation des données non structurées (textuelles). Nous dégageons ensuite les perspectives envisageables pour ces travaux.

Première partie

Etat de l'Art

Chapitre 1

Web Sémantique et ontologies

1.1 Introduction

"*Il y a une science qui étudie l'être en tant qu'être et les attributs qui lui appartiennent essentiellement.*" Aristote, livre IV de la Métaphysique.

Le Web Sémantique (WS), qui est une extension du Web actuel, a comme objectif majeur d'apporter de la sémantique à l'information manipulée afin de faciliter la communication entre agents (hommes et machines). Tim Berners-Lee, auteur du premier article sur ce thème, et ses collègues prenant acte du fait que le Web actuel n'est "compréhensible" que par les humains, indique que l'objectif du WS est de fournir aux machines un meilleur accès (automatisé) à l'information [Berners-Lee *et al.*, 2000] grâce à des métadonnées qui décrivent les informations disponibles. Ces métadonnées sont fournies par le composant pivot du WS, les ontologies. Ces dernières représentent une technologie dont le but est d'améliorer l'organisation, la gestion et la compréhension de l'information électronique. Elles servent de vocabulaire standardisé pour le partage de connaissances. Le World Wide Web Consortium (W3C)[2] soutient les activités liées au WS à travers le Web Ontology Working Group[3].

Dans ce chapitre, nous allons expliciter les différentes notions et concepts liés aux ontologies et leur ingénierie.

1.2 Ontologies : définitions

Plusieurs définitions du terme ontologie ont été proposées selon les courants et les communautés de pensée. Nous en reprenons les principales dans cette section.

1.2.1 Du point de vue de la Métaphysique

Historiquement, l'Ontologie (avec "O") est d'abord une notion philosophique. Nous pouvons en trouver une définition succincte dans les dictionnaires :

Le Petit Robert (Ed. 1983) : n.f., (1692 ; lat. philo. Ontologia, 1646). Philo. Partie de la métaphysique qui s'applique à l'être en tant qu'être, indépendamment de ses déterminations particulières.

[2] www.w3.org/ fondé en octobre 1994 pour promouvoir la compatibilité des technologies du Web
[3] http ://www.w3.org/2001/sw/WebOnt/

Le Petit Larousse (Ed. 1998) : n.f. (gr. ontos, être, et logos, science). PHILOS. 1. Etude de l'être en tant qu'être, de l'être en soi. 2. Etude de l'existence en général, dans l'existentialisme.

Encyclopaedia Universalis (Ed. 2006) : "Ontologie" veut dire : doctrine ou théorie de l'être. Cette simple définition, toute nominale d'ailleurs, propose une petite énigme de lexique : le mot "ontologie" est considérablement plus récent que la discipline qu'il désigne ; ce sont les Grecs qui ont inventé la question de l'être, mais ils n'ont pas appelé ontologie la discipline qu'ils instituaient. Aristote désigne de façon indirecte comme "la science que nous cherchons" la théorie de l'être en tant qu'être.(...)

L'acception contemporaine des ontologies qui nous intéresse ici n'est pas mentionnée. Tout au plus trouve-t-on dans la définition du Petit Robert l'idée d'abstraction, et, indirectement, d'instance, à travers la caractérisation " **indépendamment de ses déterminations particulières**". L'origine de la notion d'ontologie remonte donc à Aristote (384-322BC.), bien que le terme lui-même soit plus récent. Dans la Metaphysique, il explique que la réalité se présente sous la forme d'individualités uniques et particulières (Platon, Socrate) qu'il faut aborder à partir de concepts généraux (philosophe, homme, être vivant). Pour penser un être existant, il faut définir des propriétés (substance, qualité, quantité, lieu, temps, situation ?), regroupées par Aristote en dix **catégories** qui, selon lui, appartiennent à la réalité et ne sont pas de simples constructions mentales. Les propriétés ainsi utilisées pour caractériser les concepts ne sont pas sans évoquer les attributs utilisés aujourd'hui dans différents modèles de représentations de connaissances.

Le travail sur l'ontologie consiste donc à déterminer ce qui est universel d'un être, par delà ses représentations particulières. Après Aristote, c'est Porphyre, philosophe grec du troisième siècle de notre ère, qui a attaché son nom à l'étude de l'ontologie, à la fois sur un plan religieux et sur un plan "scientifique", en insistant, pour la catégorisation des êtres, sur les traits qui les opposent (catégorisation par identité et différence). On peut y voir l'origine de l'organisation taxinomique en usage dans différents domaines scientifiques. Cette approche binaire des ontologies est reprise aujourd'hui par C. Roche dans le système Ontologos [Roche, 2003] ainsi que B. Bachimont [Bachimont, 2000].

Depuis Russell et les Principles of Mathematics (1903), différents philosophes et mathématiciens ont travaillé sur la notion d'ontologie et sur le travail de catégorisation, qui lui est connexe. Citons Wittgenstein, Frege, Husserl, Peirce, Thom. Le courant philosophique reste très présent, et connaît aujourd'hui un regain d'activité [Smith, 1995] [Smith, 1998].

1.2.2 Du point de vue de l'ingénierie des connaissances

Dans l'univers du traitement automatique de l'information – l'informatique – le terme "ontologie" semble avoir été introduit dans les années 1990 avec les travaux de Grüber et son équipe à Stanford [Gruber, 1993]. Il est intéressant de noter que la référence la plus consultée sur le web, obtenue lors d'une recherche par le moteur de recherche Google, est l'article de Grüber *What is an Ontology* où il écrit : "une ontologie est **une spécification formelle**, **explicite** d'une **conceptualisation partagée**". Il précise que cette définition est faite dans le contexte du partage et de la réutilisation des connaissances, et que ce qui importe est ce pourquoi (quel usage) une ontologie est faite. L'état de l'art fait dans le cadre du projet IST InterOp [InterOp, 2006] apporte une explication à la définition donnée par Grüber : "Une **conceptualisation** fait référence à un modèle abstrait de certains phénomènes du monde, modèle qui identifie les concepts pertinents de ce phénomène. **Explicite** signifie que le type des concepts utilisés et les contraintes sur leur utilisation sont explicitement définis. **Formelle** se réfère au fait que l'ontologie doit être compréhensible par les machines. **Partagée** reflète la notion de connaissance consensuelle décrite par l'ontologie, c'est-à-dire qu'elle n'est pas restreinte au point de vue de certains individus

seulement, mais reflète un point de vue plus général, partagé et accepté par un groupe.

Cette définition de la notion d'ontologie a guidé un ensemble de travaux sur la définition d'un format d'échange pour l'exploitation des ontologies (KIF : Knowledge Interchange Format) [Genesereth et Fikes, 1992] et la mise en place d'une plate-forme logicielle pour la construction et l'échange d'ontologies (ONTOLINGUA) [Gruber, 1992]. C'est dans cette communauté que le terme "ontologie" a connu une nouvelle vie, qui a conduit à ses acceptions actuelles.

Parmi les nombreuses autres définitions contemporaines, citons :

1. Welty : une ontologie est la définition des classes, relations, contraintes et règles d'inférence qu'une base de connaissances peut utiliser [Welty et Nancy, 1999] ;
2. Guarino : une ontologie est un vocabulaire partagé, plus une spécification (en fait une caractérisation) du sens "convenu" de ce vocabulaire [Guarino, 1998a] ;
3. Le W3C[4] : une ontologie définit les termes utilisés pour décrire et représenter un domaine de la connaissance.

La définition de Grüber est la plus largement adoptée. Une ontologie est une spécification formelle et explicite d'une conceptualisation partagée. Une conceptualisation est une vue particulière, abstraite, au sujet des entités réelles, des événements et de leurs relations. La connaissance incluse dans l'ontologie est une forme de connaissance consensuelle, qui n'est pas propre à un individu, mais qui est acceptée par un groupe.

Typiquement, une ontologie contient une description hiérarchique des concepts importants d'un domaine et décrit les propriétés de chaque concept à travers un mécanisme d'attributs-valeurs. Des relations entre les concepts peuvent être décrites au moyen de phrases logiques ou axiomes. Selon le langage utilisé pour définir l'ontologie, certaines notions peuvent être introduites pour la modélisation du domaine et du mécanisme d'inférence. Corcho [Corcho et Perez, 2000] note que la modélisation du domaine nécessite la définition de concepts, relations, axiomes, fonctions et instances. Le processus d'analyse d'un domaine dans le but de spécifier formellement une ontologie requiert plusieurs étapes. Une des plus importantes est l'identification des types d'objets (concepts), la spécification de leurs attributs, des types de relation qu'il peut y avoir. Un concept est défini comme le signifié d'un terme dont on décide de négliger la dimension linguistique.

1.3 Typologie des ontologies

En fonction de leur usage, on distingue classiquement cinq catégories d'ontologies [van Heijst *et al.*, 1997] : génériques, de domaine, d'application, de représentation et de méthodes.

Les ontologies génériques décrivent des concepts généraux, indépendants d'un domaine ou d'un problème particulier. Il s'agit par exemple des concepts de temps, d'espace, d'événement. Elles sont prévues pour être utilisées dans des situations diverses, et pour servir une large communauté d'utilisateurs. Les ontologies génériques sont aussi appelées Modèles de haut niveau ou ontologies TOP. L'ontologie de Sowa [Sowa, 1999] et CYC [Lenat et Guha, 1990] sont des ontologies génériques.

Les ontologies de domaine spécifient un point de vue sur un domaine particulier. Le vocabulaire est généralement lié à un domaine de connaissances générique comme la médecine ou la

[4]World Wide Web Consortium

loi. Les concepts d'une ontologie de domaine sont souvent définis comme une spécialisation des concepts des ontologies génériques. *Enterprise* est un exemple d'ontologie décrivant le domaine de l'entreprise[5]. Dans l'approche KADS [Wielinga *et al.*, 1992], l'ontologie de domaine est constituée de la terminologie du domaine d'application et d'un ensemble de relations et de types de base comme la relation classe/super-classe. Une ontologie de domaine et alors constituée de :

1. une description en extension du vocabulaire du domaine,
2. une typologie,
3. une hiérarchie ou un treillis de classes.

Les ontologies d'application Une ontologie d'application décrit la structure des connaissances nécessaires à la réalisation d'une tâche particulière [van Heijst *et al.*, 1997]. Elle permet aux experts du domaine d'utiliser le même langage que celui de l'application. Elle réalise trois objectifs :

1. faciliter le processus d'acquisition des connaissances de l'application ;
2. permettre l'intégration avec les serveurs existants dans l'environnement de l'application ;
3. sélectionner les structures de données appropriées au modèle computationnel .

Les ontologie de représentation spécifient un formalisme de description qui fournit une structure de représentation et des primitives pour décrire les concepts des ontologies de domaine et des ontologies génériques. La Frame Ontology en est un exemple type [Gruber, 1992]. La Frame Ontology est associée à Ontolingua, un système de description et de traduction d'ontologies dont la syntaxe et la sémantique s'appuient sur le langage KIF [Genesereth et Fikes, 1992].

Les ontologies de méthodes, de tâches et de résolution de problèmes décrivent le processus de raisonnement d'une façon indépendante d'un domaine et d'une implémentation donnée. Elles spécifient des entités qui relèvent de la résolution d'un problème telles que l'abduction, la déduction ou l'observation [Chandrasekaran *et al.*, 1998]. Elles fournissent les définitions des concepts et relations utilisés pour spécifier un processus de raisonnement lors de la réalisation d'une tâche particulière.

En plus de cette classification, l'institut AIFB de l'université de Karlsruhe distingue les ontologies légères *Light-weight ontology* et les ontologies riches *Heavy-weight ontology* [Studer *et al.*, 1998].

Une ontologie légère comprend des concepts, des types atomiques, une hiérarchie IS-A entre les concepts, et des relations entre les concepts. C'est le type d'ontologie le plus courant.

Une ontologie riche comprend des contraintes de cardinalité, une taxonomie de relations, des axiomes/héritages sémantiques (logique de description, logique de propositions, clauses de Horn, logiques d'ordre plus élevé) et suppose l'existance d'un système d'inférence.

On peut ajouter à cette classification la notion d'ontologie terminologique (dans la catégorie des ontologies légères).

[5] http ://www.aiai.es.ac.uk/project/enterprise

Les ontologies terminologiques sont définies par SOWA[6] comme étant des ontologies dont les catégories ne sont pas spécifiées par des axiomes et des définitions. Un exemple est WordNet [Fellbaum, 1999] dont les catégories sont partiellement spécifiées par des relations telles que l'hyperonymie et la méronymie, qui déterminent les positions relatives des concepts, sans pour autant les définir de façon complète. La plupart des organisations établissent des conventions de nommage, des classifications, des standards. Ces éléments sont un début de formalisation.

Les ontologies peuvent fournir du vocabulaire pour la caractérisation sémantique de documents textuels. Une distinction mérite cependant d'être faite entre ontologie et terminologie.

1.4 Ontologies Vs Terminologies

Une distinction entre terminologie et ontologie est apportée par C. Roche qui stipule que "*...parler de terminologie c'est parler des mots et du sens des mots... L'étude du sens d'un mot relève de la linguistique...*". Dans une terminologie on s'intéresse aux mots et aux relations entre eux ; la relation structurante de base est la relation d'hyperonymie et son inverse l'hyponymie, tandis que dans une ontologie, on s'intéresse à la notion de concept et aux relations entre eux. La relation structurante de base est la "subsumption", qui relève de l'épistémologie (au sens de la théorie de la connaissance). Elle organise les concepts par abstraction de caractères communs pour aboutir à une hiérarchie de concepts correspondant à une organisation taxonomique des objets [Roche, 2005].

Cependant, ontologies et terminologies servent au même objectif : fournir à une communauté d'utilisateurs une conceptualisation partagée d'une partie spécifique du monde, dans le but de faciliter une communication efficiente d'une connaissance complexe. Gamper fournit quatre critères pour distinguer les deux notions [Gamper *et al.*, 1999] :

1. le cadre formel de leur définition : la science de la Terminologie utilise le texte libre en langue naturelle pour définir le sens des termes. L'interprétation correcte du sens véhiculé dépend de l'utilisateur. Les ontologies spécifient explicitement la connaissance conceptuelle en utilisant un langage formel avec une sémantique claire, qui permet une interprétation non ambiguë des termes ;
2. le support computationnel : les outils disponibles et leur confort d'usage diffèrent d'une discipline à l'autre. La plupart des terminologies utilisées actuellement ne fournissent pas (ou très peu) de sémantique pour une représentation explicite aussi bien de la connaissance que pour la maintenance des données. Tandis que pour les ontologies, grâce à l'utilisation de langages de représentation formels (e.g., les Loqiques de Description (DLs)[Nardi et Brachman, 2003]), il est possible de vérifier leur consistance, d'inférer de nouvelles connaissances, etc. ;
3. les utilisateurs : les terminologies sont orientées vers des utilisateurs humains (les traducteurs et autres experts de domaine étant leurs premiers utilisateurs). Les ontologies sont principalement développées pour le partage de connaissances entre agents (humains et machines) ;
4. l'usage de la langue naturelle : la terminologie s'occupe du transfert de la connaissance comme une activité linguistique, c'est-à-dire l'exploration de la langue naturelle afin d'identifier tous les termes utilisés par les humains pour faire reférence aux concepts sous-jacents. Les ontologies quant à elles sont destinées principalement à un usage computationnel et

[6] http ://www.jfsowa.com/ontology/gloss.htm (consulté le 3 Octobre 2006)

peuvent souvent ignorer l'importance de nommer les concepts avec des termes "parlants" (e.g, le CUI d'UMLS (voir la Section 1.8.1)).

La question de savoir si une ontologie donnée doit être représentée de façon formelle ou informelle est directement liée à l'objectif de cette ontologie [Uschold et King, 1995]. Si le but est d'assurer la communication entre des personnes, une description en langue naturelle peut suffire. Par contre, si elle doit assurer le rôle d'un format d'échange dans un environnement d'interopérabilité entre programmes, il est nécessaire qu'elle soit formelle afin que la communication soit cohérente et que la vérification automatique de cette cohérence soit possible.

1.5 Représentation des ontologies

Penser une ontologie ne peut se faire sans un formalisme pour la représenter. Le langage utilisé pour décrire les termes d'une ontologie a un impact direct sur le niveau de formalisme de l'ontologie. Ainsi, on distingue les ontologies informelles, qui utilisent le langage naturel et qui peuvent coïncider avec les terminologies, les ontologies semi-formelles, qui fournissent une faible axiomatisation, comme les taxonomies, et enfin les ontologies formelles, qui définissent la sémantique des termes par une axiomatisation complète et rigoureuse.

KIF et Ontolingua, qui ont marqué la naissance des travaux contemporains sur les ontologies dans le monde informatique, s'appuyaient sur la logique du premier ordre, en y introduisant quelques limites pour en permettre la manipulation informatique. Depuis, les formalismes les plus utilisés ont été les graphes conceptuels [Sowa, 1998], les frames [Minsky, 1975] et les DLs. Ces deux derniers formalismes, bénéficiant d'implantations informatiques, ont occupé le devant de la scène, et se trouvent aujourd'hui réunis dans l'initiative du Web Sémantique à travers les langages DAML+OIL [McGuinness *et al.*, 2002] et OWL [McGuinness et van Harmelen, 2004].

On retrouve là les fondements du formalisme des réseaux sémantiques [Quillian, 1968] en intelligence artificielle, modèle qui est aussi à l'origine des graphes conceptuels et des logiques de description. Ce modèle est considéré comme le plus expressif pour la communication entre humains [Abrial, 1974]. Cependant, il ne permet pas directement une implantation efficace dans le cadre de bases de données ou de bases de connaissances, d'où l'émergence des modèles de frames puis des Logiques de Description, qui proposent une structuration des connaissances permettant des inférences calculables.

Plusieurs langages, dont la syntaxe est basée sur le langage XML, ont été conçus pour une utilisation des ontologies dans le cadre du WS, notre centre d'intérêt. Parmi eux, les plus importants sont RDF/RDF(S) [Lassila et Swick, 1999][Decker *et al.*, 2000], qui est la recommandation du W3C pour représenter les métadonnées, OIL [Fensel *et al.*, 2001], DAML+OIL, OWL qui est la recommandation du W3C pour représenter des ontologies. Ajoutons à cette liste SKOS, un langage plus récent, qui est appelé à jouer un rôle important pour la représentation en particulier des terminologies. Nous allons rapidement dans la section suivante faire une revue des langages RDF/RDF(S) [Lassila et Swick, 1999], OWL et SKOS [Miles et Brickley, 2005], utilisés dans la cadre du Web Sémantique.

1.5.1 RDF et RDF(S)

RDF (Resource Description Framework) [Lassila et Swick, 1999] est une recommandation du W3C pour décrire des ressoures. C'est un modèle de graphe pour décrire les (méta-)données en permettant leur traitement automatisé. A l'origine, il a été défini pour décrire des ressources du Web telles que les pages Web ; cependant une ressource peut être toute chose ayant une identité

(objet physique, concept abstrait, etc.). RDF décrit les ressources en exprimant des propriétés et en leur attribuant des valeurs. Il utilise pour cela le vocabulaire défini par RDF Schema noté RDF(S)[Lassila et Swick, 1999].

RDF(S) fournit un ensemble de primitives simples, mais puissantes, pour la structuration de la connaissance d'un domaine en **classes** et **sous-classes**, **propriétés** et **sous-propriétés** – avec la possibilité de restreindre leur domaine d'origine (**rdf :domain**) et leur domaine d'arrivée (**rdf :range**).

L'élément de base d'un document RDF est la *ressource*, correspondant à la représentation conceptuelle d'une entité. Une ressource est identifiée par son URI (Uniform Resource Identifier), de la forme *http ://uri/du/document#courant*. Un document structuré en RDF est un ensemble de triplets. Un triplet RDF est une association < sujet, objet, prédicat > appelée assertion. Par exemple, le sujet peut être une page Web donnée, l'objet, une propriété de cette page comme son titre, et le prédicat, la valeur de cette propriété. Une des syntaxes (sérialisation) de ce langage (en plus de la forme graphique) est RDF/XML dont nous fournissons un exemple au listing 1.1.

Listing 1.1 – Exemple d'un document RDF décrivant la ressource Ora Lassila

```
<rdf:RDF
   xmlns: rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:s="http://description.org/schema/">
   <rdf:Description about="http://www.w3.org/Home/Lassila">
      <s:Creator rdf:resource="http://www.w3.org/staffId/85740"/>
   </rdf:Description>

   <rdf:Description about="http://www.w3.org/staffId/85740">
      <rdf:type resource="http://description.org/schema/Person"/>
      <v:Name>Ora Lassila</v:Name>
      <v:Email>lassila@w3.org</v:Email>
   </rdf:Description>
</rdf:RDF>
```

RDF(S) est indiqué pour la description de ressources, cependant il présente assez rapidement des limites lorsqu'il s'agit de l'utilisation comme langage de représentation d'ontologies ayant de fortes contraintes [McBride, 2004] :

1. la portée locale des propriétés : **rdf :range** permet de définir les domaines de valeurs (range) des propriétés. Cependant, il n'est pas possible avec RDF(S) de limiter leur application à un certain nombre de classes seulement. Par exemple, pour la propriété *mange* de la classe *Etre Vivant*, on peut lui assigner comme domaine de valeur *nourriture*, mais il est par la suite impossible de déclarer que certains êtres vivants mangent des plantes et que d'autres mangent de la viande ;
2. l'expression de la disjonction de classes : il arrive souvent que l'on veuille déclarer que deux classes sont disjointes (l'intersection de leurs instances est vide). Par exemple, les classes *Homme* et *Femme* sont disjointes. Dans RDF(S), il n'est pas possible de l'exprimer ; on ne peut définir que des relations de type "sous-classes" : (e.g., *Femme* est sous-classe de *Humain*) ;
3. l'expression de la combinaison booléenne de classes : on aimerait parfois construire des classes à partir de la combinaison d'autres classes en utilisant les opérateur d'Union,

d'Intersection et de Complément. Par exemple on peut vouloir définir la classe *Humain* comme l'union disjonctive des classes *Femme* et *Homme*. RDF(S) ne permet pas de le faire ;

4. l'expression de la restriction de cardinalités : parfois on voudrait définir le nombre exact de valeurs qu'une propriété donnée peut avoir. Par exemple, on peut vouloir dire qu'une personne a exactement deux parents ou qu'un cours est enseigné par au moins un professeur. Ce genre de restriction est impossible à définir dans RDF(S) ;
5. l'expression de certaines caractéristiques de propriétés : on aimerait exprimer les caractéristiques de certaines propriétés telle que la transitivité (e.g., un grand-parent est le parent d'un parent), l'inverse (*Est-Parent* est l'inverse de *Est-Enfant*), etc. De telles caractéristiques sont impossibles à exprimer avec RDF(S).

1.5.2 OWL

RDF/RDF(S) ne permet pas d'exprimer certaines notions que l'on voudrait décrire avec les ontologies. Dans le but d'étendre l'expressivité de RDF Schema, le groupe de travail sur les ontologies du W3C a proposé le langage d'ontologie du Web (OWL), basé sur le langage DAML+OIL [McGuinness *et al.*, 2002]. La sémantique du langage (ou plus exactement celle de OWL DL et OWL Lite) peut être définie via une transformation en Logiques de Description (DL), dont ce langage est inspiré. On peut utiliser les outils de raisonnement développés pour les DL afin d'effectuer des inférences sur OWL.

Le groupe de travail sur les ontologies du W3C, dans le souci de répondre aux critères exigés pour un langage de représentation d'ontologies, a défini trois différents sous-langages de OWL :

OWL Full est le langage complet. Il utilise tous les éléments disponibles en OWL. Il permet aussi de combiner de façon quelconque ces éléments avec RDF et RDF Schema. OWL Full est totalement compatible avec RDF ; ainsi tout document RDF valide est aussi un document OWL Full valide. Son inconvénient principal vient de son pouvoir d'expression élevé, qui le rend indécidable.

OWL DL (OWL Description Logic) offre une plus grande expressivité que OWL Lite et est décidable. Il est ainsi appelé à cause de son origine, le formalisme de représentation des connaissances appelé Logiques de Description (il est basé sur la DL *SHOIN(D)*)[Nardi et Brachman, 2003]. Une ontologie OWL DL peut être vue comme une TBox dans les DL, avec une relation hiérarchique décrivant le domaine en termes de classes (correspondant aux concepts), et des propriétés (correspondant aux relations). Une ontologie consiste donc en un ensemble d'axiomes qui expriment les relations (par exemple la relation de subsumption) entre les classes ou les propriétés. Chaque élément de l'ontologie appartient à une seule catégorie, c'est-à-dire qu'un même objet ne peut être défini à la fois comme une classe et comme une propriété. OWL DL a l'avantage de fournir un support pour les inférences, mais la compatibilité totale avec RDF/RDF(S) est perdue.

OWL Lite permet de représenter des classifications sous forme hiérarchique et d'exprimer des contraintes simples (contraintes de cardinalité de type 0 ou 1). La disjonction de classes, la définition de classes à partir d'une union de classes, et bien d'autres possibilités offertes par OWL DL, ne sont pas autorisées. Ce langage est particulièrement indiqué pour représenter des taxonomies et autres thésaurus.

Il y a une stricte compatibilité ascendante de ces trois langages : toute ontologie OWL Lite valide est une ontologie OWL DL valide, et toute ontologie OWL DL valide est une ontologie OWL Full valide. Ainsi, toute conclusion d'une ontologie OWL Lite est une conclusion valide de OWL DL, et toute conclusion OWL DL est une conclusion valide de OWL Full.

1.5.3 SKOS

SKOS (Simple Knowledge Organisation System)[7] désigne une famille de langages formels permettant une représentation standard des thésaurus, classifications ou tout autre type de vocabulaire contrôlé et structuré. SKOS est construit sur la base du langage RDF, dont il est une application, et son principal objectif est de permettre la publication facile de vocabulaires structurés pour leur utilisation dans le cadre du Web sémantique. SKOS est actuellement développé dans le cadre du W3C et n'a pas encore le statut de *Recommendation Candidate*. Les principaux documents publiés, tel que le Guide SKOS Core[8], ont le statut de *W3C Working Draft*.

Les différents composants de SKOS sont :

1. **le noyau SKOS** (SKOS core), qui définit les classes et propriétés suffisantes à la représentation des thésaurus standards. Les objets primitifs ne sont pas des termes, mais des concepts dont les termes sont des propriétés. Un *concept SKOS* est défini comme une ressource RDF, identifiée par une URI. Il peut avoir des propriétés : un terme préféré par langue, des synonymes, des concepts reliés, etc.
2. **le SKOS Mapping**, qui définit un vocabulaire pour exprimer des correspondances (alignements exacts ou correspondances approximatives) entre concepts provenant de schémas différents ;
3. **les extensions de SKOS**, qui devront permettre de déclarer des relations entre concepts avec une sémantique plus précise que les relations définies dans SKOS Core, par exemple des relations tout-partie ou classe-instance, ou de préciser certains attributs (note éditoriale, note historique, acronyme, etc.).

Certains vocabulaires ont été mis au format SKOS et sont disponibles pour le public, notamment les thésaurus multilingues européens AGROVOC[9] et GEMET[10].

1.5.4 Le langage du projet OBO

Le projet OBO (Open Biomedical Ontologies)[11] est une initiative d'un groupe de développeurs d'ontologies dans le domaine biomédical, qui s'est mis d'accord sur un nombre de principes spécifiant les bonnes pratiques pour le développement d'ontologies biomédicales. Les principes édictés reposent sur l'objectif d'interopérabilité entre les différentes ontologies développées. Un langage formel commun[12] est fourni pour la représentation des ontologies. Il est conçu pour permettre la prise en compte de plusieurs méta-données, et comprend un mécanisme d'historisation. Parmi les principes édictés par le projet OBO citons :

1. chaque ontologie doit avoir un identifiant unique au sein de OBO ;
2. chaque ontologie doit inclure des définitions textuelles pour chacun de ses termes ;

[7] http ://www.w3.org/2004/02/skos/
[8] http ://www.w3.org/TR/swbp-skos-core-guide/
[9] www.fao.org/agrovoc
[10] www.eionet.europa.eu/gemet
[11] http ://obofoundry.org/
[12] disponible à l'adresse http ://obo.sf.net/

3. les ontologies au sein de OBO doivent être développées de façon collaborative.

Comme toutes les ontologies décrites à l'aide des langages à base de DLs, tels que OWL, une ontologie OBO comprend des classes, des propriétés (ou types de relations), des instances. La notion de classe est similaire à celle qu'on retrouve en OWL et DAML+OIL. Il existe deux niveaux de "Propriétés" : les propriétés à l'échelle des classes et celles à l'échelle des instances. Des travaux sont en cours pour la représentation dans le format OBO des ontologies telles que la Gene Ontology[13] et la FMA (voir section 1.8).

1.6 Construction et édition d'ontologies

La construction d'une ontologie n'est pas une activité aisée, d'autant plus qu'il n'existe pas aujourd'hui une méthodologie communément admise, comme c'est le cas dans le domaine des bases de données par exemple, avec la démarche Entité/Association. Le processus de construction intègre souvent un expert du domaine. Une fois construite, l'ontologie doit être claire, cohérente, compréhensible, facile à utiliser et extensible [Guarino, 1998b].

Les travaux sur la construction des ontologies ont débuté dans les années 1990 [Gruber, 1995][Uschold et King, 1995] [Fernandez *et al.*, 1997] [Guarino, 1998b][Corcho *et al.*, 2003][Jarrar et Meersman, 2002] [Aussenac-Gilles *et al.*, 2000][Grüninger et Fox, 1995][Blazquez *et al.*, 1998]. Nous pouvons les classer comme suit [Lopez, 2001] :

1. les méthodes et méthodologies pour la construction d'ontologies en partant de zéro ;
2. les méthodes pour la réingénierire d'ontologies ;
3. les méthodes de construction coopérative d'ontologies.

Nous faisons un rapide survol de chacune de ces méthodes, tout en nous attardant sur la première, qui, d'une part, est d'un intérêt particulier pour nous dans le cadre de l'acquisition d'ontologies sur le cerveau (voir chapitre 5) et, d'autre part, englobe les méthodes d'acquisition d'ontologies à partir de textes [Biebow et Szulman, 1999][Cimiano et Voelker, 2005]. Pour une description plus détaillée des méthodes et outils sur la construction d'ontologies, nous renvoyons le lecteur à l'état de l'art fait par Corcho et ses collègues [Corcho *et al.*, 2003] et par Murshed et Ali. Corcho et ses collègues font une revue des principaux outils et langages ainsi que des méthodologies pour construire des ontologies. Murshed et Ali [ul Murshed et Ali, 2005] proposent des critères d'évaluation et des techniques de classement permettant aux développeurs d'ontologies de choisir le bon outil de construction.

1.6.1 Les méthodes et méthodologies pour la construction d'ontologies en partant de zéro

Dans cette catégorie, la première a été la méthodologie CYC [Lenat et Guha, 1990], élaborée dans le cadre du développement de l'ontologie CYC. Cette méthodologie propose deux étapes de construction : i) l'extraction manuelle de la connaissance commune implicite dans les différentes sources ; ii) l'utilisation des techniques de Traitement Automatique de la Langue Naturelle (TALN) et d'acquisition de connaissances pour générer de nouvelles connaissances à partir de celles acquises à l'étape précédente ;

L'approche de Uschold et King [Uschold et King, 1995], utilisée lors du développement de l'ontologie pour la modélisation des processus dans l'entreprise, appelée *Enterprise Ontology*,

[13] http ://www.geneontology.org/

et celle de Grüninger et Fox [Grüninger et Fox, 1995] utilisée dans le cadre du projet TOVE (Toronto Virtual Entreprise Ontology), ont suivi la méthodologie CYC.

Methontology [Fernandez *et al.*, 1997] se distingue des méthodes précédentes puisqu'elle peut être utilisée à la fois pour la construction à partir de zéro et pour la réingénierire. L'approche, qui préconise la construction d'ontologies à un niveau conceptuel, inclut les étapes suivantes : l'identification des principales activités du processus de développement d'ontologies (c'est à dire l'évaluation, la conceptualisation, l'intégration, l'implémentation, etc.) ; un cycle de vie basé sur des prototypes évolutifs ; enfin la méthodologie elle même, qui spécifie les étapes à considérer pour l'accomplissement de chaque activité, les techniques utilisées, les résultats produits et la manière de les évaluer.

Guarino [Guarino, 1998b] s'est inspiré de recherches philosophiques pour proposer une méthodologie de construction d'ontologies appelée " Formal Ontology ". Les principes de cette méthodologie incluent des notions comme la théorie des parties, la théorie des tous (theory of wholes), la théorie de l'identité, la théorie de la dépendance et la théorie d'universalité. Les principes de base incluent notamment 1) une clarification du domaine concerné, 2) la prise en compte sérieuse de la notion de l'identité, 3) l'isolement d'une structure taxonomique de base et 4) enfin l'identification explicite des relations (rôles).

Jarrar et ses collègues [Jarrar et Meersman, 2002] décrivent une approche inspirée du domaine des bases de données, appelée DOGMA, pour la conception d'ontologies formelles. Cette approche prend en compte plusieurs considérations : la réutilisation et le partage de connaissances, le stockage et la gestion des ontologies de façon efficace, la coexistence de systèmes hétérogènes utilisant la même ontologie, etc. Une des idées clés de cette approche est d'aboutir à une ontologie représentant la sémantique du domaine indépendamment de la langue. Deux niveaux de décomposition des ressources ontologiques sont définis : les faits binaires appelés lexons d'une part (ontology base), et les contraintes et règles de description d'autre part (ontology commitments).

La connaissance de toute organisation se retrouve codifiée dans les objets qu'elle manipule, parmi lesquels figurent en bonne place les documents textuels. C'est pourquoi sont apparues des méthodes de construction (semi-automatique) d'ontologies à partir de textes [Biebow et Szulman, 1999][Maedche et Staab, 2000][Fortuna *et al.*, 2006]. L'idée générale consiste en l'identification, à partir de documents, des concepts importants et des relations hiérarchiques et associatives entre ces concepts grâce à l'utilisation d'outils de TALN, d'extraction d'informations et parfois de ressources externes de connaissances. Nous décrivons dans la section suivante l'outil TERMINAE [Biebow et Szulman, 1999][Aussenac-Gilles *et al.*, 2000] qui permet la construction d'ontologies à partir de textes.

Un exemple d'outil de construction d'ontologies à partir de textes

TERMINAE [Biebow et Szulman, 1999] est un outil qui intègre les phases de conception et d'édition d'ontologies. C'est un outil d'aide à la construction d'ontologies à partir de textes, dont le développement a commencé en 1997. Une approche linguistique a été préconisée pour TERMINAE car on a constaté que la plupart des objets utilisés pour modéliser un domaine sont dénotés par des termes et sont repérables à l'intérieur des textes. L'outil intègre dans un même environnement des outils d'ingénierie linguistique et d'ingénierie des connaissances : les outils d'ingénierie linguistique permettent d'extraire du corpus l'ensemble des occurrences de différents termes puis de définir les différents sens de chacun de ces termes, appelés notions, tandis que les outils d'ingénierie des connaissances permettent de représenter les notions d'un terme par des concepts et d'insérer ces concepts dans l'ontologie.

Les principales étapes de construction d'une ontologie avec TERMINAE sont la constitution du corpus, l'analyse de ce corpus, l'étude terminologique, la normalisation, la formalisation et l'insertion dans l'ontologie. TERMINAE facilite la tâche de l'utilisateur en le guidant tout au long de l'élaboration, en automatisant certaines tâches contraignantes et en gérant la base de connaissances. La compréhension de la modélisation pour les autres utilisateurs est facilitée grâce à la traçabilité des textes vers l'ontologie au travers de liens entre les textes, les fiches terminologiques, les fiches de modélisation et les concepts terminologiques.

Complètons la liste non exhaustive des méthodes de construction à partir de zéro par les propositions de Bernaras et ses collègues [Bernaras *et al.*, 1996], qui présentent une méthode utilisée dans le cadre du projet européen KACTUS pour construire une ontologie dans le domaine des réseaux électriques, de Noy et McGuinness [Noy et McGuinness, 2001] pour la construction d'ontologies basées sur les systèmes déclaratifs de représentation de connaissances, de Roche [Roche, 2001] qui propose le principe de spécificité/différence pour la construction d'ontologies, de Dong et Mingshu pour l'approche TEMPPLE, pour "TEMPlate + ApPLET » [Dong et Li, 2002] pour la construction d'ontologies de domaines.

1.6.2 Les méthodes pour la réingénierie d'ontologies

La réingénierie d'ontologies est le processus qui consiste à reconstruire et lier un modèle conceptuel d'une ontologie déjà implémentée à une autre en cours d'implémentation [InterOp, 2006]. Une méthode représentative de cette approche est celle de Gomez-Perez et Roza [Gomez-Perez et Rojas, 1999], qui ont adapté une technique de réingénierire de schémas à une ontologie de domaine.

1.6.3 Les méthodes pour la construction coopérative d'ontologies

Les méthodes de cette catégorie prennent acte du fait qu'une ontologie doit faire l'objet d'un consensus et être acceptée par sa communauté d'utilisateurs. Elles adoptent donc une approche collaborative de construction incluant l'intervention de personnes localisées à des endroits différents. Euzenat [Euzenat, 1995][Euzenat, 1996] a identifié un certain nombre de problèmes que pose la construction d'ontologies dans un contexte distribué : la gestion de l'interaction et la communication entre les différentes personnes ; le contrôle de l'accès aux données ; la reconnaissance de l'attribution des droits d'auteurs ; le détection et la correction d'erreurs. Comme exemple de ces méthodes citons celle de Farquhar et ses collègues [Farquhar *et al.*, 1995], utilisée pour la construction d'ontologies pour l'intégration de données, CO4 de Euzenat [Euzenat, 1996], utilisée à l'INRIA pour la construction de bases de connaissances, et enfin la méthode élaborée dans le cadre de l'initiative KA [Benjamins et Fensel, 1998], utilisée pour la construction d'ontologies pour l'annotation sémantique de documents.

Notons que pour assister l'activité de construction des ontologies, des outils ont été développés dans le cadre de certaines des méthodes présentées ci-dessus (e.g., TERMINAE, l'outil ODE – Ontology Developpement Environnement – qui applique la méthode Methontology au développement d'ontologies [Blazquez *et al.*, 1998]).

1.6.4 Les outils d'édition d'ontologies

Les outils d'édition constituent une aide pour l'implémentation d'une ontologie dans laquelle les principaux choix ont déjà été faits. Les outils peuvent se subdiviser en deux groupes : les outils dont le formalisme est proche des logiques de description et ceux dont le formalisme est proche des frames. Dans la première catégorie, le plus connu est OilEd [Bechhofer *et al.*, 2001]

l'éditeur de DAML + OIL [McGuinness *et al.*, 2002] tandis que OntoEdit [Sure *et al.*, 2002] et Protégé-2000 [Gennari *et al.*, 2003] peuvent être classés dans la seconde catégorie. Une description détaillée des outils de développement d'ontologies peut être trouvée dans l'état de l'art fait dans le cadre du projet européen OntoWeb[14]. Nous faisons un rapide survol des outils cités ci-dessus, qui sont utilisés dans le cadre du Web Sémantique.

Protégé

Protégé [Gennari *et al.*, 2003], développé à l'Université de Stanford[15], est sans aucun doute l'environnement d'édition d'ontologies le plus utilisé aujourd'hui. Son noyau est basé sur le modèle des frames [Minsky, 1975] mais les ontologies devéloppées avec cet environnement peuvent être exportées dans plusieurs formats (RDF(S), OWL, XML Schema).

L'environnement d'édition offre deux possibilités de construction d'ontologies :

1. avec l'éditeur à base de frames, il permet la construction et l'instanciation d'ontologies basées sur le modèle des frames, en conformité avec le protocole OKBC [Chaudhri *et al.*, 1998] ; dans ce modèle, une ontologie est composée d'un ensemble de classes, organisé sous la forme d'une hiérarchie de subsumption ; d'un ensemble de slots associés aux classes, décrivant leurs propriétés et relations ; et enfin d'un ensemble d'instances des classes définies.
2. avec l'éditeur OWL, il permet la construction d'ontologies pour la Web Sémantique et particulièrement la construction d'ontologies OWL (voir figure 1.1). Il offre avec cet éditeur tous les outils nécessaires pour l'édition des différents éléments d'une ontologie OWL (concepts, propriétés, instances), avec la possibilité de spécifier des contraintes et d'utiliser des moteurs d'inférence externes tels que Racer [Haarslev et Möller, 2001] ou Pellet [Sirin et Parsia, 2004] pour vérifier la consistance de l'ontologie et d'inférer de nouvelles connaissances.

Protégé s'enrichit régulièrement de l'apport de la communauté des utilisateurs et développeurs grâce au système de plugins qui permettent de rajouter de nouvelles fonctionnalités à l'outil. L'outil permet d'intégrer plusieurs ontologies et de gérer les différentes versions d'une même ontologie.

OilEd

OIlEd est un outil développé à l'Université de Manchester, spécialement pour l'édition d'ontologies OIL et DAML+OIL, basées sur le formalisme des DLs. L'interface principale de OILEd est composé d'un ensemble de tables fournissant différentes options pour éditer les différents composants d'une ontologie DAML+OIL : classes, propriétés, individus (instances), axiomes, espaces de nommage.

La classement des concepts et des instances s'effectue en utilisant le moteur d'inférence par défaut, FaCT [Horrocks, 1998]. Pour cela l'ontologie DAML+OIL est transformée dans le formalisme des DL *SHIQ* [Horrocks et Sattler, 2003].

OILEd n'est pas un environnement complet d'édition d'ontologies. Ainsi, il n'est pas destiné à l'édition de grandes ontologies, ni à l'intégration ou la gestion de plusieurs versions d'une ontologie.

[14] http ://www.ontoweb.org/
[15] http ://protege.stanford.edu/

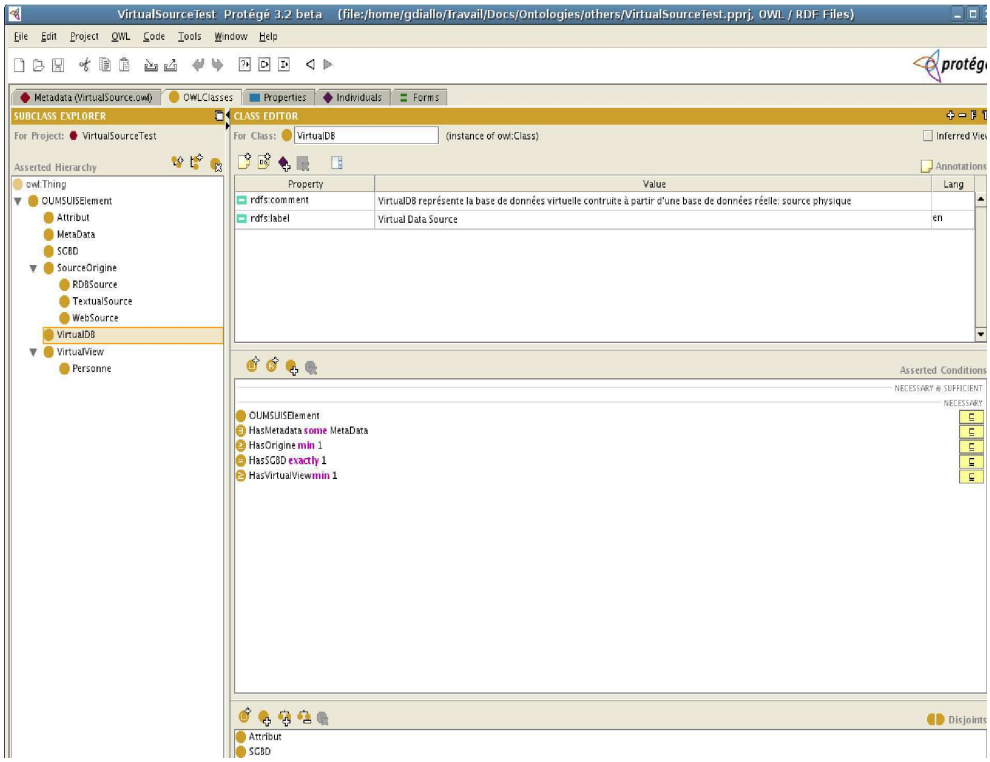


FIG. 1.1 – Interface d'édition de classes d'une ontologie sous Protégé

OntoEdit

OntoEdit [Sure *et al.*, 2002] est un outil de construction d'ontologies développé au Knowledge Management Group (AIFB) de l'Université de Karlsruhe. Il fournit un environnement graphique d'édition d'ontologies permettant l'inspection, la navigation, le codage et la modification d'une ontologie. De nouvelles fonctionnalités peuvent être ajoutées à l'outil grâce a de nouveaux plugins. Le modèle conceptuel de l'ontologie à éditer est stocké en interne suivant un modèle riche qui permet un export dans différents langages de représentation d'ontologies (XML, Flogic, RDF(S), DAML+OIL). OntoEdit, dans sa version commerciale (permettant le stockage de l'ontologie dans une base de données relationnelle), fait partie de la suite logicielle proposée par la société Ontoprise[16].

Une fois construite, l'ontologie est appelée à évoluer. En effet, elle peut ne plus, à un moment donné, refléter le domaine qu'elle est censée modéliser, compte tenu de plusieurs facteurs de changement comme l'évolution métier de l'organisation.

1.7 Enrichissement, maintenance et évolution d'ontologies

Une fois construite, l'ontologie est appelée à évoluer afin de prendre en compte les changements induits par les nouveaux besoins qui peuvent apparaître. Le changement peut concerner la structure conceptuelle, l'ajout de nouvelles définitions, de nouveaux termes, etc. L'enrichissement d'une ontologie par de nouveaux termes pour désigner les concepts est particulièrement souhaité dans le cadre de l'utilisation d'ontologies pour la recherche d'information [Simonet *et al.*, 2006].

[16] http ://www.ontoprise.de/content/index_eng.html

Cet enrichissement peut se faire à partir d'un corpus spécialisé ou à partir de sources de vocabulaires existants (e.g., UMLS dans le domaine biomédical). Les opérations d'enrichissement et de maintenance, qui font partie de l'activité globale de gestion de l'évolution, induisent des changements de l'ontologie qui impliquent des mécanismes de gestion de versions. Klein [Klein, 2002] distingue les **changements conceptuels** (la manière dont le domaine est interprété) des **changements explicatifs** (la manière dont les concepts sont spécifiés).

Deux approches sont utilisées pour la gestion de versions : l'approche basée sur les traces de changements et l'approche basée sur la comparaison de version.

Approches basées sur les traces

La première approche pour la gestion des changements préconise l'utilisation des traces, soit au niveau macroscopique, soit au niveau microscopique de l'ontologie (e.g., les triplets RDF). Oliver et ses collègues [Oliver *et al.*, 1999] proposent une approche basée sur les traces de changement à travers un modèle de concept appelé CONCORDIA. Un certain nombre de changements et de mécanismes de synchronisation basés sur ces changements sont spécifiés. Stojanovic et ses collègues [Stojanovic *et al.*, 2002] quant à eux introduisent la notion de *stratégie d'évolution*, qui permet aux développeurs de spécifier des effets de changements complexes dans le cadre de la suite logicielle KAON[17], dédiée à la gestion d'ontologies. Les logs des différents changements sont utilisés par KAON comme information de gestion des versions. Ognyanov [Ognyanov et Kiryakov, 2002] utilise la même approche des traces pour gérer les changements dans les bases RDF. L'élément de base qu'utilise Ognyanov étant le triplet, le niveau de granularité des changements par rapport aux deux méthodes précédentes n'est pas le même.

Approches basées sur la comparaison de versions

La seconde approche utilisée pour gérer les changements dans une ontologie est la comparaison des différentes versions d'une même ontologie. Les outils d'édition d'ontologies Protégé et OntoView [Klein, 2002] utilisent ce principe.

Protégé offre la possibilité de gérer les versions d'une ontologie à travers l'archivage de version offerte par l'outil. Par ailleurs, il intègre un plugin appelé PROMPT [Noy et Musen, 2004] permettant de comparer, entre autres, l'ontologie courante avec ses différentes versions.

OntoView est un outil de gestion de version d'ontologies inspiré par le système de contrôle de version CVS. L'approche suivie est basée sur la comparaison de deux versions d'une même ontologie en vue de détecter d'éventuelles différences. La détection est basée sur la comparaison des triplets RDF décrivant les classes et les propriétés des deux versions de l'ontologie à comparer.

1.8 Exemples d'ontologies et de classifications dans le domaine biomédical

Le foisonnement des classifications médicales illustre la nécessité pour une communauté scientifique de disposer d'un vocabulaire de consensus. La multiplicité de ces classifications, particulièrement notable dans le domaine médical, illustre aussi le fait que ces besoins se sont manifestés dans différentes communautés, qui y ont répondu d'une manière opportuniste, en fonction de leurs moyens et de leurs besoins. On s'est ainsi trouvé en présence de plusieurs classifications, répondant chacune à une catégorie de besoins, mais présentant des structurations différentes,

[17] http ://kaon.semanticweb.org/

bien que les concepts qu'elles définissent présentent un taux de recouvrement important. Le développement des outils informatiques, les bases de données en particulier, a mis en évidence les difficultés de communication et d'échange ainsi créées au sein de la communauté médicale : une même maladie, un même symptôme, sont codés de manière différente ; de plus, il n'y a pas toujours identité stricte entre les "concepts" représentés par les termes utilisés dans ces différentes classifications.

Pour tenter de remédier à cette situation, la NLM (National Library of Medicine) a pris en 1986 l'initiative du projet UMLS (Unified medical Language System) [Bodenreider, 2004]. L'objectif d'UMLS est d'améliorer la capacité des programmes informatiques à "comprendre la signification" des requêtes des utilisateurs dans le domaine biomédical et d'utiliser cette compréhension pour la recherche et l'intégration d'informations compréhensibles par les machines.

Quelques autres initiatives sont significatives : le projet GALEN (Generalised Architecture for Languages, Encyclopaedias and Nomenclatures in Medicine) pour le développement de systèmes multilingues de terminologies médicales, le projet Digital Anatomist pour la conceptualisation des objets physiques et spatiaux qui constituent le corps humain à une échelle macroscopique [Neal *et al.*, 1998], le thésaurus MeSH, utilisé pour l'indexation de ressources médicales en ligne (MEDLINE), et SNOMED CT[18], un vocabulaire contrôlé utilisé dans le domaine clinique.

Ces ressources présentant un intérêt certain pour notre travail de construction d'ontologies dans le domaine du cerveau, nous faisons ici un rapide survol de chacune d'elles.

1.8.1 UMLS

Le Coeur d'UMLS est constitué d'un "**metathesaurus**" qui comporte plus de 1.300.000 concepts dans sa version 2006 (version 2006AC), et qui croît régulièrement de près de 200.000 concepts par an. C'est la partie la plus évolutive d'UMLS, et le rythme de sa croissance pose la question de sa maîtrise et de sa cohérence. UMLS regroupe une centaine de sources de vocabulaires médicaux (139 dans la version 2006), dont la NBH (Neuronames Brain Hierarchy), qui est d'un intérêt particulier pour nous.

Chaque concept UMLS a un identifiant unique, le CUI (Concept Unique Identifier) : ex. C0030560 pour le concept désigné par *Parietal Lobe* en anglais. A chaque concept est associé un ensemble de termes dans différents lexiques. La langue principale est l'anglais et les autres langues sont pour l'instant peu représentées, à l'exception de l'espagnol. Chaque CUI a dans chaque langue un terme préféré unique appelé SUI (String Unique Identifier). Chaque SUI est lié à un ou plusieurs termes selon ses différentes variations lexicales, qui sont les LUI (Lexique Unique Identifier). Les SUI dans les différentes langues sont nécessaires pour la communication, mais le vrai identifiant du concept est son CUI.

Un concept a aussi un **type sémantique**. Les types sémantiques sont regroupés au sein d'un **réseau sémantique** (semantic network) de 135 types sémantiques organisés de manière arborescente et de 54 relations sémantiques. Les données lexicales (déclinaisons d'un terme, pluriel ou singulier du terme, différentes syntaxes, etc.) sont regroupées dans le **specialist lexicon** (support lexical). Ces trois structures (le réseau sémantique, le métathesaurus et le support lexical) sont les constituants principaux d'UMLS. En pratique, ils sont stockés sous forme de tables relationnelles. Il est possible d'interroger à distance UMLS à l'aide de certaines API fournies. Nous avons exploité cette possibilité lors de la construction de l'ontologie anatomique du cerveau (Cf. chapitre 5).

Même si UMLS constitue aujourd'hui une source terminologique très utilisée, certaines imper-

[18] http ://www.snomed.org/

fections sont à relever. Gangemi et ses collègues [Pisanelli *et al.*, 1998] notent certains problèmes de cycles, qui peuvent être causés par le recouvrement partiel de certains concepts.

1.8.2 SNOMED CT

SNOMED-CT (Systematized Nomenclature of Medicine, Clinical Terms)[19] est une source de vocabulaire produite par le College of American Pathologists, qui a une large couverture du domaine clinique. Elle combine la SNOMED Reference Terminology (SNOMED RT) et la Version 3 de la United Kingdom's Clinical Terms (anciennement connue sous le nom de *Read Codes*). Son usage se retrouve dans plusieurs contextes : l'indexation de documents cliniques, l'aide à la décision clinique, l'indexation d'images, ect. La terminologie est organisée en une hiérarchie de 18 catégories de premier niveau : les procédures, les entités observables, les structures du corps humain, les événements, etc. Ces entités, dites **majeures**, sont regroupées autour d'une racine appelée **Top**. Dans sa version de Juillet 2006, SNOMED CT totalise plus de 300.000 concepts et 770.000 descriptions en aglais.

1.8.3 MeSH

Le thésaurus MeSH (Medical Subject Headings) est un outil réalisé par la National Library of Medicine (NLM). Il est utilisée pour l'indexation et la recherche d'informations médicales. La première version est apparue en 1954 sous le nom de Subject Heading Authority List. Sa parution sous le nom de Medical Subject Headings date de 1963 et il contenait, dans cette édition, 5.700 descripteurs. Face au nombre sans cesse croissant de ressources médicales à gérer par les documentalistes médicaux, la NLM lança à cette période le projet MEDLARS (voir le chapitre 3 consacré à la RI) pour l'automatisation de l'indexation et la recherche de ressources médicales. Depuis, le MeSH n'a cessé d'évoluer. Il compte 23.885 descripteurs dans sa version 2006.

Les descripteurs MeSH sont organisés en 16 catégories : la catégorie A pour les termes anatomiques, la catégorie B pour les organismes, la catégorie C pour les maladies, etc. Chaque catégorie est subdivisée en sous-catégories. A l'intérieur de chaque catégorie, les descripteurs sont structurés hiérarchiquement, du plus général au plus spécifique, avec un niveau de profondeur maximum de 11.

Le thésaurus MeSH sert de vocabulaire contrôlé pour l'indexation des ressources de la base de données bibliographique MEDLINE[20]. Il est aussi utilisé aussi par les portails d'indexation et de cataloguage de ressources médicales, Health On the Net [21] et CisMef[22]. L'INSERM maintient une version Française du MeSH.

1.8.4 Galen

GALEN est une initiative européenne pour le développement de systèmes multilingues de terminologies médicales. Galen est basée sur un modèle sémantique puissant de terminologie clinique, le Galen Coding Reference (CORE), et utilise un langage de représentation appelé GRAIL (GALEN Représentation And Integration Language) dont le noyau est basé sur les logiques de description. Ce modèle consiste en une hiérarchie de subsomption d'entités élémentaires

[19] http ://www.snomed.org/snomedct/index.html
[20] accèsssible grâce au moteur Pubmed à l'adresse `www.ncbi.nlm.nih.gov/entrez/`
[21] http ://www.hon.ch
[22] www.chu-rouen.fr/cismef/

(concepts primitifs des DL) et un ensemble de déclarations liant ces entités. Le modèle terminologique permet une définition explicite des concepts dans le domaine médical. Chaque concept dans ce modèle doit avoir au moins un parent. Ce modèle est par ailleurs qualifié d'ontologie [Rector *et al.*, 1996].

Comme toute ontologie basée sur les DLs, une ontologie Galen est composée d'une hiérarchie de catégories élémentaires, une hiérarchie de liens sémantiques appelés **attributs** (aussi désignés par **rôles** ou **relations**) pour établir des relations entre les catégories. Les concepts peuvent êtres simples ou composés. Les concepts composés sont une combinaison de concepts simples. Par ailleurs, on peut définir un ensemble d'axiomes sur les concepts, ainsi que des contraintes. La figure suivante présente la Top Level Ontology de Galen. La distinction est faite à partir de DomainCategory (dans GALEN les concepts sont des **catégories**) entre les premières classes d'entités ou **Things** (partie gauche) et toutes les autres.

1.8.5 Digital Anatomist : Foundational Model of Anatomy (FMA)

Le projet Digital Anatomist de l'université de Washington, soutenu par la NLM (National Library of Medicine), a pour objectif la conceptualisation des objets physiques et spatiaux qui constituent le corps humain à une échelle macroscopique. Une ontologie anatomique du corps humain a été construite en définissant les classes et les sous-classes des entités anatomiques physiques selon leurs relations partonomiques et spatiales [Neal *et al.*, 1998][Rosse et Mejino, 2003].

Toute entité anatomique dans ce modèle peut être affectée à une des trois classes du **Top Level** (Anatomical Structure, Anatomical Spatial Entity et le Body Substance). Le modèle résultant de ce travail est connu sous le nom de The Digital Anatomist Foundational Model, ou **FMA**. La hiérarchie connue sous le nom de AO (Anatomy Ontology) est le premier composant du modèle. Les relations qui spécifient la partonomie et les relations spatiales des structures anatomiques constituent l'ASA (Anatomical Structural Abstraction). Le troisième composant de la FMA est l'ATA (Anatomical Transformation Abstraction), qui est l'instanciation de l'abstraction spatiale de l'anatomie. Le quatrième composant est le Mk (Metaknowledge), qui comprend les principes, les règles et les définitions représentées dans les autres composants. Le modèle fondamental est représenté par l'ensemble (AO, ASA, ATA, Mk).

L'abstraction spatiale de l'anatomie est présentée dans [Neal *et al.*, 1998]. Une conceptualisation de l'anatomie selon l'appartenance spatiale des attributs des entités anatomiques est proposée. Le schéma utilise des classes topologiques. Un objet spatial participe comme noeud dans trois réseaux complémentaires : le réseau topologique, le réseau partonomique (part-of) et le réseau d'associations spatiales. La classification des objets spatiaux se fait suivant la dimension spatiale. Les quatre classes proposées sont le point, la ligne, la surface et le volume représentant les objets spatiaux à zéro, une, deux ou trois dimensions.

1.8.6 Ontologies et terminologies dans le domaine du cerveau

Le cerveau est présenté suivant ses aspects anatomique et fonctionnel. Si le premier aspect a fait l'objet de travaux ayant fourni des résultats bien admis, l'aspect fonctionnel reste peu ou pas exploré, ce qui nous a amenés à faire une proposition de l'ontologie fonctionnelle du cerveau (voir le chapitre 5). La connaissance de ces deux aspects et de leurs liens était un des enjeux du projet BC^3 [Simonet *et al.*, 2003].

La Neuronames Brain Hierarchy (NBH) [Bowden et Dubach, 2005], accessible librement sur le site de BrainInfo[23] de l'Université de Washington, fournit une classification de la partie anatomique du cerveau humain et des macaques. Un de ses objectifs est de fournir une structure d'index de l'anatomie du cerveau pour les bases de connaissances automatisées. Elle a été incorporée comme une des sources de vocabulaires dans UMLS. Elle structure le cerveau avec environ 1.500 structures, représentées par 7.500 termes neuroanatomiques avec plusieurs synonymes en anglais et latin. Les structures sont mutuellement exclusives.

Ontologie du cortex anatomique du cerveau Dameron et ses collègues [Dameron *et al.*, 2003] proposent une ontologie pour le cortex anatomique du cerveau, dont l'objectif est la représentation de l'anatomie du cortex cérébral afin d'expliciter les concepts majeurs, leur relations, ainsi qu'une représentation 3D de la forme et des variations anatomiques.

1.9 Conclusion

Nous avons fait dans ce chapitre une revue des notions liées aux ontologies et au Web Sémantique. Cette revue, bien que n'étant pas exhautive car ce domaine est assez vaste aujourd'hui, introduit les éléments nécessaires qui nous permettent d'aborder les prochains chapitres.

Nous avons décrit quelques langages de représentation des ontologies utilisés dans le cadre du Web sémantique. RDF et RDF(S) ont été conçus suivant le principe de décentralisation du Web en permettant de représenter, d'intégrer et de faire évoluer des vocabulaires distribués. Nous avons vu cependant que RDF(S) ne supportait pas certaines notions à propos des ontologies, comme l'expression de l'équivalence entre concepts, les contraintes de cardinalité, etc. OWL permet d'étendre RDF(S) en fournissant des primitives pour exprimer les notions qu'il n'était pas possible d'exprimer avec RDF(S). Ce langage dispose d'une sémantique formelle et fournit un support pour les inférences grâce à son lien avec les logiques de description.

Avec l'avènement du Web Sémantique et la description sémantique de ressources, les ontologies jouent un rôle pivot. Elles sont utilisées dans plusieurs domaines dont notamment la RI afin de pallier aux insuffisances des modèles basés sur les mots-clés, le commerce électronique (où elles offrent un vocabulaire partagé), la gestion des connaissances (accès intelligent à l'information, partage de connaissances, etc.), etc. Dans le domaine de l'intégration, les ontologies sont utilisées pour expliciter le contenu des sources et fournir un vocabulaire partagé.

[23] http ://braininfo.rprc.washington.edu/

Chapitre 2

Intégration de données

2.1 Introduction

La disponibilité croissante de sources de données variées et dispersées contenant des informations cruciales pour la prise de décision au sein des organisations pose le problème de leur accès de façon efficace. Les systèmes d'information sont conçus pour fonctionner de façon autonome et non pour un besoin d'intégration. Dans la plupart des cas, les sources ont été développées indépendamment et sont par conséquent hétérogènes. Cependant, les organisations ont de nos jours le souci de faire interopérer ces différentes sources afin d'avoir une vision globale de leur système d'information. Pour réaliser cette interopérabilité, un certain nombre d'adaptations et de fonctionnalités supplémentaires sont nécessaires [Ziegler et Dittrich, 2004]. L'hétérogénéité considérée dans les systèmes d'intégration est indépendante de la distribution physique des données. Un système d'information est homogène si le logiciel qui gère les données est le même sur tous les sites, si les données ont un même format et une même structure (même modèle de données) et appartiennent à un même univers de discours [Elmagarmid *et al.*, 1999]. A l'opposé, un système est hétérogène s'il utilise des langages de programmation ou d'interrogation ou des modèles ou des systèmes de gestion de données différents [Solar et Doucet., 2002].

Dans ce chapitre nous décrivons les problèmes que pose l'intégration de plusieurs sources de données, en particulier les sources de données structurées, problèmes qui ont largement été abordés dans la littérature. Nous situons également la place des ontologies dans le processus d'intégration et nous discutons d'un certain nombre de systèmes permettant l'intégration.

2.2 Les différents niveaux d'intégration

Avant d'aborder réellement le problème de l'intégration, situons tout d'abord la place de l'intégration au sein de l'architecture globale d'un système d'information. En effet, le terme a été utilisé dans différents travaux tendant à assurer l'interopérabilité entre différents systèmes, les élements à intégrer pouvant être des applications, des ontologies ou des bases de connaissances, des bases de données, des services web, etc.

Les systèmes d'information d'une organisation peuvent être décrits en utilisant une architecture en couches comme indiqué dans la figure 2.1.

Au niveau de la couche supérieure de l'architecture, les utilisateurs accèdent aux données et aux services à travers différentes interfaces implémentées au-dessus de la couche d'application. Les applications peuvent utiliser des middlewares tel que SQL-middleware pour accéder aux données à travers la couche d'accès aux données. Les données elles-mêmes sont gérées à travers un système

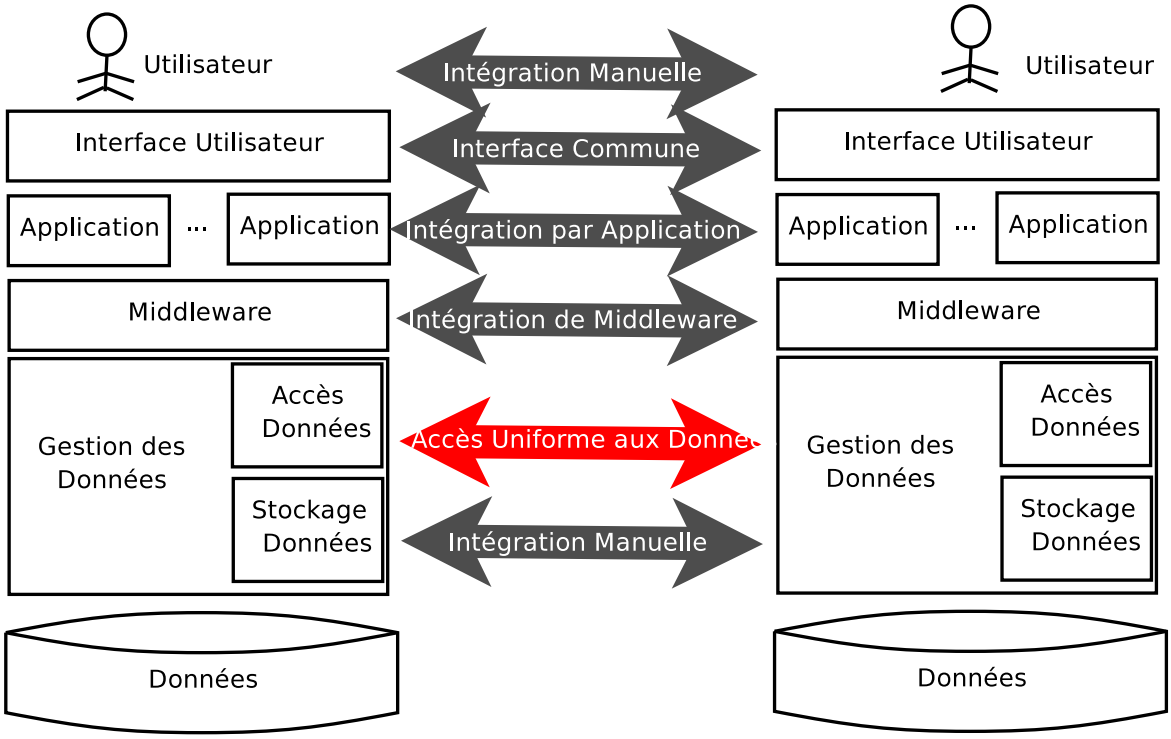


FIG. 2.1 – Les différentes approches d'intégration selon les différentes couches de l'architecture d'un système d'information

de gestion de données (SGBD pour les bases de données, système de fichiers pour les fichiers textes, etc.). A chaque niveau correspond un type d'intégration bien défini. Ce type d'architecture est appelé *n-tier achitecture* ou *achitecture à n couches/niveaux*. Nous nous intéressons ici à l'intégration par un **accès uniforme aux données**.

2.3 Les différentes approches utilisées pour l'intégration de données

Classiquement, deux approches sont utilisées lorsqu'il s'agit d'interroger de façon simultanée plusieurs sources d'information. La première approche préconise la centralisation de toutes les données des sources au niveau d'une **localisation unique**. Les requêtes des utilisateurs sont posées sur cette localisation centralisée. La seconde approche préconise la définition d'un **schéma global virtuel** sur lequel sont posées les requêtes. Un mécanisme de réécriture de requêtes est alors proposé pour traduire les requêtes sur le schéma global en requêtes sur les schémas des sources. Les systèmes qui utilisent ces deux approches sont appelés respectivement **systèmes d'entrepôts de données** et **systèmes d'intégration (ou de médiation) de données**.

2.3.1 Les entrepôts de données

Le but des entrepôts de données est de récupérer et stocker en un même emplacement toutes les données utiles pour une application donnée (Cf. figure 2.2). L'alimentation de l'entrepôt en données se fait généralement en mode batch à travers un module permettant l'extraction,

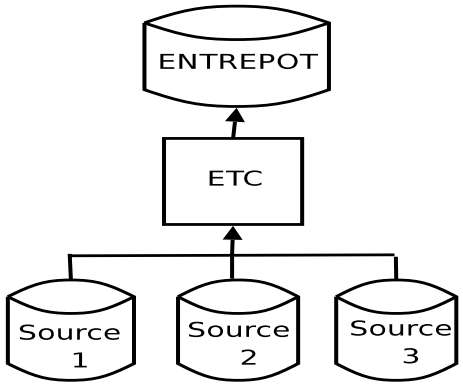


FIG. 2.2 – Architecture d'un entrepôt de données

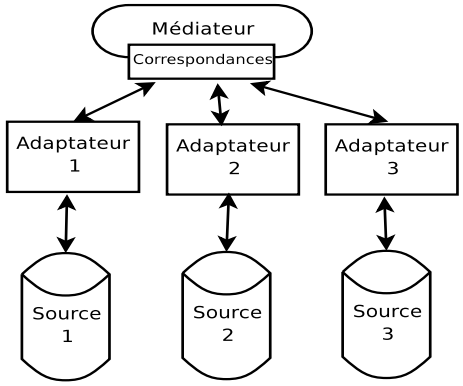


FIG. 2.3 – Architecture générale d'un système d'intégration de données

la transformation et le chargement des données (ETC) [Wu et Buchmann, 1997]. Les requêtes posées sur le schéma de l'entrepôt sont ensuite exécutées directement sur son contenu.

L'approche entrepôt de données présente certains avantages. L'entrepôt simplifie l'optimisation et le traitement de requêtes en triant les données localement en fonction d'un seul schéma global. Les utilisateurs peuvent ajouter leurs propres annotations sur les données stockées et spécifier certaines conditions de filtrage sur ces données. Cependant, l'entrepôt doit être rafraîchi régulièrement. En effet, les données au niveau des sources sont sujettes à des mises-à-jour. L'entrepôt doit également être mis à jour afin de garder une image fidèle des données qu'il représente.

2.3.2 Les systèmes d'intégration de données

L'intégration de données est un domaine de l'informatique qui s'occupe de l'uniformisation de l'accès à une grande quantité de données variées et dispersées. La problématique de ce type de système peut être décrite comme suit. Étant donnés un ensemble de sources d'information **autonomes** et **hétérogènes** (*sources locales*) qui utilisent des schémas possiblement différents, on veut pouvoir interroger les données de ces sources comme si elles constituaient une seule source, avec un schéma unique.

L'approche classique utilisée pour atteindre ce but est la définition d'un **schéma global** [Lenzerini, 2002] pour les données, qui est par la suite lié aux schémas des sources locales. Le schéma global constitue une représentation virtuelle des données stockées dans les sources. Le système d'intégration garde aussi une représentation de chaque source concernée par le processus

d'intégration. On parle de schémas de sources ou **schémas locaux**. Les entités du schéma global et les entités des différents schémas locaux doivent être liées à travers un ensemble de correspondances, appelé *mapping*.

Puisque les données ne résident qu'au niveau des sources, une requête posée sur le schéma global doit être réécrite en termes de requêtes sur les schémas sources. Les résultats obtenus à partir des différentes sources doivent ensuite être combinés conformément au schéma global avant d'être présentés à l'utilisateur.

Le composant qui gère le schéma global, la réécriture de requêtes et les correspondances, est communément appelé *médiateur*[Wiederhold, 1997]. Le médiateur fournit aux utilisateurs une interface unique pour interroger le système. Des contraintes d'intégrité peuvent être ajoutées au schéma global en vue de mieux décrire les besoins du domaine [Calì *et al.*, 2004].

Les composants qui s'occupent de la traduction des requêtes sur le schéma global en des requêtes compréhensibles par les sources locales et le "formatage", sous forme compréhensible par le médiateur, des données fournies par ces sources, sont appelés *adaptateurs*. La figure 2.3 présente l'architecture générale d'un système d'intégration de données.

De manière formelle, un système classique d'intégration de données I est un triplet $< G, S, M >$ [Lenzerini, 2002] où :

1. G est le schéma global, un ensemble de symboles de prédicats globaux (ou relations globales) ayant chacun une arité donnée, et un ensemble de contraintes exprimées sur ces symboles de prédicats ;
2. S est le schéma source, un ensemble de prédicats locaux ou relations locales (disjoint de G) qui constituent la représentation des données contenues dans les sources ;
3. M représente les correspondances entre G et S. Les correspondances sont constituées par un ensemble d'assertions qui établissent le lien entre les prédicats (relations) du schéma global et les prédicats (relations) du schéma source.

L'interrogation du système constitué se fait à l'aide des prédicats globaux définis dans G.

L'établissement des correspondances entre les schémas locaux et le schéma global s'effectue en tenant compte des problèmes liés à l'hétérogénéité des sources en présence.

2.4 Résolution de l'hétérogénéité dans l'interopérabilité de sources

La résolution de l'hétérogénéité sémantique est une des questions fondamentales à résoudre quand on veut réaliser l'interopérabilité entre plusieurs sources. Elle consiste en l'identification d'objets sémantiquement liés dans différentes bases de données et la résolution de la différence de schémas entre eux [Sheth et Kashyap, 1992]. Dans le domaine des bases de données fédérées, Sheth et Larson soulignent que l'hétérogénéité sémantique survient lorsqu'il n'y a pas entente sur la signification, l'interprétation, ou sur l'utilisation souhaitée des données semblables ou liées [Sheth et Larson, 1990]. D'un point de vue plus général, traiter le problème de l'hétérogénéité sémantique revient à traiter le problème de la correspondance entre schémas. Dans le domaine de l'intégration c'est un problème crucial, d'autant qu'intégrer des sources différentes (souvent développées dans des contextes différents) nécessite l'identification des éléments qui peuvent être liés entre les différents schémas. On parle alors de correspondances (ou mapping) entre schémas [Rahm et Bernstein, 2001].

Plusieurs types de conflits dus à l'hétérogénéité peuvent être considérés dans l'établissement des correspondances entre schémas lors de l'intégration de données. De nombreuses taxonomies de conflits, dont Sheth et Kashyap fournissent un large panorama [Sheth et Kashyap, 1992], ont

Tab. 2.1 – Taxonomie de conflits

Conflit	Description
Classification	les populations du monde réel représentées par les deux types sont différentes
Description	les types ont des ensembles différents de propriétés et/ou leurs propriétés sont représentées de manière différente
Structure	les concepts utilisés pour décrire les types sont différents
Hétérogénéité	les modèles de données utilisés sont différents
Méta-données	la correspondance relie un type à un méta-type (un ensemble de noms de types)
Données	des instances en correspondance ont des valeurs différentes pour des propriétés en correspondance

été proposées. Une version plus simple, constituée de six types de conflits, est reprise par Parent et Spaccapietra [Parent et Spaccapietra, 1996] (Cf. tableau 2.1).

Nous détaillons chacun des conflits.

- **Conflits de classification** : ils apparaissent lorsque les types en correspondance décrivent des ensembles différents, mais sémantiquement liés, du monde réel. Par exemple, deux sources médicales peuvent contenir chacune une entité nommée *Médecin* avec pour extension, pour la première, tous les médecins de l'hôpital, et pour la seconde uniquement les médecins spécialistes. La solution standard pour ce genre de conflit est d'inclure dans le schéma intégré la hiérarchie de généralisation/spécialisation appropriée.
- **Conflits descriptifs** : ils surviennent dès qu'il y a une différence entre les propriétés des types en correspondance (les types d'objets peuvent différer selon leurs noms, leurs clés, leurs attributs ; les attributs peuvent aussi différer selon leur noms, leurs structures, etc.). Un exemple de ce type de conflits (différence selon le nom) est l'utilisation de termes synonymes ou homonymes dans la désignation d'un même type d'entité dans deux schémas différents : *Thésard* dans l'un et *Doctorant* dans l'autre.
- **Conflits structurels** : un conflit structurel survient lorsque les éléments en correspondance sont décrits par des concepts de niveaux de représentation différents ou soumis à des contraintes différentes. Par exemple, une classe d'objet, et un attribut, ou un type d'entité et un type d'association. Par exemple, *adresse* qui est un attribut d'une table relationnelle dans un schéma A peut correspondre à une table dans un schéma B.
- **Conflits d'hétérogénéité des modèles de données** : nous avons déjà souligné le fait que la plupart des travaux sur l'intégration ne considèrent que des schémas exprimés sur le même modèle. Les schémas qui ne sont pas exprimés dans ce modèle doivent au préalable être traduits lors d'une phase de prétraitement.
- **Conflits données/méta-données** : ils surviennent lorsqu'une donnée dans une base est en correspondance avec une méta-donnée (le nom d'un type) dans le schéma d'une autre base. En prenant comme exemple deux schémas de bases de données de praticiens avec *praticien1 (id, neurologue, ...)* et *praticien2 (id, fonction, ...)*, certaines valeurs de l'attribut *fonction* de *praticien2* peuvent correspondre au nom de l'attribut *neurologue* de *praticien1*.
- **Conflits de données** : ils surviennent au niveau des instances, lorsque des occurrences en correspondance ont des valeurs en conflit pour des attributs en correspondance. Les conflits de données sont détectés lors de l'exécution d'une requête de l'utilisateur. Ce type de conflits a fait l'objet d'une attention particulière de la part de la communauté des bases

de données, avec les travaux sur la *résolution d'entités* (RE) [Benjelloun *et al.*, 2005].

La résolution d'entités La résolution d'entités, ou RE, également connue sous le nom de déduplication, traite du problème de représentations multiples de mêmes "entités du monde réel" (e.g., des patients, des documents, etc.) dans différentes sources de données. Par exemple, deux sources différentes peuvent stocker la même instance d'un patient de diverses façons (ou avec des erreurs de saisie) : l'adresse dans l'une des sources peut ne pas être représentée de la même façon que dans l'autre. L'objectif de la RE est d'identifier de telles instances et d'effectuer une réconcialiation d'entités afin d'obtenir une instance unique.

Goh et ses collègues [Goh *et al.*, 1999] proposent une autre classification des conflits qui peuvent apparaître lors de la mise en correspondance entre schémas. Ce sont les conflits de nommage (naming), les conflits de graduation (scaling), les conflits de confusion (confounding conflicts) et les conflits de représentation.

1. **Les conflits de nommage** se produisent lors de l'attribution des noms dans des schémas qui diffèrent de manière significative. Les cas les plus fréquents sont les cas de présence de synonymes et d'homonymes.
2. **Les conflits de graduation** apparaissent lorsque différents systèmes de graduation sont utilisés pour mesurer une valeur. On peut citer en exemple la mesure de la température (degrés Celsius ou Farenheit).
3. **Les conflits de confusion** se produisent lorsque les concepts paraissent avoir la même signification mais diffèrent en réalité. Ce type de confusion peut être causé par des contextes temporels différents par exemple.
4. **Les conflits de représentation**, qui se produisent quand deux schémas sources décrivent le même concept de manière différente. Par exemple, dans une source l'adresse peut être designée par une chaîne de caractères tandis que dans une autre l'adresse est une structure composée du numéro et du nom de la rue, du code postal et de la ville.

Pour une revue complète des différentes approches utilisées dans le monde des bases de données pour traiter le problème de l'hétérogénéité entre schémas, nous renvoyons le lecteur à l'état de l'art fait par Rahm et Bernstein [Rahm et Bernstein, 2001] sur les différentes approches de correspondance automatique de schémas, et à celui effectué par Kalfoglou et ses collègues [Kalfoglou et Schorlemmer, 2005] sur la correspondance entre ontologies – les problèmes qui apparaissent lors de la mise en correspondance de schémas peuvent se retrouver lors de la mise en correspondance d'ontologies. Kalfoglou et ses collègues présentent les caractéristiques de 35 projets relatifs à la mise en correspondance entre ontologies. Pinto et ses collègues [Pinto *et al.*, 1999] fournissent aussi une classification des différentes approches pour l'intégration d'ontologies.

Les deux principales approches utilisées pour établir les correspondances entre le schéma source et le schéma global sont *i)* l'approche Global-As-View (GAV), aussi appelée approche centrée-requête, et *ii)* l'approche Local-As-View (LAV) aussi appelée approche centrée-source [Levy *et al.*, 1995].

2.5 Approche centrée-requête Vs Approche centrée-source

Dans l'approche centrée-requête, on définit les prédicats du schéma global en fonction de ceux des sources tandis que dans l'approche centrée-source les prédicats de chaque source sont définis en fonction des éléments du schéma global. Cette mise en correspondance a deux niveaux d'implication :

- sur la façon dont de nouveaux éléments peuvent être ajoutés, tant au niveau global qu'au niveau local ;
- sur l'effet de l'ajout ou de la suppression de sources au niveau local.

2.5.1 L'approche LAV (Local-As-View)

Dans cette approche, M exprime les prédicats locaux en fonction des prédicats du schéma global [Levy *et al.*, 1995]. En d'autres termes, les sources de données sont définies comme un ensemble de vues sur G. Dans ce cas, la base de correspondance M associe à chaque élément de S une requête sur G.

L'avantage principal de l'approche LAV est la facilité offerte pour l'ajout d'une nouvelle source. Cependant l'évaluation des requêtes peut s'avérer complexe. Par ailleurs, tout changement au niveau du schéma global peut être difficile à répercuter au niveau des vues définissant les sources. C'est pourquoi cette approche est préconisée dans le cas où le schéma global n'est pas sujet à de fréquentes modifications.

2.5.2 L'approche GAV (Global-As-View)

Dans l'approche GAV les prédicats du schéma médiateur (schéma global) sont exprimés en terme des prédicats des sources. En d'autre termes, chaque entité du schéma médiateur est une requête sur les sources de données.

L'approche GAV a l'avantage de faciliter considérablement la reformulation de requêtes, se réduisant en fait en un simple dépliement. Cependant, l'ajout d'une nouvelle source peut engendrer des mises à jour complexes du médiateur.

2.5.3 L'approche mixte

Les avantages des approches LAV et GAV ont été combinés dans des approches mixtes. C'est ainsi qu'ont été proposées l'approche Global-Local-As-View (GLAV) [Friedman *et al.*, 1999], qui utilise des règles de correspondance ayant plus d'un terme dans leur en-tête (l'en-tête peut être la combinaison d'un nombre quelconque de prédicats, contrairement à l'approche LAV ou GAV), et l'approche Both-as-View (BAV) [McBrien et Poulovassilis, 2003], qui utilise des transformations incrémentales afin de lier deux schémas. L'approche BAV a été introduite dans le cadre du projet d'intégration AutoMed [Boyd *et al.*, 2002].

2.6 Traitement des requêtes dans un système d'intégration

Dans un système d'intégration, l'interrogation s'effectue généralement en utilisant des *requêtes conjonctives* à base de règles (des requêtes de type *select-project-joint*)[Ullman, 2000] définies à l'aide du vocabulaire du schéma global qui exprime les vues sur les différentes sources. On parle alors d'interrogation basée sur les vues [Levy *et al.*, 1995].

Etant donné un schéma S, une requête conjonctive est de la forme $rep(x) \longleftarrow R_1(x_1), ..., R_n(x_n)$ où

- rep(x) est appelée la tête de la règle,
- $R_1(x_1), ..., R_n(x_n)$ est appelé le corps de la règle,
- les $R(x_i)$ sont appelés des atomes,
- R_i est un nom de prédicat de S,
- chaque x_i est une expression de la forme $e_1, ..., e_{m_i}$,

– chaque e_j est soit une variable, soit une constante de domaine.

Dans un système d'intégration, les requêtes étant posées sur le schéma médiateur ou schéma virtuel (et non sur les bases sources) en utilisant un certain nombre de vues, le système essaie d'effectuer la réécriture des requêtes des utilisateurs en s'assurant que les requêtes réécrites sont soit **équivalentes** aux requêtes initiales, soit **incluses** (de façon maximale) dans chacune des requêtes initiales. La réécriture maximale de requête essaie de trouver la meilleure solution possible par rapport aux sources disponibles.

Etant donné un schéma S et son extension, une requête A sur S contient (inclut) une requête B sur S, dénoté $A \supset B$, si le résultat de l'exécution de la requête B est un sous-ensemble du résultat de l'exécution de la requête A, et ce pour n'importe quelle extension de S. Les deux requêtes A et B sont dites *équivalentes* si les deux ensembles résultats sont égaux quelle que soit l'extension du schéma.

L'algorithme jugé le plus performant pour la réécriture de requêtes est l'algorithme *MiniCon* [Pottinger et Levy, 2000]. Cet algorithme résulte de la combinaison de deux algorithmes. Le premier nommé *bucket* a été développé dans le cadre du système *Information Manifold* [Kirk *et al.*, 1995]. Le second, nommé *inverse-rules,* a été mis en oeuvre dans le système Infomaster [Genesereth *et al.*, 1997]. Les requêtes et les vues acceptées par *MiniCon* sont des requêtes conjonctives avec des prédicats de comparaison arithmétique.

Avec le développement du Web et des technologies de l'information ces dernières années, d'autres approches d'intégration tels que les systèmes Peer-To-Peer (Pair-à-Pair) ont été proposées.

2.7 Les systèmes Pair A Pair (P2P)

L'émergence de systèmes pair-à-pair (Peer-to-Peer) de partage de fichiers a conduit les chercheurs à considérer l'architecture P2P dans le contexte de l'intégration et le partage de données [Ives *et al.*, 2004][Rousset *et al.*, 2006][Nejdl *et al.*, 2002a][Yang et Garcia-Molina, 2002][Milo *et al.*, 2003]. Ces systèmes d'intégration P2P suivent une approche décentralisée pour l'intégration de *pairs* autonomes et distribués contenant des données qui peuvent être partagées. L'objectif principal de tels systèmes est de fournir une interopérabilité sémantique entre plusieurs sources en l'absence de schéma global.

Chaque pair est interconnecté avec un certain nombre de pairs du réseau (appelés voisins) à l'aide de *formules de coordination* [Bernstein *et al.*, 2002]. Edutella [Nejdl *et al.*, 2002a] [Nejdl *et al.*, 2002b], SomeWhere [Rousset *et al.*, 2006], PIAZZA [Ives *et al.*, 2004] sont des exemples de travaux récents sur les systèmes P2P. Compte tenu de l'absence de schéma global, les approches LAV et GAV sont mal adaptées pour ces systèmes, qui utilisent plutôt des approches de type GLAV.

2.8 Utilisation des ontologies pour l'intégration

L'ontologie comme représentation consensuelle et explicite d'une conceptualisation [Gruber, 1993] permet de fournir un vocabulaire partagé pour les différentes sources. Elle permet et facilite la communication de connaissances. D'un point de vue général, Ushold et Gruninger [Uschold et Grüninger, 1996] divisent l'espace d'utilisation des ontologies en trois parties :

1. la communication entre personnes ayant des points de vue et des besoins différents ;
2. l'interopérabilité entre utilisateurs qui ont besoin de s'échanger des données et qui emploient des outils différents ;

3. l'ingénierie des systèmes, où la capacité des ontologies à faire partager et réutiliser des connaissances est exploitée dans la construction et l'utilisation des systèmes à base de connaissances.

L'utilité des ontologies dans le domaine de l'intégration de données peut se retrouver dans les deux premières catégories identifiées par Ushold et Gruninger.

Les ontologies peuvent jouer plusieurs rôles dans le processus d'intégration. Elles peuvent être utilisées pour établir les liens sémantiques entre des éléments dans des sources différentes (e.g., OBSERVER [Mena *et al.*, 2000]). Elles peuvent aussi servir de modèle d'interrogation pour le système intégré lorsqu'elles sont utilisées pour décrire le schéma global (e.g., SIMS [Arens *et al.*, 1997], PICSEL [Rousset et Reynaud, 2003]). Dans le cadre de l'intégration explicite de sources structurées et non structurées, nous décrivons dans le chapitre 5 leur utilisation pour la caractérisation sémantique des sources textuelles.

Classiquement, nous distinguons trois architectures en fontion de la façon dont les ontologies sont utilisées au sein d'une infrastructure d'intégration [Wache *et al.*, 2001] :

1. **Architecture utilisant une ontologie unique** : dans cette approche, chaque source à intégrer est liée à une seule ontologie de domaine globale (e.g., PICSEL [Rousset et Reynaud, 2003]). Ceci implique qu'une nouvelle source ne peut être décrite par sa propre ontologie. Tout ajout de source peut entraîner la modification de l'ontologie globale.
2. **Architecture utilisant plusieurs ontologies** : dans cette approche, chaque source à intégrer est décrite par sa propre ontologie, indépendamment des autres sources. Des correspondances entre les ontologies doivent être définies (exemple de OBSERVER [Mena *et al.*, 2000]). Ces correspondances peuvent se révéler parfois très complexes, notamment à cause de niveaux de granularité différents entre les ontologies.
3. **Architecture utilisant une approche hybride** : dans cette approche, la sémantique de chaque source est décrite par sa propre ontologie. Cependant, les différentes ontologies sont connectées entre elles par un vocabulaire commun partagé (e.g. le projet Kraft [Visser *et al.*, 1999]).

2.9 Exemples de prototypes de systèmes d'intégration

Les bases de données ont été introduites dans les entreprises au début des années 60, au départ pour assurer la gestion de petites applications commerciales. Depuis, un long chemin a été parcouru, et les applications sont devenues de plus en plus complexes et stratégiques pour les organisations. Le développement rapide d'internet et des nouvelles technologies a fait naître des applications nouvelles. Pour une organisation, le besoin d'intégrer les données issues de ces différentes applications devient une évidence.

Les premières approches d'intégration, sous forme de systèmes fédérés, ont vu le jour dans les années 1980 [Hurson et Bright, 1991]. Comme exemple on peut citer le système MULTIBASE [Landers et Rosenberg., 1982]. Après les systèmes multibases ou fédérés, sont apparus les systèmes d'intégration à base de médiateurs (e.g., Garlic [Cody *et al.*, 1995]), les systèmes à base d'agents (e.g, InfoSleuth [Bayardo, Jr. *et al.*, 1997]), et plus récemment encore les système à base d'ontologies (e.g., OBSERVER [Mena *et al.*, 2000], MOMIS [Beneventano *et al.*, 2001] et PICSEL [Rousset et Reynaud, 2003]), les systèmes Pair-à-Pair (e.g., Edutella [Nejdl *et al.*, 2002a], Hyperion [Arenas *et al.*, 2003] et SomeWhere [Rousset *et al.*, 2006]) et enfin les systèmes d'intégration à base de web services (e.g., Active XML [Abiteboul *et al.*, 2002a]).

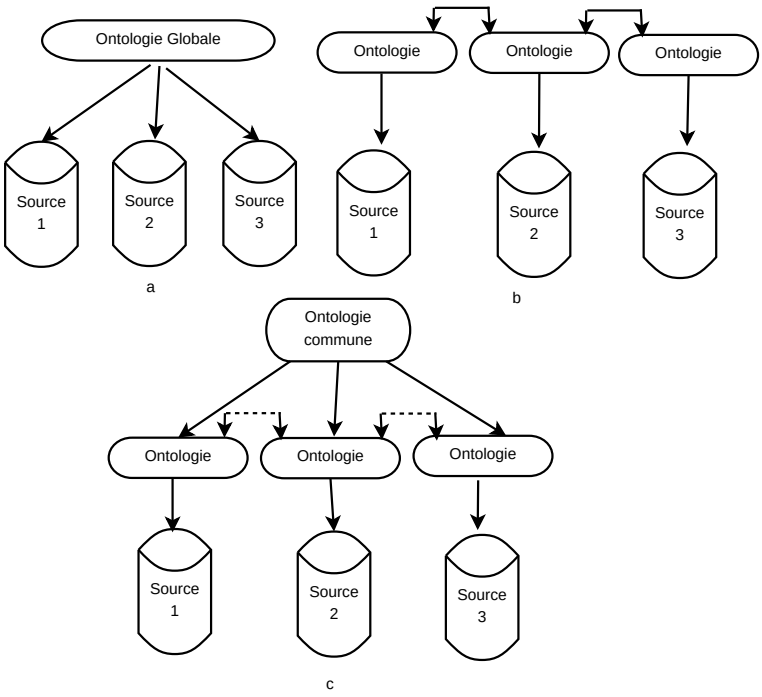


FIG. 2.4 – Utilisation classique des ontologies dans l'intégration

Dans cette section, nous décrivons sommairement un certain nombre d'entre eux (nous ne pouvons tous les présenter, mais les systèmes choisis sont représentatifs de leur famille). Nous allons discuter rapidement des systèmes suivants : Information Manifold [Kirk *et al.*, 1995] comme exemple de système utilisant l'approche LAV, TSIMMIS [Garcia-Molina *et al.*, 1997] comme exemple de système utilisant plusieurs médiateurs, SIMS [Arens *et al.*, 1994][Arens *et al.*, 1997] qui utilise une ontologie globale et l'approche centrée-requête (GAV), MOMIS [Beneventano *et al.*, 2001] comme exemple de système utilisant une approche de découverte semi-automatique de correspondances, OBSERVER [Mena *et al.*, 2000] comme exemple de système utilisant plusieurs ontologies, PICSEL [Rousset et Reynaud, 2003] qui repose sur la combinaison d'un formalisme à base de règles et d'un formalisme à base de classes, AutoMed [Boyd *et al.*, 2002] comme exemple de système utilisant une approche mixte pour l'expression des correspondances et enfin XYLEME [Abiteboul *et al.*, 2002b] comme exemple de système d'intégration de données semi-structurées.

2.9.1 Information Manifold

Information Manifold (IM) est un système de navigation et d'interrogation pour des sources structurées et peu structurées. Son objectif est de fournir une interface d'interrogation uniforme pour plusieurs sources de données. Le système a été notamment utilisé pour intégrer des informations provenant de plusieurs sources sur le web.

IM est basé sur l'utilisation d'une base de connaissances qui décrit le modèle conceptuel du domaine et les propriétés des sources d'information. La base de connaissances repose sur un modèle de données très riche. IM considère comme source structurée les bases de données, les documents SGML, tandis que les sources peu structurées sont les sites FTP, les fichiers bibliographiques, etc. Le langage utilisé pour représenter le contenu des sources d'information

est une combinaison de règles de Horn et de notions issues de la logique de description CLASSIC [Brachman *et al.*, 1991]. Les correspondances entre le schéma de médiation et les sources suivent l'approche LAV, qu'IM a été le premier à proposer.

2.9.2 TSIMMIS

TSIMMIS (The Stanford-IBM Manager of Multiple Information Sources) est un système destiné à l'intégration (manuelle) de plusieurs sources hétérogènes. C'est un système fédéré basé sur l'architecture médiateur/adaptateurs utilisant l'approche centrée-requête [Garcia-Molina *et al.*, 1997]. Le modèle de données utilisé pour décrire le contenu des sources et pour exprimer les requêtes est l'Object Exchange Model (OEM). Chaque adaptateur est un programme qui s'occupe d'une source particulière. Il reçoit les requêtes posées dans le langage OEM et les transforme en langage compréhensible par la source. Les sources traitées sont structurées et semi-structurées. TSIMMIS ne fournit pas de schéma global unique et utilise plusieurs médiateurs.

2.9.3 SIMS

SIMS (Services and Information Management of Decision Systems) est un système qui permet d'intégrer des bases de données et de connaissances ainsi que des pages web [Arens *et al.*, 1994]. Le système d'intégration utilise l'architecture médiateur/adaptateurs suivant l'approche centrée-source et une ontologie globale, appelée le modèle du domaine, décrite en LOOM [MacGregor, 1991]. L'interrogation du système se fait en utilisant des requêtes exprimées en langage LOOM et un sous-ensemble de SQL (qui inclut les opérateurs de jointure, de sélection et de projection). Le traitement des requêtes inclut des phases de planification et d'optimisation sémantique de requêtes lors desquelles plusieurs plans de résolution sont évalués.

2.9.4 OBSERVER

OBSERVER utilise plusieurs ontologies de domaine préexistantes pour décrire les informations contenues dans les sources à l'aide d'un système basé sur les DL.

Les termes issus des ontologies sont utilisés comme vocabulaire de caractérisation des informations spécifiques à un domaine (pour chaque source). Une des ontologies considérées dans OBSERVER est la ressource lexicale WordNet [Fellbaum, 1999], utilisée pour expliciter le sens des noms d'attributs et des tables relationnelles de certaines sources. Comme le système gère plusieurs ontologies, la notion de correspondance entre ontologies a été introduite. Elle s'établit à l'aide de relations inter-ontologies (Interontologies Terminological Relationships) telles que la synonymie et l'hyperonymie. Ainsi, pour une source décrite à l'aide d'une ontologie A, le système est capable de l'interroger à l'aide de termes d'une ontologie B, si toutefois des relations inter-ontologiques ont été définies entre ces ontologies.

OBSERVER ne prend pas en compte la notion d'ontologie globale, et des ontologies déjà existantes sont utilisées pour la description des sources (de nouvelles ontologies pour décrire une source ne sont pas construites). Dans cette approche, on considère qu'il est plus facile de gérer séparement plusieurs ontologies de taille réduite : si plusieurs ontologies sont disponibles, il est considéré comme plus aisé de définir des correspondances entre elles plutôt que de les intégrer dans une ontologie globale.

2.9.5 PICSEL

PICSEL, Production d'Interface à base de Connaissances pour les Services En Ligne [Goasdoué *et al.*, 2000] est un projet développé au sein de l'équipe IASI du LRI pour le compte de France Télécom R&D. Le système intègre des bases de données relationnelles et des documents XML. PICSEL utilise un schéma global (qui modélise le domaine concerné par l'intégration) exprimé dans un langage appelé CARIN [Levy et Rousset, 1995] combinant le pouvoir d'expression d'un formalisme à base de règles et d'un formalisme à base de classes.

2.9.6 MOMIS

MOMIS (Mediator envirOnment for Multiple Information Sources) [Beneventano *et al.*, 2001], qui implémente l'approche médiateur/adaptateurs, est fondé sur l'utilisation d'une logique de description très riche, ODL-I3, pour décrire les schémas des sources de données à intégrer. Les sources peuvent à la fois être des sources structurées et semi-structurées. Le schéma global est inféré à partir des différents schémas sources (approche GAV). MOMIS utilise ARTEMIS (Analysis and Reconciliation Tool Environment for Multiple Information Sources) [Castano *et al.*, 2001][Castano et Antonellis, 1999] pour les correspondances entre les différents schémas. ARTEMIS procède par analyse sémantique et par clustering de classes décrites en ODL-I3 pour suggérer les correspondances entre schémas.

2.9.7 AutoMed

AutoMed [Boyd *et al.*, 2002] propose une approche d'intégration basée sur une séquence d'étapes de transformations bidirectionnelles [McBrien et Poulovassilis, 2003]. Pour transformer un schéma donné en un autre, on procède de façon incrémentale à l'ajout, la suppression ou le renommage de constructeurs. Une des particularités du système est l'utilisation d'un modèle de données basé sur les graphes, appelé hypergraphe data model (HDM), permettant aussi bien de prendre en compte le modèle entité/association, le modèle relationnel, UML, etc. Il propose comme approche de mise en correspondance des schéma locaux et du schéma global l'approche BAV (Both-As-View). Les chemins définis à travers les tranformations BAV permettent de définir les relations logiques entre schémas. Une autre particularité d'AutoMed est qu'il peut aussi bien implémenter un schéma virtuel de médiation qu'un entrepôt de données.

2.9.8 XYLEME

XYLEME [Abiteboul *et al.*, 2002b][Rousset et Reynaud, 2003] est un exemple de système d'intégration de données semi-structurées XML. Dans son médiateur, le schéma global est un ensemble de *DTDs abstraites* constituant la hiérarchie de termes qui structurent le vocabulaire du domaine concerné par le processus d'intégration. XYLEME combine l'approche LAV et GAV, car la correspondance entre le schéma global et les sources est exprimée par de simples correspondances de chemins.

2.10 Les systèmes d'intégration dans le domaine biomédical

La multiplicité des sources biomédicales et la nécessité d'intégrer des résultats provenant de plusieurs appareils de mesures font que des travaux sur l'intégration sont de plus en plus *appliqués* au domaine biomédical. Par exemple, pour étudier un phénomène donné, les biologistes sont obligés de tenir compte des aspects physiologique, génétique, anatomique, biochimique, etc.

Tandis que les médecins, pour établir un diagnostic donné ou la corrélation entre certains phénomènes, doivent prendre en compte les aspects anatomique, fonctionnel, etc. Plusieurs travaux ont été menés dans le domaine de l'intégration de données biomédicales. Karp [Karp, 1996] fournit une taxonomie des approches utilisées.

Nous présentons ici des travaux contemporains dans le domaine.

2.10.1 Sequence Retrieval System

Sequence Retrieval System (SRS) [Zdobnov *et al.*, 2002] est plus proche d'un système de recherche basé sur les mots-clés que d'un système intégré. Son approche d'intégration consiste à analyser des fichiers plats ou des banques de données qui contiennent des fichiers textes de structurés avec les noms de champs. Il crée et stocke un index pour chaque champ et utilise ces index locaux au moment de l'interrogation pour retrouver les entrées pertinentes. SRS possède son propre analyseur de fichiers plats appelé ICARUS (Interpreter of Commands And Recursive Syntax). L'utilisation de ICARUS permet à SRS de détecter la présence de liens et d'indexer tous les enregistrements en utilisant une technique d'indexation basée sur les mots-clés. SRS offre la possibilité de garder les références croisées entre sources. Si une source A fait reférence à une source B, alors la reférence de B vers A est aussi considérée. De même, des références entre sources peuvent être générées par transitivité.

2.10.2 TAMBIS

TAMBIS (Transparent Access to Multiple Bioinformatics Information Sources) est un système de médiation basé sur une ontologie [Stevens *et al.*, 2000]. Les requêtes dans TAMBIS sont formulées à travers une interface graphique où l'utilisateur navigue à travers les concepts définis au niveau du schéma global et choisit ceux qui l'intéressent pour la requête courante. Le système utilise la logique de description GRAIL [Rector et Bechhofer, 1997], qui est aussi utilisée pour exprimer des requêtes sur le système. Toute requête exprimée en GRAIL est traduite en QIF (Query Internal Format), puis dans un plan d'exécution dépendant des sources.

2.10.3 DiscoveryLink

DiscoveryLink, développé chez IBM, est un système d'intégration de sources basé sur les adaptateurs [Haas *et al.*, 2001]. Il sert d'intermédiaire aux applications qui ont besoin d'accéder à plusieur sources biomédicales. DiscoveryLink consiste en fait en une couche d'intégration contruite sur le projet Garlic. Il sert de middleware entre les applications et un ensemble d'adaptateurs. Les applications soumettent des requêtes SQL à DiscoveryLink sans se soucier de la nature des sources. Garlic est un système fédéré de traitement de requêtes qui communique avec plusieurs adaptateurs dédiés pour déterminer le plan optimal d'exécution d'une requête et l'exécuter [Roth *et al.*, 1996]. DiscoveryLink est basé sur un modèle orienté-objet.

2.10.4 Knowledge-based Integration of Neuroscience Data : KIND

KIND [Gupta *et al.*, 2000] est un système d'intégration de sources de données dans le domaine des neurosciences. Il étend l'approche conventionnelle médiateur/adaptateurs avec l'utilisation de plusieurs bases de connaissances qui fournissent le lien sémantique entre les sources à travers des faits et règles sur le domaine d'application. KIND est basé sur le langage F-logic [Kifer *et al.*, 1995]. Notons que le système n'intègre que des sources structurées.

2.10.5 ONTOFUSION

Ontofusion [P ?rez-Rey *et al.*, 2006] est un système d'intégration de données biomédicales (structurées) basé sur les ontologies. L'intégration s'effectue au travers de deux processus : un processus de mapping et un processus d'unification. Dans le processus de mapping, le schéma physique de chaque base de données est relié à un schéma qualifié de "schéma virtuel". Un schéma virtuel est défini dans ce système comme étant l'ensemble des ontologies qui représentent, à un niveau conceptuel, la structure des informations contenues dans une source donnée. Durant le processus d'unification, plusieurs schémas virtuels correspondant aux différentes bases sont regroupés au sein d'un schéma virtuel unique. Le système distingue trois types de sources : *i)* les bases de données dites publiques (e.g., SwissProt[24]) ; *ii)* les bases de données privées qui ont un schéma physique accessible et connu *iii)* et enfin les bases de données stockant des vocabulaires biomédicaux. Cette dernière catégorie sert à stocker des ontologies ou vocabulaires biomédicaux comme UMLS [Bodenreider, 2004], et la Gene Ontology [Consortium, 2001]. Ce stockage local des vocabulaires peut poser la question de leur mise à jour lorsque leur source primaire est modifiée.

2.10.6 Neurobase

Neurobase [Barillot *et al.*, 2003] est un projet commun entre plusieurs laboratoires français (2003-2005), dédié à la gestion de données et de connaissances réparties en neuroimagerie. Son objectif initial était de spécifier un modèle de reférence pour le partage de données hétérogènes et distribuées en imagerie cérébrale. Neurobase implémente un système fédéré suivant l'approche méditaur/adaptateurs utilisant un référentiel sémantique commun (une ontologie médicale définie pour les besoins du projet). Il est basé sur le médiateur Le Select[25], qui permet de partager des données et des programmes hétérogènes et distribuées à travers un langage de requêtes de haut niveau.

2.11 Conclusion

Dans ce chapitre, nous avons fait une rapide présentation des travaux concernant l'intégration de données hétérogènes. Après avoir décrit les deux grandes approches utilisées pour la gestion unifiée, nous avons décrit les différents types de conflits à résoudre lors de l'intégration de systèmes hétérogènes. L'aspect modélisation des données, c'est à dire comment intégrer les différents schémas des sources, et leur interrogation, c'est à dire comment répondre efficacement aux requêtes posées sur le schéma global, ont été abondamment abordés dans la littérature [Halevy, 2001][Goasdoué *et al.*, 2000]. Nous avons décrit les approches utilisées pour établir des correspondances entre différents schémas. Les systèmes que nous avons décrits ne s'occupent pas de l'étape de prétraitement des sources à intégrer.

Pour l'établissement des correspondances entre schémas, la plupart des systèmes sont basés sur une analyse manuelle du schéma effectuée par un expert, soit du domaine, soit d'intégration, même si des approches semi-automatiques de mise en correspondance sont parfois proposées, avec notamment l'utilisation de techniques issues de l'intelligence artifielle. Apporter une certaine automatisation dans le processus de prise en compte des sources peut permettre une économie de temps et de coût.

[24] http ://www.ebi.ac.uk/swissprot/

[25] `http://www-caravel.inria.fr/Faction_Le_Select.html`

2.11.1 Le rôle de l'intelligence artificielle (IA)

La communauté IA est une communauté assez active dans le domaine de l'intégration de données. Son rôle peut se situer à trois niveaux : les logiques de description, les travaux sur la planification en IA et les travaux sur l'apprentissage (Machine Learning) [Halevy, 2005].

1. Les DLs : Les logiques de description, une branche de la Représentation de la Connaissance, ont été identifiées comme un bon moyen de description des relations entre les différentes sources de données [Catarci et Lenzerini, 1993]. L'approche LAV a été inspirée d'une part par le fait que les sources de données doivent être représentées de manière déclarative (le schéma de médiation de IM est basé sur la DL CLASSIC [Borgida *et al.*, 1989]) et d'autre part par les travaux combinant le pouvoir expressif des DLs et les langages d'interrogation des bases de données [Levy et Rousset, 1995].
2. Les travaux sur la planification en IA : ces travaux ont influencé la reflexion sur la reformulation et le traitement de requêtes dans les systèmes d'intégration [Arens *et al.*, 1994][Barish et Knoblock, 2003].
3. Les travaux sur l'apprentissage en IA jouent un rôle central pour les systèmes d'intégration, dans la génération semi-automatique de correspondances sémantiques [Rahm et Bernstein, 2001].

2.11.2 Le rôle des ontologies

En vue de faciliter l'interopérabilité, les ontologies fournissent un vocabulaire partagé qui peut servir à décrire les différentes sources. Les systèmes existants utilisent essentiellement les ontologies dans la définition du schéma global (schéma de médiation), parfois dans la description des sources locales.

Nous montrerons dans l'architecture de gestion unifiée que nous proposons (voir chapitre 4), que les ontologies peuvent aussi être utilisées pour décrire le contenu de sources textuelles en vue de leur prise en compte dans un système d'intégration.

La proposition que nous faisons tient compte des avancées faites dans le domaine de l'intégration, en particulier sur la réécriture de requêtes et la mise en correspondance entre schémas (ontologies), que nous n'avons pas abordée dans notre travail.

Chapitre 3

La Recherche d'Information

3.1 Introduction

Dans ce chapitre nous faisons un rappel sur la recherche d'information (RI) et les principaux modèles utilisés par les systèmes de recherche d'information (SRI).

Le domaine de la RI remonte au début des années 1950, peu après l'invention des ordinateurs. A l'origine, les chercheurs s'intéressaient à la problématique de l'**indexation des documents**. Les premiers systèmes, parmi lesquels on peut noter KWIC, de H.P. Luhn, virent le jour à la fin des années 1950. Le système d'indexation KWIC sélectionnait les index selon la fréquence des mots dans les documents et filtrait des mots vides de sens en employant des ***stop lists*** (**liste de mots outils**). C'est dans cette période que le domaine de la RI est vraiment né.

Classiquement, la recherche d'information est définie comme étant l'ensemble des actions, méthodes et procédures ayant pour objet d'extraire d'un ensemble de documents les informations voulues (d'après l'AFNOR). Dans un sens plus large, c'est toute opération (ou ensemble d'opérations) ayant pour objet la collecte et l'exploitation d'informations en réponse à une question sur un sujet précis. Le système qui permet de mettre en oeuvre ces différentes opérations est appelé Système de Recherche d'Information (SRI). Son but est de retrouver les documents **pertinentes** dans une grande **collection de documents**. La notion de RI renvoie généralement, comme indiqué par Van Rijsbergen [van Rijsbergen, 1979], aux travaux publiés par Cleverdon [Cleverdon *et al.*, 1966], Salton [Salton, 1971], Sparck Jones [Jones et Needham, 1968b], Lancaster [Lancaster, 1968] et bien d'autres.

Notons que même si les premiers travaux en RI sont plutôt d'origine anglo-saxone [Cleverdon, 1967] [Lancaster, 1968][Salton, 1968][Blair et Maron, 1985], des équipes francophones se sont intéressées à la question depuis les années 1980 [Chiaramella et Defude, 1987].

3.1.1 Les quatre principaux modes de recherche d'information

Selon la façon dont s'effectue la recherche d'information, on distingue :

1. La recherche par navigation arborescente : ce type de recherche s'effectue par menus successifs. L'information est structurée en partant du général au particulier (e.g., l'annuaire de Google ou de Yahoo !, la table des matières d'un livre).
2. La recherche par navigation hypertextuelle : elle s'effectue par navigation dans un réseau de noeuds et de liens (e.g., navigation à travers un ensemble de sites web).
3. La recherche par requête sur les métadonnées du document : dans ce mode, l'information est au préalable indexée, en général manuellement. La recherche s'effectue en spécifiant

différents champs, souvent liés par des opérateurs booléens (e.g., catalogue de bibliothèque).

4. La recherche par requête sur le texte intégral : dans ce mode, une démarche d'analyse linguistique est effectuée au préalable afin de procéder à une **représentation** automatique des documents sur lesquels la recherche doit s'effectuer. Ce type de recherche utilise des outils de traitement automatique du langage naturel (TALN). La recherche d'information dont nous parlons dans ce chapitre fait référence en priorité à ce mode de recherche.

L'objectif principal d'un système de recherche d'information est de fournir aux utilisateurs les documents qui satisfont un certain **besoin d'information** exprimé au travers d'une **requête**.

3.1.2 Définition des notions de base

Collection de documents : La collection de documents représente les documents disponibles, à utiliser par le SRI. On notera que le Web constitue la plus vaste collection de documents actuellement disponible, collection qui n'est exploitée entièrement par aucun moteur de recherche.

Document : Le document représente un item de la collection. Il peut aussi bien s'agir d'un document dans sa totalité que d'une partie d'un document, et le terme "document" peut désigner des informations non textuelles tels que des documents multimédias, de la vidéo, etc. Le terme "document" fait donc référence à n'importe quel élément stocké dans le SRI. Dans la suite, ce terme représentera des informations textuelles.

Requête : Une requête représente l'expression du besoin d'information de l'utilisateur. Elle encode son besoin dans une forme compréhensible par le SRI.

Besoin d'information : la notion de besoin d'information renvoie souvent à la notion du besoin de l'utilisateur, classé en trois catégories par Ingwersen [Ingwersen, 1992] : i) le besoin de vérification (l'utilisateur possède déjà les données qui lui permettent d'accéder à l'information), ii) le besoin thématique connu (l'utilisateur cherche à clarifier ou à trouver des informations dans un domaine connu), iii) le besoin thématique inconnu (l'utilisateur cherche de nouvelles informations qui lui sont totalement inconnues).

Pertinence : la notion de pertinence d'un document pour une question a émergé en même temps que les premiers systèmes de RI, avec les premières mesures permettant de comparer les différents résultats renvoyés par les systèmes de RI. Saravecic [Saracevic, 1970] a répertorié un ensemble de définitions de cette notion, dont nous reproduisons ci-dessous les deux qui nous semblent les plus représentatives :

1. La pertinence caractérise la correspondance entre un document et une requête ; c'est une mesure de l'informativité du document à la requête ;
2. La pertinence est un degré de relation (chevauchement, etc.) entre le document et la requête.

La pertinence mesure donc le degré de correspondance entre un document et une requête. C'est une notion subjective qui se mesure souvent en termes de précision et de rappel (notions sur lesquels nous reviendrons à la Section 3.6).

La représentation des documents : les documents écrits dans le langage naturel ne peuvent pas être directement comparés à la requête. Une représentation interne, exploitable, de ces documents, doit être obtenue au préalable. Ce processus de représentation interne s'appelle l'**indexation**. Le but de l'indexation est double :

1. identifier les descripteurs les plus significatifs contenus dans chaque document ;
2. mesurer l'importance de chaque descripteur pour chaque document.

Les descripteurs du document se réduisent la plupart du temps aux termes simples (souvent transformés par des opérations telle que la lemmatisation) présents dans le document. L'indexation obtenue dans ce cas est une indexation basée sur les termes. La requête doit aussi être représentée dans la même forme interne que les documents.

On peut considérer, selon le niveau d'automatisation, trois catégories d'indexation :

1. l'indexation manuelle : le processus d'identification des descripteurs représentant le document est effectué manuellement (c'est le travail des documentalistes) ;
2. l'indexation automatique : l'indexation s'effectue de façon entièrement automatisée ;
3. l'indexation semi-automatique : généralement le processus d'indexation s'effectue en deux étapes, une première étape d'identification automatique des descripteurs et une seconde étape ou le spécialiste de l'indexation sélectionne les "bons" descripteurs ;

L'indexation automatisée est sans aucun doute celle qui a fait l'objet de plus d'attention par la communauté de la RI.

Plusieurs techniques permettent d'obtenir une représentation interne des documents :

1. utilisation d'un vocabulaire contrôlé, c'est-à-dire, une liste de termes définie a priori ;
2. sélection de termes présents dans le document ;
3. une combinaison des deux.

La figure 3.1 présente l'architecture général d'un SRI classique [Lancaster et Warner, 1993]

3.2 Les modèles utilisés par les SRI

Le rôle d'un modèle est tout d'abord de donner une signification au résultat de l'indexation. Un document est représenté par un ensemble de termes significatifs et leurs poids. S'il est communément admis qu'un terme d'un document présent dans l'index est censé être important pour ce document, la façon d'interpréter son poids (ou son importance) pour le document, peut varier. Un modèle théorique est censé d'une part donner un cadre d'interprétation précis à ce poids, mais aussi les relations possibles qui peuvent exister entre les termes d'indexation. Ceci nous ramène à la notion de représentation des documents (ou des requêtes, considérés aussi comme des documents). Le modèle doit déterminer la relation entre le document et la requête à partir de leurs représentations, en procédant le plus souvent à un calcul de similarité.

On peut classer les modèles de recherche d'information en trois catégories. Les modèles booléens, les modèles statistiques – qui incluent le modèle vectoriel et le modèle probabiliste – et enfin les modèles basés sur les thésaurus et autres ressources externes de connaissances. Belkin et Croft [Belkin et Croft, 1992] qualifient le modèle booléen de modèle d'**appariement exact** tandis que les autres sont qualifiés de **meilleur appariement**.

La figure 3.2 représente de façon simplifiée un modèle de RI. La partie gauche représente l'index des documents. La partie droite est le besoin d'information, représenté par une requête, exprimée en langage naturel ou dans un langage plus formel. Au centre, nous avons la fonction

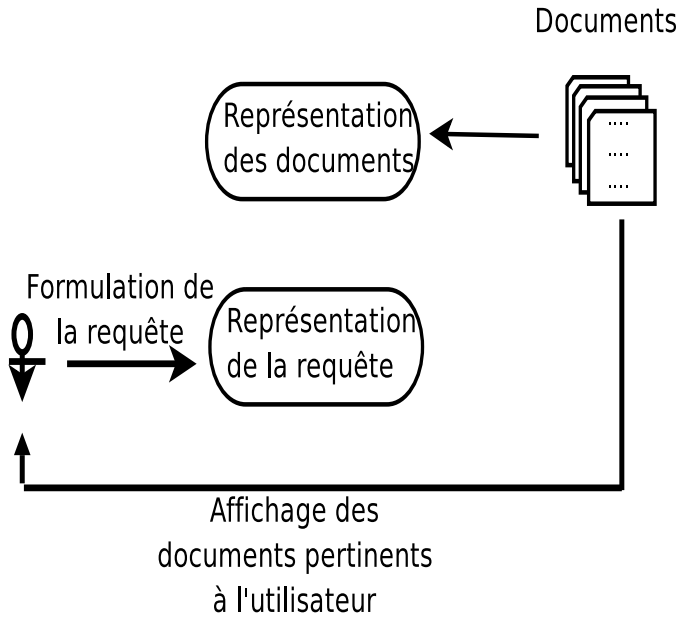


FIG. 3.1 – Architecture d'un SRI

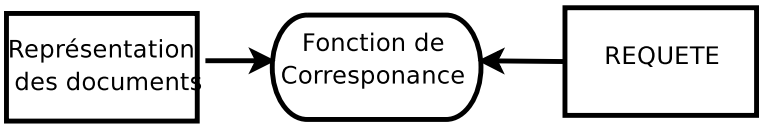


FIG. 3.2 – Un simple modèle de RI

de correspondance, qui compare les éléments de la requête aux éléments de l'index et retourne les documents qui satisfont cette correspondance.

3.2.1 Le modèle booléen

Le modèle booléen, le plus simple des modèles de recherche et le plus facile à implémenter, utilise les opérateurs booléens. Il applique le principe du tout ou rien. Les termes de la requête sont liés par les opérateurs ET, OU et NOT. Même si plusieurs moteurs de recherche contemporains implémentent ce modèle, il présente un certain nombre d'inconvénients :

- L'utilisateur doit avoir une certaine connaissance sur le sujet pour que la recherche soit efficace. En effet, l'utilisation d'un mot inapproprié dans la requête peut mal classer un document pertinent. L'expression des requêtes booléennes est souvent jugée fastidieuse par les utilisateurs [Belkin et Croft, 1992] ;
- L'opérateur booléen *ET* tend à être très restrictif. Une recherche sur "A *ET* B *ET* C" élimine tout document qui ne contient pas ces trois termes à la fois.
- Aucun poids ne peut être affecté aux termes de la requête pour dénoter leur importance, même si certains termes de la requête paraissent plus importants à l'utilisateur.
- Le classement des résultats par pertinence est impossible.

D'autres techniques ont été combinées avec l'approche booléenne pour en améliorer la performance. Salton, Fox et Wu ont introduit en 1983 le modèle booléen étendu, en proposant de pondérer les termes des documents à travers la méthode dite P-norm [Salton *et al.*, 1983b].

Carpineto et Romano [Carpineto et Romano, 1998] la combinent avec la navigation basée sur le contenu. Lee et ses collègues [Lee *et al.*, 1993] ont remplacé les opérateurs booléens par des opérateurs flous. Ils montrent que l'utilisation d'une classe d'opérateurs appelée opérateurs de compensation positifs permet d'améliorer l'efficacité de leur système. Kwon [Kwon *et al.*, 1994] utilise un thésaurus pour l'expansion de requêtes pondérées.

3.2.2 Le modèle probabiliste

Le modèle probabiliste [Fuhr, 1992] est basé sur le principe d'ordonnancement probabiliste, qui considère qu'un SRI est supposé ordonner les documents sur la base de leur pertinence par rapport à la requête. La probabilité qu'un document soit jugé pertinent par rapport à une requête est basée sur l'hypothèse que les termes sont distribués différemment dans les documents pertinents et non pertinents. La formule de probabilité est souvent dérivée du théorème de Bayes [van Rijsbergen, 1979]. Les recherches sur les modèles probabilistes ont commencé dans le milieu des années 1970. Van Rijsbergen [van Rijsbergen, 1977], Robertson et Spark Jones [Robertson et Jones, 1976], font partie des précurseurs. Les systèmes Inquery [Callan *et al.*, 1992] et Okapi [Robertson *et al.*, 1995] sont basés sur ce modèle.

3.2.3 Le modèle vectoriel

Le modèle vectoriel représente les documents et les requêtes de l'utilisateur dans un espace multi-dimensionnel dans lequel les dimensions sont les termes utilisés pour construire l'index représentant les documents [Salton et McGill, 1986]. La figure 3.3 montre l'architecture d'un système de recherche d'information utilisant le modèle vectoriel [Croft, 1993].

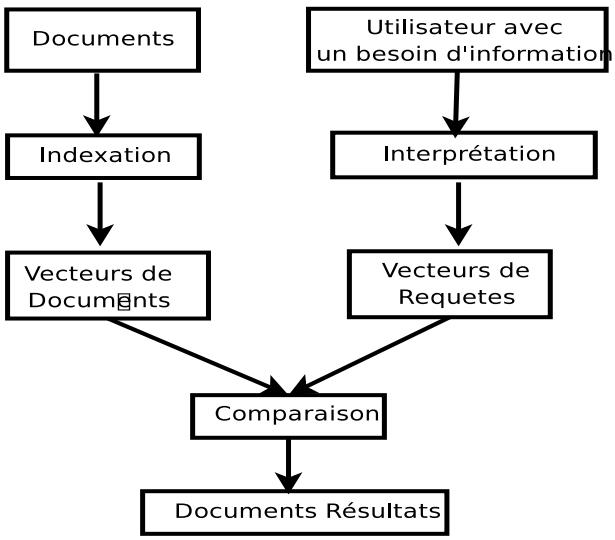


FIG. 3.3 – Architecture d'un SRI utilisant le modèle vectoriel

Un vecteur est utilisé pour représenter chaque document dans une collection. Chaque composant du vecteur représente un concept particulier, un mot clé ou un terme associé au document donné. La valeur affectée à ce composant reflète l'importance de l'item dans la caractérisation du document. Typiquement, la valeur est fonction de la fréquence avec laquelle l'item apparaît dans le document ou dans la collection totale.

Les modèles que nous venons de présenter utilisent classiquement des descripteurs qui se réduisent aux termes simples présents dans les documents (représentation en *sac-de-mots*, l'ordre des mots n'importe pas). Sachant que ces termes peuvent avoir plusieurs synonymes ou homonymes, cette représentation des documents peut avoir des effets négatifs sur la performance des SRI. En effet, elle peut être source de bruit (plusieurs documents non pertinents retournés) et de silence (plusieurs documents pertinents non retournés). L'indexation Sémantique Latente a été proposée pour améliorer les approches classiques.

3.2.4 Indexation Sémantique Latente (LSI)

L'indexation sémantique latente (Latent Semantic Indexing) [Deerwester *et al.*, 1990][Furnas *et al.*, 1988] est une méthode statistique de recherche d'information conçue pour répondre aux deux problèmes fréquents en RI causés par la synonymie et la polysémie. Le LSI essaye de tirer profit du contenu conceptuel des documents. Au lieu d'effectuer la recherche sur des termes pris individuellement, la recherche s'effectue sur un espace conceptuel. Les termes sont appariés sur un espace de concepts. L'association entre les termes et les documents est calculée et exploitée à la phase de recherche. Le LSI utilise une technique de réduction de dimensions appelée Décomposition en Valeurs Singulières (*Singular Value Decomposition*) ou SVD.

Une première critique qu'on peut faire au LSI est que l'espace conceptuel est incompréhensible par l'homme (en comparaison à l'utilisation d'une ontologie). L'information est codée en valeurs numériques rendant le LSI difficile à comprendre.

Une seconde critique est sa performance. L'algorithme SVD est $O(N^2k^3)$ [Deerwester *et al.*, 1990], où N est le nombre de termes et de documents et k est le nombre de dimensions dans l'espace conceptuel. L'algorithme se révèle inefficace pour une grande collection de documents dynamique, mais acceptable pour une collection stable. L'ajout d'un nouveau document ou d'un nouveau terme dans la collection peut aussi poser problème car il faut reconstruire l'espace conceptuel.

3.2.5 La classification de textes (CT)

Etant donnée une collection de documents, la classification de textes est une activité qui consiste à affecter aux documents de la collection des catégories prédéfinies. Elle trouve son origine dans les travaux de Maron [Maron, 1961] sur la classification probabiliste de textes.

La CT est utilisée dans plusieurs contextes allant de l'indexation de documents basée sur un vocabulaire contrôlé au filtrage de documents, la génération automatique de méta-données, la désambiguïsation de termes, etc. [Sebastiani, 2002]. Deux approches sont communément utilisées pour la CT :

1. l'ingénierie des connaissances, qui consiste à encoder manuellement la façon dont un expert affecte des documents dans des catégories prédéfinies ;
2. l'apprentissage, qui consiste à construire un classifieur automatique de textes à partir d'un ensemble de documents pré-classifés, en utilisant une approche inductive. L'approche par apprentissage a l'avantage d'économiser le temps de l'expert.

Formellement, la CT est décrite comme suit : étant donné un ensemble de documents noté D et un ensemble de catégories noté C, l'activité de classification consiste à affecter une valeur booléenne B à chaque couple $< d_j, c_i > \in DxC$, où D est le domaine des documents et $C = \{c_1, ..., c_n\}$ est l'ensemble des catégories prédéfinies. Une valeur positive de B indique que le document d_j doit être classé dans la catégorie c_i alors qu'une valeur négative (B=*faux*) de B indique que d_j

ne doit pas être classé sous la catégorie c_i. La tâche de catégorisation revient alors à approximer une fonction inconnue $\Theta' = DxC \rightarrow \{T, F\}$ à travers une fonction $\Theta = DxC \rightarrow \{T, F\}$ appelée fonction de classification (ou modèle), de telle sorte que Θ et Θ' coïncident le plus souvent possible.

La classification peut s'effectuer en utilisant ou non des connaissances externes disponibles. Dans le cas de l'utilisation de connaissances implicites, la classification des documents s'effectue uniquement sur la base de leur *sémantique intrinsèque,* considérée comme une notion subjective. L'appartenance d'un document à une catégorie ne peut être décidée de manière déterministe dans ce cas. Selon le nombre de catégories affectées à chaque document d_j (0 ou plusieurs), on parle de classification à label simple ou à labels multiples. Pour le cas simple, chaque document est affecté à une catégorie c ou à son complément c'.

La représentation des documents pour la classification est basée traditionnellement sur l'approche classique *sac-de-mots* [Yang, 1999] qui présente un certain nombre d'inconvénients. C'est pourquoi, ces dernières années, des travaux ont été menés en vue de substituer à l'approche sac-de-mots une approche à base de concepts issus par exemple d'une ontologie [Gabrilovich et Markovitch, 2005][Litvak *et al.*, 2005][Bloehdorn *et al.*, 2005][Wang *et al.*, 2003][Wu *et al.*, 2003]. Comparées aux autres approches de représentation de la sémantique d'un document pour la classification, les ontologies présentent plusieurs avantages :

1. elles peuvent être interprétées et éditées par l'homme ;
2. les éventuelles erreurs peuvent être détectées et corrigées ;
3. elles peuvent être utilisées par plusieurs applications dans des contextes différents, contrairement à l'espace conceptuel généré par exemple par le LSI.

3.2.6 Le clustering de documents

Pour une requête donnée, un SRI est censé parcourir tous les documents (en réalité leur représentation) afin des les comparer à la requête. Comme l'examen de tous les documents peut être coûteux (surtout pour un très grand corpus), une alternative raisonnable est de regrouper au préalable les documents qui sont similaires dans des **catégories** ou **clusters** [Kobayashi et Takeda, 1999]. L'utilisation des techniques de clustering – ou classification non supervisée – est basée sur l'*hypothèse du clustering*, qui stipule que les documents ayant des contenus similaires sont également pertinents pour la même requête [van Rijsbergen, 1979]. Rasmussen [Rasmussen, 1992] mentionne un certain nombre d'approches de clustering :

1. le clustering suivant les termes que les documents contiennent. Cette approche inclut des techniques statistiques pour déterminer les relations entre les termes ;
2. le clustering suivant la co-occurrence des références (e.g., [SMALL, 1973]) . Cette approche est notamment utilisée dans le contexte du clustering de documents issus du Web.

Plusieurs considérantions sont à prendre en compte lors du développement et de l'implémentation d'un algorithme de clustering. Parmi elles, mentionnons [Kobayashi et Takeda, 1999] : i) le choix des attributs pertinents à utiliser comme base pour l'identification des clusters, et l'identification d'un poids approprié pour ces attributs ; ii) le choix d'une méthode de clustering et d'une mesure de similarité appropriées.

Il existe plusieurs algorithmes de clustering, produisant principalement deux types de structures : le clustering hiérarchique et le clustering plat (ou non hiérarchique) [Willet, 1988]. Le premier type d'algorithme essaie de créer une hiérarchie des clusters, les documents les plus similaires étant regroupés dans des clusters aux plus bas niveaux, tandis que les document moins similaires sont regroupés dans des clusters aux plus haut niveaux.

Les algorithmes de clustering peuvent aussi être classés selon le fait qu'ils effectuent un **soft clustering** ou un **hard clustering**. Dans le soft clustering, l'appartenance à plusieurs catégories est permise. Pour cela, un degré d'appartenance à chaque catégorie est déterminé. Dans le clustering dit hard, chaque objet à classer est affecté à une et une seule catégorie.

Parmi les algorithmes de clustering les plus utilisés, citons l'algorithme K-means [McQueen, 1967] et l'algorithme EM (Expectation-Maximisation).

La réduction de la taille de l'espace de classement des documents est un problème crucial dans le clustering. Dans les algorithmes de classification, on considère que chaque mot index correspond à une dimension différente. Pour une langue donnée, il y a souvent des centaines de milliers d'index possibles. Il est impossible dans ce cas d'effectuer une classification de façon efficace. Par ailleurs, ces dimensions ne sont pas toutes significatives pour la classification, car elles contiennent beaucoup de bruit. Comme pour l'indexation des documents, on doit éliminer le plus possible le bruit et les aspects non-significatifs. Parmi les solutions proposées, citons l'utilisation d'une stop-liste, la sélection des dimensions dont la fréquence d'occurrence est parmi les plus fortes pour une classe, etc. Il y a aussi des méthodes, telles que le LSI, les SOMs (Self Organizing Maps) [Kohonen, 1984][Kohonen *et al.*, 2000], qui extraient des caractéristiques à partir des dimensions originales. WEBSOM [Kaski *et al.*, 1998] est un exemple d'approche de clustering utilisant les SOMs.

Certains moteurs de recherche commerciaux utilisent le clustering, soit pour classer les documents issus d'un même site (e.g., Google qui regroupe sous forme hiérarchique les pages provenant d'un même site), soit pour classer les documents en catégories selon leur contenu (e.g., Vivisimo[26], Exalead[27]).

3.3 Modèle à base de ressources linguistiques et de sources externes de connaissances

La question de l'organisation des connaissances d'un domaine s'est d'abord manifestée dans le monde informatique dans le domaine de la recherche d'information. Pour ranger et ensuite interroger un ensemble de documents, il est nécessaire de convenir entre celui qui range (qui indexe) et celui qui interroge, de l'ensemble des termes utilisables.

Les modèles classiques de recherche d'information sont basés sur l'approche sac-de-mots. L'index est représenté par un ensemble de mots-clés présents dans les documents du corpus. Cet ensemble de mots clés n'informe nullement à propos des relations sémantiques qui peuvent exister entre eux. Par ailleurs, les mots peuvent être polysémiques ou avoir plusieurs synonymes, ce qui est source de bruit et de silence pour la recherche d'information. Des pistes d'amélioration de l'approche classique par l'utilisation de sources externes ont été proposées.

Une des tentative est l'utilisation des thésaurus, outils de contrôle de vocabulaire assurant qu'un sujet sera toujours décrit avec les mêmes termes préférentiels toutes les fois qu'il sera indexé. Ils ont été et sont toujours largement utilisés en documentation. L'autre piste est l'utilisation d'organisations plus formelles, les ontologies.

3.3.1 Utilisation des thésaurus

Un thésaurus est un vocabulaire contrôlé, généralement pourvu d'une structure hiérarchique, utilisant des relations associatives en plus des relations de parent-voisin. L'expressivité des rela-

[26] http ://vivisimo.com
[27] http ://www.exalead.fr

tions peut varier mais elle se réduit en général à des relations de type A *est voisin* de B.

L'utilisation des thésaurus en RI pose trois problèmes : leur construction, leur accès par le SRI et leur évaluation afin de prouver leur efficacité [Jing et Croft, 1994]. Selon la façon dont le thésaurus est construit, on distingue les thésaurus constuits manuellement et les thésaurus constuits automatiquement. La première catégorie peut elle-même être subdivisée en deux types : i) les thésaurus généralistes tels que le thésaurus Roget [Kirkpatrick, 1988] et WordNet, basés sur les mots et ii) les thésaurus orientés-termes tels que INSPEC [28] et le MeSH (Medical Subject Headings). Cette dernière catégorie est très souvent utilisée comme vocabulaire contrôlé pour l'indexation et la recherche. Les thésaurus automatiques sont habituellement dépendants de la collection à partir de laquelle ils ont été construits.

WordNet est une base de données lexicale developpée et maintenue par l'université de Princeton depuis 1985 [Fellbaum, 1999]. Elle couvre essentiellement le vocabulaire anglais et est utilisée dans plusieurs travaux de désambiguïsation de termes [Voorhees, 1993][Sanderson, 1994]. WordNet n'est pas conçu pour des applications spécifiques, il présente des limites, notamment dans la couverture du vocabulaire médical [Burgun et Bodenreider, 2001]. C'est pourquoi des initiatives de développement de ressources lexicales spécifiques au domaine de la médecine ont été prises (e.g., Medical WordNet [Fellbaum *et al.*, 2006]).

Les premières utilisations des thésaurus en RI remontent aux travaux de Sparck Jones [Jones et Needham, 1968a][Jones et Jackson, 1970], Salton sur la construction automatique de thésaurus et l'expansion de requête [Salton, 1968][Salton *et al.*, 1983a] et Van Rijsbergen et ses collègues sur la co-occurrence de termes [Rijsbergen *et al.*, 1981]. Les expériences menées dans ces travaux ont montré que sans l'utilisation de la rétroaction de pertinence ou du jugement de pertinence, aucune amélioration de performance n'était obtenue. Cependant, Salton [Salton, 1968] a montré que la combinaison du jugement de pertinence et du thésaurus HARRIS produisait des améliorations significatives.

Un certain nombre de travaux ont utilisé les thésaurus pour des tâches de désambiguïsation en RI. La désambiguïsation est le processus qui consiste à determiner lequel des sens d'un mot donné est invoqué dans un contexte spécifique. Un mot est supposé avoir un nombre fini de sens, souvent donnés par un dictionnaire, un thésaurus ou tout autre source externe de reférence. La tâche de désambiguïsation consiste alors à choisir parmi les différents sens du mot, celui qui est supposé être le bon en fonction du contexte. Parmi les travaux qui ont utilisé un thésaurus pour la désambiguïsation automatique, citons l'exemple de Yarowsky [Yarowsky, 1992] qui a utilisé le thésaurus Roget, tandis que Voorhees [Voorhees, 1993], Richardson et Smeaton [Richardson et Smeaton, 1995], Gonzalo et ses collègues [Gonzalo *et al.*, 1998] ont utilisé WordNet.

Les expériences menées sur l'effet de l'utilisation des vocabulaires contrôlés sur la performance des SRI montrent des résultats parfois contradictoires [Cleverdon *et al.*, 1966][Tenopir, 1985][Rajashekar et Croft, 1995][Savoy, 2005], l'utilisation des modèles classiques et de l'approche sac-de-mots étant jugée plus performante que l'utilisation de vocabulaire contrôlé assigné manuellement.

3.3.2 L'usage des ontologies en recherche d'information

Généralités

Une ontologie offre une compréhension commune, structurée et partagée d'un domaine ou d'une tâche, qui peut être utilisée pour la communication entre les personnes (indexeur et chercheur d'informations) et les machines. Comparées aux thésaurus, les ontologies sont des structures

[28] http ://www.iee.org/Publish/Inspec/index.cfm

complexes. En plus de la hiérarchie de concepts basée sur des relations d'hyperonymie/hyponymie (Is-A), ou de méronymie (Part-Of), elles peuvent contenir tout type de relation jugé utile ainsi que des contraintes portant sur le domaine concerné.

L'utilisation des ontologies en RI, qui vise à dépasser les limites de la recherche basée sur les mots clés, est un des enjeux actuels du Web Sémantique. Les premiers travaux datent des années 1990 [Guarino, 1999][McGuinness, 1998][Fensel *et al.*, 1999][Heflin *et al.*, 2003]. L'usage des ontologies pour la RI a pris vraiment son envol avec l'avènement du WS, qui préconise une caractérisation sémantique des ressources du Web [Berners-Lee *et al.*, 2000].

Dans les SRI utilisant un modèle basé sur une ontologie, l'index est construit sur la base des concepts présents dans l'ontologie et non sur la base des mots présents dans les documents. On parle d'indexation conceptuelle basée sur une ontologie. Comparé à un index basé sur les mots-clés, un index basé sur les ontologies présente deux avantages [Kabel *et al.*, 2004] :

1. la présentation des résultats de recherche peut s'effectuer suivant les catégories présentes dans l'ontologie ;
2. la formulation et le raffinement de requêtes peuvent reposer sur un vocabulaire structuré fourni par l'ontologie.

L'approche par ontologie permet de répondre à des requêtes *classiques*, telles que celles offertes par l'approche par mots-clés, mais aussi des interrogations *ontologiques*, faisant appel à des traitements exploitant les propriétés (relations) spécifiées dans l'ontologie. Un exemple d'expression d'une interrogation de cette catégorie peut être *Trouver tous les documents relatifs au concept C et tous les documents relatif au concept C' lié à C par la relation R*, mais on peut avoir aussi des expressions plus structurées (à la SQL) en utilisant des langages d'interrogation basés sur RDF (e.g., SPARQL [Perez *et al.*, 2006]). L'utilisation des ontologies pour l'indexation permet d'envisager l'apparition d'une nouvelle génération de SRI capables de répondre à des requêtes de la forme :

> Retrouver les documents de tous les x, y , z tel que
> x est un Patient, y est un Médecin, z est un Hôpital
> x a comme N° dossier "ABC", y a comme nom "Renault",
> z a comme nom "CHU de La Tronche"

Dans le cas de la requête ci-dessus, le SRI à base d'ontologies est censé éliminer tout document mentionnant les voitures de marque Renault puisqu'ici *Renault* est le nom d'un patient et non la marque de voiture. Notons toutefois que ce type de requête suppose deux choses : i) l'utilisation d'un langage spécifique pour l'interrogation, ou tout au moins, l'identification par le SRI des constituants de la requête si elle est exprimée en langue naturelle ; ii) que les documents aient été au préalable correctement indexés, i.e, que les instances de la ou des ontologies utilisées aient été correctement identifiées. Ceci pose bien évidemment plusieurs problèmes qui sont d'actualité dans la problématique des SRI à base d'ontologies.

L'un des principaux problèmes à considérer pour la recherche d'information à base d'ontologies est l'identification des (instances de) concepts dans les documents à indexer, l'enjeu étant de ne pas "louper" des concepts pertinents pour la représentation des documents. Les documents contiennent du texte en langage naturel et non des concepts. En conséquence, le processus d'identification des concepts passe par l'analyse des phrases du document. Ainsi, il peut se faire soit manuellement, soit de façon automatique, soit de façon semi-automatique. Le processus manuel suppose l'intervention d'un expert du domaine concerné ; cette approche est peu envisageable pour de très grandes collections à traiter. Le processus automatique passe par l'utilisation de

techniques d'extraction automatisée d'informations [Cunningham, 2005][Bontcheva, 2004], qui incluent un processus de désambiguïsation automatique [Dill *et al.*, 2003]. L'identification semi-automatique utilise une première phase d'identification, automatisée, et une seconde phase de correction manuelle des erreurs éventuelles générées par la phase précédente [Diallo *et al.*, 2006a]. L'approche que nous préconisons pour la représentation conceptuelle de sources textuelles utilise l'approche semi-automatique. Elle est détaillée au chapitre 6 de ce mémoire.

L'autre problème est l'ordonnancement des résultats obtenus par une requête, et plus généralement la définition de la notion de pertinence de documents (ou entités) par rapport à une reqûete, quand une ou plusieurs ontologies sont utilisées. Certains SRI basés sur les concepts retournent des entités (instances de concepts) en réponse à une requêtes (e.g., KIM [Popov *et al.*, 2003]) tandis que d'autres, plus proches de l'approche classique, estiment la pertinence des documents en utilisant une fonction de similarité basée sur les concepts [Vallet *et al.*, 2005].

SRI utilisant des ontologies

Selon la manière de formuler les requêtes, nous distinguons quatre catégories de SRI basés sur les concepts [Lei *et al.*, 2006] :

1. les moteurs de recherche basés sur les **formulaires**, utilisant des interfaces d'interrogation sophistiquées. Les interfaces permettent aux utilisateurs de spécifier des requêtes en choisissant les ontologies, les classes, les propriétés et valeurs à utiliser ;
2. les moteurs de recherche utilisant des **langages d'interrogation basés sur RDF**, qui permettent d'exprimer des requêtes complexes ;
3. les moteurs de recherche sémantiques utilisant les **mots-clés** : ils améliorent la performance des techniques traditionnelles basées sur les mots-clés par l'utilisation des données sémantiques disponibles (métadonnées) ;
4. enfin les systèmes de **Question/Réponse**, qui permettent de répondre à des interrogations précises.

Le système OntoSeek de Guarino [Guarino, 1999] (1996-1999) fut l'une des premières expériences d'utilisation des ontologies pour la RI. Il est dédié au traitement des pages jaunes et les catalogues de produits en ligne. OntoSeek utilise l'ontologie linguistique SENSUS – qui est une extension et une réorganisation de WordNet – pour décrire à la fois les requêtes des utilisateurs et les descriptions des ressources. Guarino et ses collègues ont montré que l'utilisation de SENSUS a permis d'améliorer les performances du SRI (tout au moins pour leur domaine d'étude, les pages jaunes). Corese [Corby *et al.*, 2006] est un moteur de recherche sémantique développé à l'INRIA, dont le modèle interne est basé sur les graphes conceptuels, bien que l'interrogation s'effectue en utilisant un langage construit sur le modèle des triplets RDF. L'outil utilise une ontologie, représentée en RDF, qui permet d'annoter les ressources à traiter. L'une des originalité de l'outil est l'introduction de la notion de distance sémantique au sein d'une ontologie pour l'évaluation des requêtes. Roussey et ses collègues [Roussey *et al.*, 2001a][Roussey *et al.*, 2001b] présentent une approche basée sur l'utilisation des graphes conceptuels [Sowa, 1984] pour la description sémantique de documents multilingues.

Kim [Kim, 2005] décrit ONTOWEB, un système de recherche sur le Web basé sur une ontologie. Ce système permet une recherche sémantique sur les sites des organisations internationales comme la banque mondiale [29] ou l'OCDE [30]. Une comparaison des performances de ONTOWEB

[29] http ://www.banquemondiale.org/
[30] www.oecd.org

et d'autres moteurs de recherche classiques sur internet en termes de pertinences des résultats et de temps passé pour effectuer une recherche fructueuse a montré qu'ONTOWEB permet, en plus de l'amélioration de la précision des résultats, de réduire le temps de recherche.

Rappelons, dans le cadre du Web sémantique le système FindUR [McGuinness, 1998], développé par McGuiness au laboratoire AT&T, qui améliora considérablement le rappel et la précision, tout en offrant un support à la formulation des requêtes, l'initiative $(KA)^2$ de Benjamins et Fensel [Benjamins et Fensel, 1998], la plateforme KIM [Kiryakov *et al.*, 2003][Popov *et al.*, 2003].

Les évaluations des SRI à bases d'ontologies montrent globalement que ces derniers, en plus de l'amélioration de la performance dans les résultats, permettent de réduire le temps de recherche [Sure *et al.*, 2002][McGuinness, 1998][Guarino, 1999].

L'usage des ontologies en RI a un lien fort avec la notion d'annotation (sémantique) de ressources, que nous présentons dans la section suivante.

3.4 Annotation de ressources textuelles

Les ressources textuelles des organisations contiennent des informations qui sont souvent complémentaires de celles gérées par le Système d'Information de l'entreprise. Cependant, ces ressources, de par leur manque de structure, sont plus difficiles à exploiter que les bases de données ou les documents XML. En ajoutant des métadonnées (des données à propos de données) sur les documents, l'annotation permet de mieux expliciter leur contenu, en les rendant plus exploitables, par exemple par les moteurs de recherche.

3.4.1 Les différentes acceptions

On constate que le terme *annotation* est utilisé par différentes communautés et peut désigner plusieurs types de pratiques : l'annotation de catalogage, l'annotation textuelle et l'annotation sémantique.

Annotation de catalogage

Dans cette acception, une annotation consiste en l'ajout d'information sur les documents au moyen de méta-données qui servent à décrire le type d'informations que les bibliothécaires mettent depuis toujours dans les catalogues. Ce sont des informations descriptives comme l'auteur, le titre, la date de création ou de publication, etc. Dans ce domaine, la norme du Dublin Core (DC), qui est un ensemble d'éléments simples permettant de décrire une grande variété de ressources en réseau, fait l'objet d'un consensus international [Hillman, 2001]. CisMef[31], portail Français pour l'indexation et la recherche de ressources médicales, utilise un catalogue spécialisé et des méta-données, dont celles de Dublin Core, pour décrire et indexer les ressources médicales.

La norme de métadonnées DC est constituée de 15 éléments résultant d'un consensus international de professionnels de diverses disciplines telles que l'informatique ou la bibliothéconomie. Les éléments peuvent être optionnels ou répétés, et peuvent avoir un ensemble limité de qualificatifs, ce qui permet de les raffiner. Dublin Core tente de concilier les caractéristiques suivantes :

- La simplicité de création et de gestion.
- Une sémantique communément comprise.
- Une portée internationale (versions dans plusieurs langues).

[31] http ://www.chu-rouen.fr/cismef

- L'extensibilité : DC a été conçue de telle sorte que l'ensemble de ses éléments puisse être étendu.

Les éléments de base peuvent être classés en trois types :

- ceux liés au contenu de la ressource, comme Description ou le Type du document ;
- ceux liés à la propriété intellectuelle, comme le Créateur ou l'Editeur ;
- ceux liés à une instance particulière de la ressource, comme la Date ou le Format de la ressource ;

En juillet 2000 l'IMDC (Initiative de Métadonnées de Dublin Core) a émis la liste recommandée de qualificatifs de DC.

Annotation textuelle

La notion d'annotation textuelle est très proche du processus d'annotation réalisé lors de la lecture d'un document papier. Il s'agit de capturer les remarques, les commentaires, le jeu de questions/réponses qui qualifient toute activité scientifique. L'annotation textuelle, réalisée à travers un outil logiciel, a pour objectif de satisfaire trois besoins essentiels d'une communauté de pratiques :

1. éditer : La plupart du temps les diverses réflexions instantanées sur un dossier particulier ou un cas spécifique sont perdues une fois le dossier clôturé ou le cas traité. Il s'agit de proposer un outil dont l'ergonomie est axée sur la rapidité et la simplicité d'édition ainsi que sur le partage immédiat de l'information.
2. archiver : Dans une organisation comportant plusieurs services, il est important de centraliser l'information collectée afin de la diffuser selon des schémas reposant sur des profils utilisateurs prédéfinis. Grâce à l'outil d'annotation, chaque acteur peut interagir et contribuer en temps réel à l'élaboration d'une connaissance partagée. Cette connaissance peut être vue comme un réservoir d'informations dynamiques.
3. diffuser : Une communauté active partage un ensemble de ressources documentaires. L'outil d'annotation participe au processus de sélection des ressources et permet de mutualiser les efforts de chacun afin de rendre la communauté plus efficace.

Annotea est une initiative du W3C, qui fut l'un des précurseurs dans la mise en oeuvre de la notion d'annotation textuelle partagée [Kahan *et al.*, 2002]. Le projet Annotea, dont le développement est aujourd'hui abandonné, a défini un format XML pour le stockage des annotations textuelles, qui fait office de standard. Il est repris dans de nombreux projets d'annotation textuelle, dont celui développé dans le cadre du projet européen Noesis dans lequel notre équipe est impliquée [Patriarche *et al.*, 2005].

Annotation sémantique

Le Web Sémantique préconise l'annotation des documents en utilisant un vocabulaire provenant d'une ontologie de domaine [Berners-Lee *et al.*, 2000]. Dans cette acception, le processus d'annotation permet d'attacher au document, ou à des parties du document, des concepts issus d'une structure formelle, représentée le plus souvent par des classifications ou des ontologies. On pourrait à ce niveau la confondre avec l'indexation. L'annotation sémantique évolue vers une forme d'indexation fine, qui permet une recherche plus précise dans le contenu d'un document.

L'annotation sémantique est prioritairement destinée à une utilisation par les machines. Elle peut s'effectuer de façon automatique ou de façon manuelle. Lorsqu'elle est effectuée de façon automatique, une phase de désambiguïsation, permettant de corriger d'éventuelles erreurs et de

lever certains ambiguïtés, doit être effectuée. Cette désambiguïsation peut elle-même s'effectuer automatiquement [Dill *et al.*, 2003] ou manuellement. Ainsi, un passage d'un texte peut mentionner le terme *mémoire*, qui peut aussi bien désigner la mémoire (au sens mental) que la mémoire de stockage informatique. C'est à la phase de désambiguïsation qu'on peut décider si *mémoire* est instance de l'une ou l'autre interprétation.

SHOE [Heflin *et al.*, 2003] et ONTOBROKER (ainsi que son successeur On2Broker) [Fensel *et al.*, 1999] font partie des précurseurs dans le domaine de l'annotation conceptuelle de documents HTML utilisant une ontologie. SHOE fournit une extension du langage HTML à travers des balises spécifiques permettant de rajouter des informations de nature sémantique aux pages HTML. La balisage sémantique s'effectue à l'aide du Knowledge Annotator de SHOE après avoir identifié les ontologies pertinentes pour les documents à annoter. ONTOBROKER permet aussi bien d'annoter des documents web avec des métadonnées provenant d'une ontologie, que d'effectuer une recherche (guidée par une ontologie) sur les documents annotés. ONTOBROKER utilise le langage F-logic [Kifer *et al.*, 1995] permettant d'effectuer des inférences complexes.

La plateforme KIM [Popov *et al.*, 2003] s'appuie sur l'ontologie de domaine de KIM [Kiryakov *et al.*, 2003] (qui contient environ 250 concepts et 40 relations) pour annoter sémantiquement des documents textuels. L'annotation sémantique est ici assimilée à l'identification d'entités nommées dans les documents textuels. Les entités nommées sont des noms de personnes, de lieux, d'organisations, etc. Le résultat du processus d'annotation est stocké dans la base de connaissances KIM. Le système de recherche de KIM ne retourne pas des documents, mais des entités nommées telles que des noms de personnes, des localisations géographiques, etc.

L'outil de balisage sémantique SemTag [Dill *et al.*, 2003], développé chez IBM, qui intègre un module de désambiguïsation automatique et qui permet d'identifier notamment des entités nommées, est à notre connaissance le système automatique qui a fait l'objet de la plus importante expérimentation à grande échelle (plusieurs millions de pages web) de balisage sémantique. Il est conçu à partir de l'ontologie TAP[32] développée à l'Université de Stanford.

La notion d'annotation que nous venons de présenter permet une caractérisation plus fine des documents à traiter afin de les retrouver plus facilement. Les outils d'annotation traitent en général de façon séparée les trois formes présentées. Nous présenterons dans le chapitre consacré à la représentation des sources textuelles un outil d'annotation manuelle intégrant les trois formes.

Des métriques sont utilisées pour mesurer d'une part, l'importance (le poids) des descripteurs pour les documents, et d'autre part le degré de correspondance entre chaque document et les requêtes des utilisateurs.

3.5 Les mesures en recherche d'information

3.5.1 La mesure du TF*IDF

La mesure du *tf*idf* [Salton *et al.*, 1983a] est la combinaison du nombre d'occurrences d'un terme t dans un document (tf) d et de la valeur de l'inverse du nombre de documents dans lesquels apparaît le terme t (idf). En effet, la seule mesure du nombre d'occurrences d'un terme dans une collection ne permet pas de connaître sa spécificité. Un terme qui est commun à beaucoup de documents est en effet moins utile pour la recherche qu'un terme qui n'apparaît que dans peu de documents. Ce constat a motivé la combinaison de la simple mesure du tf et de la simple mesure de l'idf. En bref, le tf mesure l'importance d'un terme dans un document tandis que l'idf mesure sa spécificité dans toute la collection. La formule du *tf*idf* s'écrit :

[32] http ://tap.stanford.edu

$$w_{i,j} = tf_{i,j} log(\frac{N}{df_i}) \tag{3.1}$$

Où :

1. $w_{i,j}$ est le poids du terme i dans le document j ;
2. $tf_{i,j}$ est la fréquence d'apparition du terme i dans le document j ;
3. $log(\frac{N}{df_i})$ est l'*idf*,df_i est le nombre de documents contenant i et N est le nombre de documents de la collection.

3.5.2 La mesure de la similarité entre documents et requêtes

Etant donné une requête et une collection de documents, pour obtenir un ensemble de réponses il est nécessaire de comparer la requête aux documents. Plusieurs mesures de similarité ont été proposées [Salton, 1971][Robertson, 1997]. Le modèle vectoriel, par exemple, représente la requête par un vecteur, dans le même espace, que les documents. De cette façon il est possible de mesurer la similarité entre le vecteur de la requête et les vecteurs des documents en utilisant une technique algébrique simple. La mesure la plus utilisée dans ce modèle est la valeur du cosinus de l'angle entre le vecteur de la requête et le vecteur de chaque document. Cette mesure a l'avantage d'être facile à mettre en oeuvre. Si les vecteurs sont normalisés, le cosinus se calcule par la formule suivante :

$$cos(\overrightarrow{q}, \overrightarrow{d}) = \frac{q * d}{\| q \| \| d \|} \tag{3.2}$$

où $\overrightarrow{q}$ représente le vecteur de la requête et $\overrightarrow{d}$ représente le vecteur du document.

3.6 Évaluation en recherche d'information

L'évaluation est une tradition ancrée dans le monde de la recherche d'information, et ce depuis son origine. Afin d'évaluer les performances d'un système ou d'une technique de recherche d'information, on se base sur une collection standard de test. Une collection test consiste en une base de documents, un ensemble de thèmes (topics) de recherche et un ensemble de jugements de pertinences, qui lient un certain nombre de documents de la base à chaque thème de recherche. Partant de cette collection, la performance est évaluée en posant une requête (ou une série de requêtes) pour chaque thème de recherche, et en mesurant l'efficacité à certains points fixes. La performance moyenne est ensuite obtenue en effectuant une moyenne pour tous les thèmes de recherche. Ainsi, les résultats obtenus par un système (ou une méthode) A sont comparés à ceux obtenus par un système (ou une méthode) B.
L'évaluation peut être classée en [Kazai *et al.*, 2003] :

1. l'évaluation système, qui évalue le système (ou la méthode) de recherche en terme de temps de traitement, d'ingénierie du SRI, de présentation des résultats, etc. ;
2. l'évaluation centrée-utilisateur qui évalue par exemple l'effort de l'utilisateur pour obtenir l'information souhaitée.

La mesure du rappel et de la précision : Pour mesurer la performance d'un système ou d'une méthode de recherche, des métriques et des mesures sont nécessaires. Les plus utilisées sont la mesure du *Rappel* et de la *Précision*. Le rappel correspond "*à la capacité du système à retourner un nombre de documents pertinents le plus élevé possible*" tandis que la précision

indique "*la capacité du système à retourner le moins possible de documents non pertinents*". Si nous considérons que R est le rappel et P la précision, les formules habituellement utilisées pour calculer ces deux mesures s'écrivent :

$$R = \frac{\mid Pert \mid \bigcap \mid Ret \mid}{\mid Ret \mid} \tag{3.3}$$

et

$$P = \frac{\mid Pert \mid \bigcap \mid Ret \mid}{\mid Pert \mid} \tag{3.4}$$

où *Pert* désigne les documents pertinents retournés et *Ret* désigne les documents retournés. La précision est calculée à certains points du rappel (entre 0 et 1). Ce qui donne la courbe de *rappel/précision.*

Les mesures combinées Les mesures du rappel et de la précision permettent d'évaluer une performance partielle d'un SRI. Ceci a amené certains chercheurs à proposer des mesures combinées qui permettent de mieux évaluer la performance globale d'un SRI.

La **mesure F (F-measure)** et **la mesure E (E-measure)** : Van Rijsbergen [van Rijsbergen, 1979] a proposé de combiner le rappel et la précision dans la F-measure et la E-measure (qui est simplement 1 - F-measure), qui se calculent comme suit :

$$F = \frac{1}{\alpha \frac{1}{P} + (1 - \alpha) \frac{1}{R}} \tag{3.5}$$

et

$$E = 1 - F \tag{3.6}$$

où α est un facteur qui détermine l'importance relative du rappel et de la précision. Avec ces mesures, l'utilisateur peut pondérer les valeurs de précision et de rappel. La valeur $\alpha = 0.5$ est souvent choisie, ce qui donne un poids égal à la précision et au rappel. La formule de la F-measure se réduit dans ce cas à *2PR/(R + P).*

Les SRI sont communément tester à travers des campagnes d'évaluation qui offrent un moyen d'avoir une évaluation comparative des modèles, des technologies et des méthodes mise en oeuvre pour solutionner le problème de la RI.

3.6.1 La campagne TREC

Le programme américain TREC (Text REtrieval Conference)[33] a pour objectif de tester des méthodes et des SRI en utilisant des collections de très grande taille. Il a démarré en 1992 [Harman, 1992] et en est à sa 15^{ime} édition. Les tâches (tracks dans le jargon TREC) changent régulièrement tout en reflétant les besoins des chercheurs. Au fil des années, il y a eu la RI ad hoc (soumettre des requêtes sur une collection statique), le filtrage de l'information, la RI multimédia, la RI translinguistique (retrouver des documents dans une langue différente de celle de la requête), etc.

[33] http ://trec.nist.gov

3.6.2 La campagne CLEF

Le programme CLEF (Cross Language Evaluation Forum)[34] est une initiative récente de la communauté européenne (2000), coordonné en Europe par DELOS (Network of Excellence for Digital Libraries), et organisé en collaboration avec l'institut américain NIST (National Institute of Standards and Technology)[35] et le programme TREC.

Les campagnes d'évaluation de CLEF ont pour objectif de promouvoir la recherche et le développement dans le domaine de la recherche d'information multilingue (Cross-Language Information Retrieval, CLIR), d'une part en offrant une infrastructure pour tester et évaluer les SRI sur des supports écrits dans les différentes langues européennes, en mode monolingue, multilingue ou interlangue, et d'autre part en mettant au point des séries de tests composés de données qui peuvent être réutilisées par les développeurs de systèmes, pour l'évaluation.

3.7 Conclusion

Dans ce chapitre, nous avons fait un rappel des notions et concepts en RI et nous avons rappelé les principaux travaux qui ont marqué cette discipline. Nous avons décrit les principaux modèles utilisés par les SRI et nous avons présenté succinctement l'utilisation des ontologies par les SRI ainsi que la notion d'annotation sémantique.

Lors de l'utilisation d'un SRI, on a souvent un grand nombre de documents en réponse à une requête, et il appartient à l'utilisateur de trouver dans ces documents l'information qu'il recherche, ce qui peut être fastidieux. Cette problématique a motivé les recherches sur les systèmes Question/Réponse. Ainsi, l'extraction des entités nommées prend de l'importance, et les moteurs de recherche, comme Google[36], commencent à s'y intéresser.

De nombreuses études ont porté sur des améliorations possibles des techniques d'indexation et de recherche. Parmi les tentatives les plus marquantes on retrouve notamment :

1. La rétroaction de pertinence ou reformulation de requête (relevance feedback), qui vise à étendre la portée de la recherche en intégrant des termes issus soit des documents pertinents, soit des documents en tête de la liste de réponse trouvée automatiquement. La rétroaction de pertinence fut proposée pour la première fois par Rocchio [Rocchio, 1971]. Son idée était basée sur le fait que l'utilisateur était le mieux placé pour juger de la pertinence des documents par rapport à sa requête. De nouvelles techniques de reformulation en l'absence de l'intervention de l'utilisateur, connues sous le nom d'expansion de requêtes, telles que la pseudo-rétroaction de pertinence, furent proposées dans les années 1990 [Buckley *et al.*, 1995].
2. L'expansion de requête vise à renforcer l'expression de la requête de l'utilisateur (qui est souvent très courte, environ 2 à 3 mots[37], ou mal exprimée) par l'intégration d'autres termes reliés. Ces termes reliés peuvent être obtenus soit en exploitant un thésaurus, soit en utilisant un calcul basé sur des co-occurences de termes.

Les systèmes classiques ne "comprennent" pas les requêtes qu'ils reçoivent, et ne "comprennent" pas non plus les résultats qu'ils renvoient. La pertinence des réponses peut donc être sérieusement améliorée grâce aux outils sémantiques et à l'utilisation des techniques d'indexation conceptuelle. C'est là l'un des objectifs du Web Sémantique. L'indexation conceptuelle peut notamment être

[34] http ://www.clef-campaign.org/
[35] www.nist.gov/
[36] http ://local.google.com/
[37] Etude menée par AOL, 2006

une réponse aux problèmes que pose l'utilisation d'une langue différente de la langue du corpus lors de la recherche. En effet avec une indexation basée sur des termes simples, avec une requête en français on ne peut trouver que des document en français, à moins d'une coïncidence ou l'utilisation d'un dictionnaire pour la traduction des termes de la requête.

Il faudra certainement encore plusieurs années pour que la sémantique – tout en ayant fait ses preuves – achève de révolutionner les méthodologies de recherche. L'utilisation de la sémantique dans la recherche d'information sera abordée au chapitre 6.

Deuxième partie

Architecture à base d'ontologies pour la gestion de données structurées et non structurées

Chapitre 4

Une architecture à base d'ontologies pour la gestion unifiée de données structurées et non structurées

4.1 Introduction

Ce chapitre est consacré à la présentation générale de notre approche pour la gestion conjointe de données gérées par des bases de données relationnelles et de données de nature textuelle. Nous abordons également l'approche de représentation virtuelle des sources de données structurées qui est utilisée dans la cadre du système d'intégration.

Notre objectif est de proposer une architecture pour la gestion unifiée prenant en compte, de façon explicite, des sources non structurées pouvant être multilingues.

4.2 Modèle pour le système de gestion unifiée

Le système d'intégration que nous proposons suit le modèle classique d'intégration avec un schéma global [Lenzerini, 2002], avec la prise en compte explicite de sources textuelles. Le modèle est un quadruplet $\{O_{PI}, S_{local}, O, M\}$ avec [Diallo *et al.*, 2006b] :

1. O_{PI}, l'ontologie pour la perspective d'intégration est l'ontologie globale du système. Elle utilise un vocabulaire A_{global} ;
2. S_{local} est l'ensemble des ontologies locales, construites à partir des schémas locaux. Il est exprimé à l'aide d'un vocabulaire A_{local}. Cet ensemble est l'union des ontologies locales représentant les sources non structurées, notées S_{ns}, et des ontologies représentant les sources structurées, notées S_s. Ainsi $S_{local} = S_s \cup S_{ns}$;
3. $O = \bigcup O_{i(i=1..p)}$ est l'ensemble des ontologies pour la caractérisation sémantique des sources locales ;
4. M est l'ensemble des correspondances entre l'ontologie globale et l'ensemble des ontologies locales S_{local}.

Partant de ce modèle, nous avons défini une architecture pour le système de gestion unifiée que nous proposons. Nous présentons dans la figure 4.1 l'architecture globale du système de gestion unifiée.

4.3 Architecture générale du système

4.3.1 Motivations pour le choix de l'architecture

Un certain nombre de considérations ont été prises en compte pour la proposition de l'architecture de gestion unifiée :

1. pour un même ensemble de sources à intégrer, plusieurs besoins d'informations symbolisés par différents points de vue, sont à considérer. En effet, les besoins d'information du chef d'un service hospitalier, par exemple, ne sont pas les mêmes que ceux d'un(e) secrétaire de service ;
2. l'autonomie des sources à intégrer doit être préservée. Cette considération est classique dans le cadre de l'intégration ;
3. chaque source non structurée à intégrer doit être caractérisée sémantiquement par un ensemble d'ontologies. Ces ontologies peuvent être communes à plusieurs sources. Les ontologies ne seront pas stockées localement, mais seront référencées par leur identifiant universel (leur URI – Uniforme Resource Identifier) ; les sources non structurées à intégrer peuvent être multilingues ;
4. chaque source à intégrer doit être représentée par une ontologie servant à la décrire. Cette ontologie est l'unique moyen d'accès à la source dans le cadre du système intégré ;
5. une intervention humaine dans le processus d'intégration est nécessaire, le processus d'intégration ne pouvant s'effectuer de façon entièrement automatique.

Nous détaillons les différentes composantes de l'architecture dans la section suivante.

4.4 Détail des différentes couches

4.4.1 Le premier niveau : les sources

Le premier niveau de l'architecture est constitué des sources concernées par l'intégration. Elles sont de deux types. Les sources constituées exclusivement de documents textuels (e.g., la base des comptes rendus médicaux, une collection de publications scientifiques) et les sources de nature structurée (e.g., bases de données de patients, bases de données génomiques). Les premières nécessitent un prétraitement lors de leur prise en compte dans l'infrastructure de gestion unifiée. Notre objectif est en effet de pouvoir poser sur ces sources des requêtes portant sur leur contenu. Nous utilisons donc un processus de représentation sémantique de documents pour chacune de ces sources. La seconde catégorie de sources ne nécessite aucune opération de prétraitement (sources décrites à l'aide du modèle relationnel).

4.4.2 Le second niveau : la représentation structurée de sources textuelles

Ce niveau est constitué, pour chaque source non structurée (source textuelle), du résultat du processus de prétraitement. Le processus de prétraitement de sources textuelles est détaillé au chapitre 6. Chaque représentation structurée de textes est décrite selon le modèle relationnel, ce qui fournit un premier niveau d'homogénéité entre les sources textuelles et les sources structurées. Rappelons que, classiquement, les systèmes d'intégration ne s'occupent pas de l'étape de prétraitement de sources pour obtenir un modèle commun de représentation des sources à intégrer [Parent et Spaccapietra, 1996].

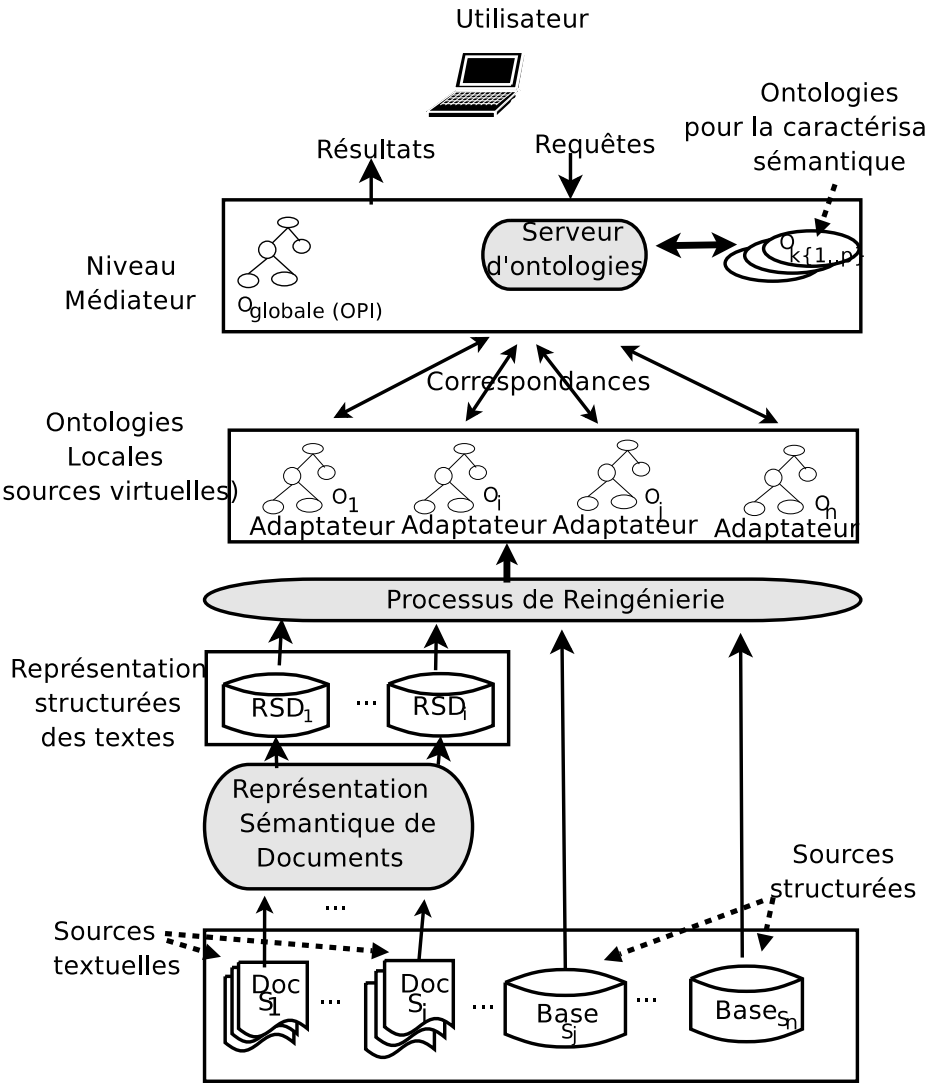


FIG. 4.1 – Architecture générale du système de gestion unifiée

4.4.3 Le troisième niveau : les sources virtuelles ou ontologies locales

Une fois que nous avons obtenu le modèle commun de représentation de toutes les sources à intégrer, nous proposons de décrire à l'aide d'une ontologie les entités de chaque source qui sont nécessaires pour le besoin d'intégration en cours. Rappelons que dans, notre contexte, l'interrogation du système de gestion unifiée doit s'effectuer à l'aide d'une ontologie. Nous avons donc préconisé une représentation "ontologique" de chaque source pour rester homogène avec le niveau global. Cette représentation est constituée des ontologies locales, qui sont articulées avec l'ontologie globale à travers un ensemble de correspondances. Ainsi les requêtes posées à travers l'ontologie globale sont traduites en termes de requêtes sur les ontologies locales à travers les correspondances qui ont été définies. Cependant, compte tenu du fait que nous ne disposons à ce niveau que des représentations des sources (sans les données qu'elles contiennent), les requêtes sur chaque ontologie locale doivent être traduites en requêtes sur la source physique concernée. Ce travail se fait à l'aide des adaptateurs. Notons que dans le cadre de notre travail de thèse notre objectif n'est pas de traiter le problème de réécriture de requêtes et le problème d'établis-

sement des correspondances entre des ontologies [Kalfoglou et Schorlemmer, 2005][Choi *et al.*, 2006]. Toutefois, notre travail bénéficera naturellement des avancées faites dans ces domaines.

4.4.4 Le quatrième niveau : le niveau médiateur

Le niveau médiateur représente le point d'entrée du système puisqu'il contient l'ontologie pour la perspective d'intégration, c'est-à-dire l'ontologie globale utilisée pour l'interrogation du système. Il assure l'interrogation du système et les différentes ontologies utilisées pour la caractérisation sémantique de sources textuelles sont déclarées à ce niveau. Ces ontologies sont gérées à travers un serveur d'ontologies qui permet de les interroger ; il assure le support d'interrogation du système de médiation. Au niveau du médiateur, deux primitives, essentielles pour la prise en compte de l'aspect évolutif des ontologies, sont définies : AssignOnto et RemoveOnto, qui ont respectivement pour rôle de déclarer ou supprimer une ontologie de caractérisation sémantique pour une source et de mettre à jour du serveur d'ontologies (qui fera l'objet du chapitre 7).

L'approche de gestion unifiée que nous proposons conduit à considérer plusieurs catégories d'ontologies. Chaque catégorie joue un rôle particulier dans l'infrastructure. Nous avons évoqué ci-dessus respectivement l'ontologie pour la perspective d'intégration, les différentes ontologies locales et enfin les ontologies pour la caractérisation sémantique de sources textuelles. Nous les détaillons dans la section suivante.

4.5 Les différents types d'ontologies utilisées dans le processus d'intégration

4.5.1 Ontologie pour la perspective d'intégration

L'interaction avec le système de gestion unifiée se fait à travers une ontologie de domaine décrite dans le langage OWL et modélisant les entités qui décrivent le domaine couvert par le besoin d'intégration en cours. A la différence de l'approche classique présentée dans l'état de l'art consacré à l'intégration de données, ce n'est pas l'ontologie de domaine de l'organisation que nous considérons comme l'ontologie globale pour l'intégration. En effet, nous préconisons, pour chaque besoin d'intégration, de tenir compte de façon explicite des sources à intégrer. La mise en place de l'ontologie pour l'intégration prend en considération les éléments des sources (concepts, attributs, ...). En particulier l'ontologie contient sous partie dediée à la gestion des sources textuelles (les éléments sont issus du schéma implémenté pour la représentation hybride des textes)[38]. Cette sous partie de l'ontologie globale reste la même quelque soit la source textuelle à intégrer. Le serveur d'ontologies étant mis-à-jour si de nouvelles ontologies caractérisent la nouvelle source.

4.5.2 Ontologies pour la représentation des sources à intégrer

Chaque source concernée par le processus d'intégration est vue à travers une ontologie locale qui décrit le contenu de la source en question et de la manière d'y accéder. Chaque ontologie locale est décrite dans le langage OWL. Nous détaillons à la section 4.7 notre approche de construction des ontologies locales.

[38] Un exemple simplifié d'une ontologie golable en OWL pour le cas du cerveau est disponible à l'adresse `http://www-timc.imag.fr/Gayo.Diallo/Resources/Ontologies/OIPCerveau.owl`

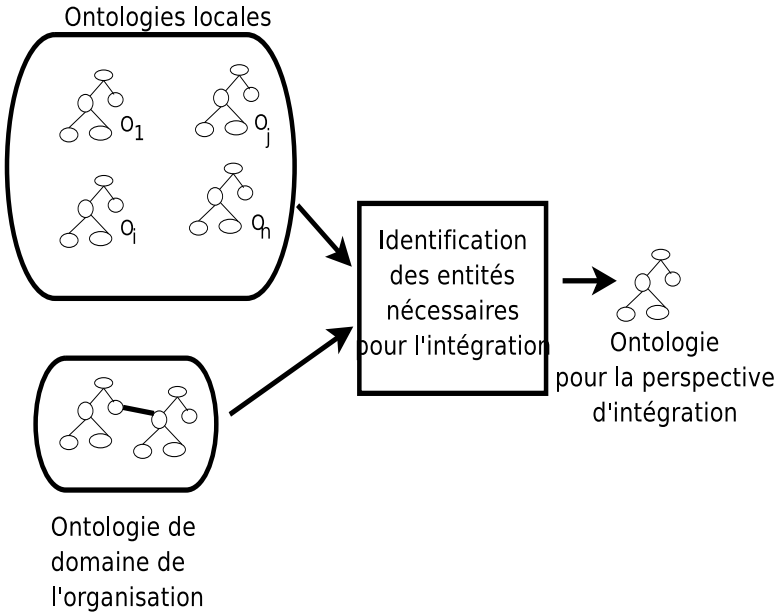


FIG. 4.2 – Construction de l'ontologie globale pour l'intégration

4.5.3 Ontologies pour la caractérisation sémantique de sources

Nous prenons en compte le fait que de nos jours des efforts considérables ont été faits pour rendre disponibles des ontologies et des vocabulaires contrôlés, en particulier dans le domaine de la médecine, qui est notre premier champ d'application. Citons le thésaurus MeSH, la Gene Ontology, UMLS, SNOMED-CT, etc. Ces ontologies et autres vocabulaires contrôlés peuvent être soit utilisés directement pour l'annotation sémantique de ressources, soit servir à la construction de nouvelles ontologies (voir chapitre 5).

Nous considérons que l'utilisation de plusieurs ontologies pour la caractérisation sémantique d'une même source peut offrir des vues complémentaires sur la source en question, car les aspects abordés par les ontologies peuvent différer. Soulignons à titre d'exemple le cas d'une source sur le cerveau, que l'on peut caractériser à la fois selon l'aspect fonctionnel et selon l'aspect anatomique.

Selon la classification des ontologies donnée au chapitre (ontologies), pour le besoin de caractérisation sémantique automatique les ontologies terminologiques peuvent suffire.

Naturellement l'utilisation de plusieurs ontologies pour la caractérisation sémantique pose le problème de leur gestion.

4.6 Changement de perspective sur les sources

Certaines ontologies pour la caractérisation sémantique des sources peuvent ne pas être disponibles à un moment donné. Cette indisponibilité peut être due à des besoins de maintenance de l'ontologie en question. Nous préconisons donc de momentanément "inhiber" cette ontologie afin de ne pas la rendre utilisable par le serveur d'ontologies, ce qui empêche l'interrogation des sources déclarées pour cette ontologie. La manière de percevoir les sources de données peut varier au cours du temps. Nous parlons de changement de perspective sur les sources de données. Le changement de perspective s'effectue grâce aux deux primitives *AssignOnto* et *RemoveOnto*. Etant donné une source S et une ontologie de caractérisation sémantique O, *AssignOnto* déclare

l'ontologie O pour la source S, tandis que RemoveOnto supprime l'ontologie O pour la source S. Le serveur d'ontologies est mis à jour au travers de ces deux primitives.

Notons que les caractérisations sémantiques effectuées à travers une ontologie, qui subit par la suite un certain nombre de modifications, peuvent devenir obsolètes. Une nouvelle caractérisation des sources concernées par cette ontologie est donc envisagée. On voit ici l'avantage de l'utilisation des ontologies locales, distinctes des ontologies pour la caractérisation sémantique. En effet les correspondances éventuelles établies avec l'ontologie globale restent inchangées. Si nous utilisions les ontologies pour la caractérisation sémantique comme ontologies locales, les modifications de ces ontologies auraient pu affecter les correspondances, nécessitant donc des mises-à-jour.

Nous détaillons à présent le processus de représentation des sources pour l'obtention des ontologies locales.

4.7 Représentation virtuelle de sources à intégrer

Nous détaillons dans cette section le processus de représentation d'une source à intégrer dans le cadre de l'approche de gestion unifiée que nous proposons. Nous avons indiqué ci-dessus que chaque source à intégrer est représentée par un ensemble de vues fournissant l'ensemble des informations utiles dans le contexte d'intégration courant. Les liens entre le niveau médiation et le niveau des sources se font à travers les correspondances établies entre le schéma global d'intégration (l'ontologie globale) et les ontologies locales représentant les sources. L'objectif est de pouvoir établir une correspondance entre ontologies. Il faut donc disposer d'une représentation "ontologique" de chaque source structurée (relationnelle). Cette représentation ontologique sera décrite dans le langage OWL/RDF afin d'être homogène avec la représentation de l'ontologie globale et permettre son interrogation à travers les langages d'interrogation du Web Sémantique.

La mise à disposition des bases de données relationnelles dans l'infrastructure du Web Sémantique suscite actuellement un fort engouement [Stephen Harris, 2005][Gregory Karvounarakis, 2001][Pan et Heflin, 2003][Bizer, 2003][An *et al.*, 2006][Petrini et Risch, 2004][de Laborda et Conrad, 2005]. Cependant, l'objectif premier de la plupart des travaux dans ce domaine est de disposer rapidement d'une représentation sous forme de triplets RDF des données de la base en vue de leur interrogation avec des langages du Web Sémantique (e.g., SPARQL)[Stephen Harris, 2005][Gregory Karvounarakis, 2001][Pan et Heflin, 2003]. Cette démarche pose le problème de duplication des données, et elle n'est pas acceptable pour une approche d'intégration basée sur la médiation. L'utilisation d'une ontologie de domaine comme schéma pivot a été adoptée par le système Kaon Reverse [39]. La difficulté est dans ce cas de trouver la bonne ontologie de domaine, qui fera office de schéma pivot. L'utilisation d'un schéma générique de représentation d'une base de données décrit en OWL/RDF a fait son apparition très récemment [de Laborda et Conrad, 2005]. De Laborda et Conrad utilisent un schéma OWL générique qui est instancié pour toute base de données relationnelle à représenter. Nous adoptons une approche similaire tout en prenant en compte les informations concernant la connexion à la base. Notre but premier n'est pas de pouvoir disposer des données contenues dans une base relationnelle dans un format compatible avec le Web Sémantique, mais de pouvoir accéder à ces données résidentes dans leur format d'origine (base relationnelle) à travers des outils du Web sémantique. L'accès à la base se fait au travers des vues qu'elle offre.

Ainsi, ce qui nous intéresse en premier est la représentation dans un langage du Web Sémantique du schéma de toute base relationnelle, et ensuite de pouvoir effectuer des interrogations

[39] http ://kaon.semanticweb.org/alphaworld/reverse

de ce schéma afin de récupérer les données résidant dans la source d'origine. Pour cela, il faut disposer de :

1. un schéma virtuel de description de toute base relationnelle ;
2. l'implémentation de ce schéma dans un langage du Web Sémantique ;
3. un mécanisme de traduction, d'une part des requêtes sur le schéma virtuel, et d'autre part des résultats provenant de la base.

Nous proposons un schéma générique décrit dans le langage OWL, qui une fois utilisé pour représenter une base de données (source) servira d'ontologie locale pour cette source. Nous allons présenter tout d'abord le schéma générique que nous proposons, avant de présenter le processus de transformation.

4.7.1 Le schéma virtuel VirtualSource.owl

Nous avons défini un schéma générique de représentation d'une base de données relationnelle, appelé **VirtualSource.owl**[40]. Ce schéma définit les classes et les propriétés nécessaires pour la représentation abstraite d'une base de données (voir Annexe B) : VirtualView pour les vues de la base, Attribute pour représenter les attributs de la base, etc. L'une des particularités de VirtualSource.owl est la représentation de la base de données, construite non pas en considérant ses tables relationnelles, mais les vues définies sur la base. La figure 4.3 présente une vue générale de VirtualSource.owl.

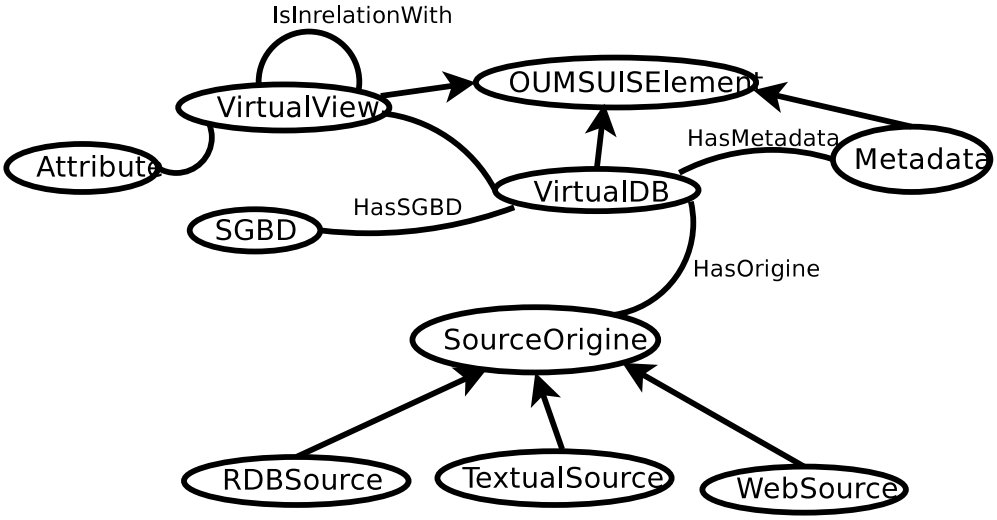


Fig. 4.3 – Schéma simplifié de VirtualSource.owl

Le listing OWL complet de VirtualSource.owl est donné en annexe de ce mémoire. Nous donnons ici un bref aperçu de quelques éléments.

Exemples de classes définies dans VirtualSource.owl

OUMSUISElement représente tout élément d'une base de données relationnelle[41]. Les autres classes héritent de cette classe à travers la relation Is-A.

[40] http ://www-timc.imag.fr/Gayo.Diallo/Resources/Ontologies/VirtualSource/

[41] OUMSUIS : Ontology-based Unified Management of Structured and Unstructured Sources, est le nom désignant l'approche de gestion unifiée

VirtualDB est la classe abstraite de représentation d'une base de données.

VirtualView représente une vue définie au sein de la base de données. Les vues virtuelles peuvent être liées aux classes de l'ontologie globale.

Attribute représente la classe abstraite de tout attribut de la base de données ou plus précisement les attributs de toute vue à exporter.

SGBD représente le système de gestion de la base de données. Un SGBD a une version et un driver.

Metadata représente la classe abstraite des métadonnées de la base. Elle sert à décrire les informations sur l'accès à la base (nom de la base, login, mot de passe).

TextualSource, RDBSource, WebSource représentent respectivement les sources d'origine de la base. Le résultat de la représentation sémantique de documents par exemple est une base de données textuelle. Les différentes sources d'origine sont des sous-classes de la classe abstraite **SourceOrigine**, qui représente l'origine de la base virtuelle décrite en OWL.

4.7.2 Exemple d'un attribut défini dans VirtualSource.owl

Nous avons défini un certain nombre d'attributs pour les classes présentées ci-dessus.

HasURL est l'attribut qui permet de spécifier l'URL de la base de données à représenter. Nous donnons sa définition OWL ci-dessous.

Listing 4.1 – Définition de l'attribut HasURL

```
<owl:DatatypeProperty rdf:about="#HasURL">
    <rdfs:range
        rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdfs:subPropertyOf>
      <owl:DatatypeProperty rdf:ID="Has"/>
    </rdfs:subPropertyOf>
    <rdfs:domain rdf:resource="#MetaData"/>
  </owl:DatatypeProperty>
```

4.7.3 Exemples de relations définies dans VirtualSource.owl

HasAttribut est une relation (owl :ObjectProperty) permettant de lier une *VirtualView* à ses attributs. Une VirtualView a au moins un attribut, cette contrainte étant spécifiée dans le schéma à l'aide de la restriction de cardinalité *owl :minCardinality*. Sa définition OWL est indiquée ci-dessous.

Listing 4.2 – Définition de la relation HasAttribut

```
<owl:ObjectProperty rdf:about="#HasAttribut">
    <rdfs:domain rdf:resource="#VirtualView"/>
    <rdfs:subPropertyOf rdf:resource="#Compose_De"/>
```

```
    <rdfs:comment
        rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
        Propriété spécifiant l'attribut d'une vue virtuelle
    </rdfs:comment>
  </owl:ObjectProperty>
```

Pour obtenir une représentation virtuelle d'une base relationnelle, nous procédons en deux étapes. Une première étape consiste à utiliser des techniques de réingénierie de bases de données afin de récupérer les éléments de la base nécessaires pour le besoin d'intégration. Nous appelons cette étape **RetroBD**. La seconde étape consiste à utiliser le résultat obtenu à l'étape **RetroBD** pour la description OWL de la base à l'aide du schéma générique. Nous appelerons cette étape **VirtualSource**.

Nous illustrons ces deux étapes à travers un exemple de représentation partielle d'une base de données de patient dont le schéma est présenté par la table 4.1. On suppose que pour chaque table une vue est définie.

TAB. 4.1 – Exemple d'un schéma de base de données de gestion de patients.

ExamPatient(IDExamPat,IDCAC,ExamDate,ExamCode,IDMed,noteMede)
Examen (codeExam,LibExam,typeExam)
LesionPatient (IDCAC,IntituleZoneAnat)
Medecin (IDmedecin,NomMed,PnomMed,ADRMed,VilleMed,TelMed,SpecialitMed)
Patient (IDCAC,NumDossier,PnomPatient,Sexe,AnneeNais,lateralite,VillePatient)

La base gère les informations concernant les patients, les médecins, les examens, etc. La table *LesionPatient* décrit les zones anatomiques du cerveau d'un patient touchées suite à un traumatisme cérébral. La base a autant de vues définies que des tables relationnelles.

4.7.4 L'étape RetroBD

Première étape du processus de représentation d'une base de données relationnelle en OWL, son objectif principal est l'acquisition des métadonnées de la base de données à représenter. Nous utilisons le système ISIS [Simonet et Simonet, 1999] pour cette étape. ISIS (Information System Initial Specification) est un environnement développé au sein de notre équipe, dédié aux bases de données pour la conception, la rétro-conception et la réingénierie de bases de données. Il permet notamment l'acquisition des métadonnées d'une base de données (tables relationnelles, vues, attributs, etc.).

Après avoir fourni les informations de connexion à la base de données (driver, url de la base, etc.) le système produit les métadonnées de la base. La figure 4.4 présente le processus de réingénierie. Les informations produites par cette étape sont sauvegardées dans un document XML qui est exploité par la seconde étape.

4.7.5 L'étape VirtualSource

Cette étape consiste en la représentation proprement dite de la base en OWL en utilisant les métadonnées fournies par l'étape RetroDB et le schéma générique VirtualSource.owl défini pour représenter virtuellement une base de données de type relationnel. Pour la représentation dans le schéma OWL, on applique les règles de transformation suivantes :

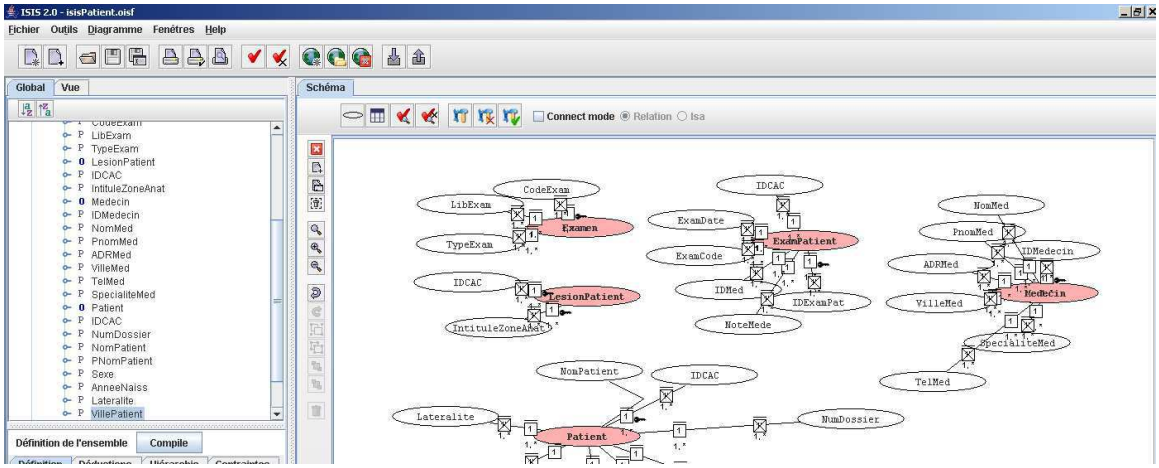


FIG. 4.4 – Réingénierie de la base de données Patient

1. chaque vue de la base devient une sous-classe de la classe VirtualView ;
2. chaque attribut de la base devient une sous-classe de la classe Attribute ;
3. les informations de description de la base de données (SGBD, login, url, ...) sont récupérées afin de renseigner les éléments correspondants du schéma générique.

Le listing 4.3 donne un extrait de la représentation OWL de la vue Patient de base de données patients du tableau 4.1.

Listing 4.3 – Définition partielle de la vue Patient dans le schéma VirtualSource.owl

```
<owl:Class rdf:ID="Patient">
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty>
          <owl:ObjectProperty rdf:ID="HasAttribut"/>
        </owl:onProperty>
        <owl:someValuesFrom rdf:resource="#NumDossier"/>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty>
          <owl:ObjectProperty rdf:about="#HasAttribut"/>
        </owl:onProperty>
        <owl:someValuesFrom rdf:resource="#IDCAC"/>
      </owl:Restriction>
    </rdfs:subClassOf>
...
    <rdfs:subClassOf>
      <owl:Class rdf:about="#VirtualView"/>
    </rdfs:subClassOf>
</owl:Class>
```

Pour la définition des différents attributs de la classe OWL "Patient", des restrictions OWL lui sont appliquées. Ainsi pour l'attribut "NumDossier" par exemple, on applique une restriction sur la propriété (owl :ObjectProperty) "HasAttribut". Par ailleurs "Patient" est une sous-classe de la classe "VirtualView".

4.8 Conclusion

Nous avons présenté de façon générale dans ce chapitre notre approche de gestion unifiée de données structurées et textuelles ainsi que l'approche de construction semi-automatique d'ontologies locales à partir de bases de données relationnelles. Nous avons proposé un schéma décrit en OWL permettant de représenter de manière générique une base relationnelle dans le cadre de l'intégration. L'implémentation du traducteur pour les requêtes est un travail en cours.

Nous avons décrit plusieurs types d'ontologies utilisées dans le cadre du processus d'intégration. Notons toutefois que des différences subsistent avec les systèmes tel que OBSERVER [Mena *et al.*, 2000], qui préconise aussi l'utilisation de plusieurs ontologies. Nous traitons explicitement les sources textuelles (en tenant compte de leur contenu), ce qui n'est pas le cas des autres systèmes. Une distinction est aussi faite entre les ontologies de représentation des sources et celles utilisées pour la caractérisation sémantique.

Les chapitres suivants seront consacrés au détail des composantes de l'architecture concernant a) la représentation hybride de sources textuelles, b) les ontologies sur le cerveau pour la caractérisation sémantique de sources, et c) la gestion de plusieurs ontologies et le support d'interrogation.

Chapitre 5

Ontologies pour l'organisation des connaissances sur le cerveau

5.1 Le contexte : le projet BC^3

L'objectif du projet BC^3 (Bases de Connaissances Coeur-Cerveau), qui s'est déroulé de 2000-2003, était l'extraction et la représentation des connaissances concernant deux organes vitaux, le cerveau et le coeur, dans les domaines anatomique et fonctionnel. Le projet engageait plusieurs équipes de la région Rhône-Alpes[42] et le Centre de Neuropsychologie de l'hôpital de la Pitié-Salpêtrière (Paris).

Pour le cas spécifique du cerveau, un des enjeux était la compréhension des mécanismes cérébraux chez l'homme afin d'améliorer la prise en charge des patients cérébro-lésés. Cette compréhension passe notamment par la réalisation d'une base de connaissances qui rend disponibles les connaissances de nature anatomique et fonctionnelle sur le cerveau, ainsi que les liens anatomo-fonctionnels entre les zones anatomiques du cerveau et les fonctions cognitives.

5.1.1 Les données disponibles dans le cadre du projet

Une description plus complète des données peut être trouvée dans le mémoire de thèse de B. Batrancourt [Batrancourt, 2003] et dans le rapport final du projet BC3. Nous en faisons un rapide survol.

A l'hôpital de la Pitié-Salpêtrière, le Centre de Neuropsychologie est spécialisé dans le bilan des séquelles neuropsychologiques des lésions cérébrales. La reconstruction du siège anatomique précis de la lésion responsable est maintenant possible à partir d'images acquises en IRM-3D. Un Centre d'Anatomie Cognitive (C.A.C.) a été constitué afin de réaliser le travail de recueil et de traitement des données médicales neuroanatomiques, neuropsychologiques et psychocomportementales auprès des patients cérébro-lésés. Le C.A.C. permet de mieux interpréter les désordres produits par ces lésions, d'en apprécier le pronostic et de définir la stratégie optimale en termes

[42] Ce projet a été soutenu par la région Rhône-Alpes. Il regroupait les laboratoires suivants :
CREATIS (Centre de Recherche et d'Applications en Traitement de l'Image et du Signal, Université Claude Bernard Lyon 1, Hôpital Cardiologique L.Pradel Service de Radiologie)
ERIC (Equipe de Recherche en Ingénierie des Connaissances, Université Lumière Lyon 2)
ISC (Institut des Sciences Cognitives, Université Claude Bernard Lyon 1)
LASS (Laboratoire d'Analyse des Systèmes de Santé, Université Claude Bernard Lyon 1)
TIMC (Techniques en Imagerie Modélisation Cognition, Faculté de Médecine, Université Joseph Fourier Grenoble 1)

de rééducation fonctionnelle. C'est aussi le point de départ pour l'acquisition des données natives nécessaires au projet de synthèse anatomo-fonctionnelle.

Les données sont recueillies auprès de patients présentant des séquelles de lésions cérébrales focales. La population principale est constituée de patients souffrant d'accident vasculaire cérébral. Ces patients sont adressés au Centre de Neuropsychologie de l'Hôpital Pitié-Salpêtrière afin d'être évalués sur le plan cognitif par l'établissement d'un bilan neuropsychologique ; certains de ces patients bénéficient en plus d'un entretien d'évaluation comportementale renseigné par eux-mêmes et un membre de leur famille. Les patients sont ensuite adressés au Centre de Neuroradiologie afin de réaliser un examen d'imagerie cérébrale permettant de préciser le siège exact de la lésion. Il s'agit d'un examen IRM de type T1, acquis dans les trois plans de l'espace. Lorsque l'ensemble des données est recueilli et exploité par les cliniciens, un staff C.A.C. (Centre d'Anatomie Cognitive) se réunit afin d'étudier le dossier complet du patient. L'objectif de l'étude BC^3 était de réaliser une synthèse anatomo-fonctionnelle des déficits cognitifs attribués aux séquelles de la lésion cérébrale.

B. Batrancourt [Batrancourt, 2003], dans le cadre de son travail de thèse, a formalisé dans une base de données anatomo-fonctionnelles (que nous appelerons BD CAC) les données recueillies selon le processus ci-dessus. En outre, les fichiers numériques générés par cette étude sont stockés sur un serveur de fichiers et une station de traitement d'images. La base de données initiale contenait des données recueillies entre Janvier 2000 et juin 2003. Ces données représentaient 75 dossiers patients (dossiers C.A.C). La base contenait en coutre des connaissances d'experts sur le cerveau. Ces connaissances ont été recueillies suite à de nombreuses réunions, tant avec des neuropsychologues pour l'aspect fonctionnel du cerveau qu'avec des neuroanatomistes pour l'aspect anatomique.

Notre tâche était de formaliser la connaissance sur le cerveau au travers d'ontologies, en vue de favoriser le partage et la réutilisation, et de faciliter la recherche d'information sur les comptes rendus médicaux et la littérature scientifique sur le cerveau. Notre base de travail a été une version réduite de la BD CAC, contenant des données anonymisées, et 45 fichiers de bilans neuropsychologiques (les comptes-rendus médicaux), anonymisés pour des raisons de confidentialité.

Nous décrivons la démarche que nous avons adoptée pour la construction d'ontologies à partir de ressources existantes dans le domaine du cerveau [Diallo *et al.*, 2003][Simonet *et al.*, 2003]. Nous commençons par décrire les ressources que nous avons utilisées dans ce travail.

5.1.2 Description sommaire de la BD CAC

Les dossiers de patients contenus dans la BD CAC sont structurés autour d'un certain nombre de types de données :

1. des données de nature générale concernant le patient, telles que la date de naissance, le sexe, la latéralité, l'étiologie de la lésion cérébrale ;
2. des données neuropsychologiques, comportementales et neuroanatomiques ;
3. les éléments sur le suivi et la rééducation entreprise par le patient ;
4. des références de la littérature en rapport avec le cas clinique.

La base répertorie :

1. les fonctions cognitives à évaluer, qui sont structurées en hiérarchie et subdivisées en grandes et sous-fonctions cognitives, pour établir les bilans neuropsychologiques ;
2. les différentes zones anatomiques du cerveau selon la hiérarchie de la Neuronames Brain Hierarchy (NBH) [Bowden et Dubach, 2005] ;

3. les tests cognitifs utilisés par les psychologues pour évaluer les fonctions cognitives.

A chacune des fonctions correspond une liste de tests ainsi que la performance du sujet témoin.

5.1.3 Les bilans neuropsychologiques

Un bilan neuropsychologique répertorie les résultats de l'évaluation des fonctions cognitives choisies pour un patient selon les symptômes qu'il présente. Le bilan pour un patient prend la forme d'une liste de tests auxquels sont associées les performances obtenues par le patient. Les conclusions de l'analyse de chaque bilan sont consignées à la suite de la liste des fonctions cognitives évaluées. L'analyse d'un bilan comporte une dimension quantitative (méthodes statistiques) et qualitative (interprétation faite par un psychologue des processus cognitifs et des comportements, sur la base des performances aux épreuves). Notons qu'un bilan neuropsychologique peut aussi évaluer les fonctions cognitives majeures au travers de leurs sous-systèmes. A titre d'exemple, la *mémoire* peut s'évaluer au travers de la *mémoire épisodique*, la *mémoire à court terme*, la *mémoire procédurale*, etc. La mémoire sémantique peut être évaluée selon ses modalités visuelles, verbales, etc. Le *langage* peut être évalué au travers de la *compréhension* et de l'*expression*.

Comme nous l'avons indiqué au chapitre 1, le cerveau peut être vu selon ses aspects anatomique et fonctionnel. Chaque aspect donne une perspective particulière sur le cerveau. Nous avons été amenés à expliciter une ontologie anatomique et une ontologie fonctionnelle du cerveau. Nous avons aussi développé une ontologie qui sert à l'identification des valeurs de tests cognitifs dans des bilans neuropsychologiques. Cette dernière ontologie peut être vue comme une sous partie de l'ontologie des fonctions cognitives (incluant les tests cognitifs), à laquelle nous avons ajouté des entités et des relations spécifiques relatives aux tests cognitifs.

5.2 Démarche de construction d'ontologies à partir de ressources existantes

Nous avons rappelé dans le chapitre consacré aux ontologies qu'il n'existe pas aujourd'hui de méthode standard pour la construction d'ontologies. Dans le cadre de la construction d'ontologies sur le cerveau, nous avons adapté les approches existantes décrites dans la partie état de l'art.

L'approche de construction d'ontologies que nous avons adoptée inclut les phases suivantes :

1. la spécification de l'ontologie, incluant l'identification des besoins et le balisage du domaine ;
2. une phase d'étude de l'existant, qui inclut l'identification des ressources existantes, utilisables dans la construction de la nouvelle ontologie ;
3. la conceptualisation, incluant la mise en place de la structure conceptuelle et la définition des entités identifiées ;
4. l'enrichissement conceptuel et terminologique, prenant en compte l'incorporation d'autres ressources recensées lors de la phase d'étude de l'existant ;
5. la formalisation et l'implémentation de l'ontologie, incluant la traduction dans un formalisme de représentation des connaissances de la structure conceptuelle enrichie et le codage dans un langage de représentation d'ontologies ;
6. le test de la cohérence interne, au moyen d'un moteur d'inférences, et la validation de l'ontologie.

Naturellement, une fois construite, l'ontologie est appelée à évoluer pour la prise en compte de nouveaux besoins et l'actualisation de la connaissance (en particulier dans le domaine médical). Cette phase, postérieure à la construction, entre dans le cadre de la maintenance de l'ontologie.

5.2.1 Phase de spécification : recensement des besoins et balisage du domaine

Fernandez [Fernandez *et al.*, 1997] préconise de ne pas commencer le développement d'une ontologie sans savoir quels seront ses buts et sa portée. Il est donc important de savoir pourquoi l'ontologie va être créée et quels vont être ses utilisateurs. Selon Grüninger et Fox [Grüninger et Fox, 1995] le développement d'une ontologie est motivé par les scénarios qui apparaissent dans l'application. Ces scénarios sont des problèmes ou exemples qui ne sont pas traités par les ontologies existantes.

Dans notre contexte, les principales motivations pour l'explicitation de nouvelles ontologies sont :

1. séparer les données et les connaissances du domaine gérées par la base de données CAC, en vue de la réutilisation des connaissances dans d'autres contextes ;
2. favoriser l'indexation conceptuelle automatique et la recherche de documents textuels sur le cerveau (e.g., comptes rendus médicaux) ;
3. faciliter la compréhension du cerveau (objectif pédagogique).

Nous avons fait le choix d'expliciter la connaissance sur le cerveau à travers une ontologie anatomique du cerveau (pour le volet anatomique du cerveau) et une ontologie fonctionnelle du cerveau (pour les aspects liés aux fonctions et aux tests cognitifs). Nous avons identifié une troisième ontologie, utile dans le cadre de la caractérisation des bilans neuropsychologiques, qui servira de support à l'identification des performances réalisées par les patients (valeurs de tests cognitifs). Les ontologies doivent fournir une terminologie en français et en anglais afin de pouvoir traiter des ressources multilingues.

5.2.2 Phase d'identification des ressources disponibles

Par rapport à l'objectif fixé pour les ontologies à construire, nous avons exploré un certain nombre de ressources d'intérêt pour notre tâche. Notre base de départ est constituée des ressources disponibles dans le cadre du projet BC^3, que nous avons brièvement présentées ci-dessus. Nous avons étudié d'autres ontologies et vocabulaires contrôlés dans le domaine du cerveau, que nous pouvions réutiliser. Nous avons en particulier exploré la NBH, la FMA, UMLS, SNOMED-CT et le thésaurus MeSH (anglais et français) (voir le tableau 5.2.2).

L'ontologie du cortex anatomique du cerveau [Dameron *et al.*, 2003] est destinée à une utilisation dans le domaine de l'imagerie neuroanatomique. Elle inclut des relations de nature topologique et elle n'était pas adaptée pour nous pour la caractérisation automatique de documents. Nous avons éliminé de notre champ d'étude certaines ontologies biomédicales telles que la GO (Gene Ontology), dont le but est de fournir un vocabulaire contrôlé pour décrire les gènes des organimes, ce qui n'est pas notre objectif présent.

Nous avons ajouté à la liste de ressources explorées le découpage cytoarchitectonique de Brodmann, connu sous le nom des aires de Brodmann [Brodmann, 1909] et la Classification Internationale du Fonctionnemement connue sous le nom de CIF[43], établie par l'Organisation Mondiale de la Santé[44]. La CIF est une classification sur la santé et les domaines connexes qui décrit les fonctions et les structures du corps humain.

[43] http ://www.who.int/classifications/icf/en/ (consulté le 2 Novembre 2006)
[44] www.who.org (consulté le 2 Novembre 2006)

La CIF

La CIF couvre tous les aspects de la santé humaine et certaines composantes du bien-être qui relèvent de la santé. Elle les décrit en termes de **domaine de la santé** et **domaines connexes de la santé**. Parmi les domaine de la santé on peut citer la vision, l'audition, la marche, l'apprentissage, la mémoire, tandis que parmi les domaines connexes de la santé, on trouve la mobilité, l'éducation, etc.

La CIF organise l'information en deux parties. La partie 1 traite du fonctionnement et du handicap, alors que la partie 2 couvre les facteurs contextuels. La figure 5.1 donne un aperçu de la structuration de la CIF. Dans la CIF, notre intérêt porte principalement sur la composante **fonctions organiques**. Les fonctions organiques désignent les fonctions physiologiques des systèmes organiques (y compris les fonctions psychologiques) incluant les sens de base comme la vue (fonction de la vision). La CIF, disponible dans plusieurs langues (dont le français et l'anglais) fournit une définition de toutes les fonctions classées.

<table>
<tr><th></th><th colspan="2">Partie 1
Fonctionnement et handicap</th><th colspan="2">Partie 2
Facteurs contextuels</th></tr>
<tr><th>Composantes</th><th>Fonctions organiques et structures anatomiques</th><th>Activités et participation</th><th>Facteurs environnementaux</th><th>Facteurs personnels</th></tr>
<tr><td>Domaines</td><td>Fonctions organiques

Structures anatomiques</td><td>Domaines de la vie (tâches, actions)</td><td>Facteurs externes affectant le fonctionnement et le handicap</td><td>Facteurs internes affectant le fonctionnement et le handicap</td></tr>
<tr><td>Schémas</td><td>Changement dans les fonctions organiques (physiologie)

Changement dans la structure anatomique</td><td>Capacité
réaliser des tâches dans un environnement standard
Performance
réaliser des tâches dans l'environnement réel</td><td>Impact (facilitateur ou obstacle) de la réalité physique, de la réalité sociale ou des attitudes</td><td>Impact des attributs de la personne</td></tr>
<tr><td rowspan="2">Aspect positif</td><td>Intégrité fonctionnelle et structurelle</td><td>Activité
Participation</td><td rowspan="2">Facilitateurs</td><td rowspan="2">Sans objet</td></tr>
<tr><td colspan="2">Fonctionnement</td></tr>
<tr><td rowspan="2">Aspect négatif</td><td>Déficience</td><td>Limitation de l'activité
Restriction de la participation</td><td rowspan="2">Barrières/ obstacles</td><td rowspan="2">Sans objet</td></tr>
<tr><td colspan="2">Handicap</td></tr>
</table>

FIG. 5.1 – Aperçu de la CIF

Les aires de Brodmann

Les travaux du Dr. Korbinian BRODMANN [Brodmann, 1909] sur des cerveaux de macaques et d'humains ont permis un découpage anatomique du cerveau en zones couramment appelées

aires de Brodmann (voir figure 5.2). Chaque région du cortex ayant la même organisation cellulaire a reçu un numéro allant de 1 à 52. Un des intérêts de la classification de Brodmann est qu'elle fournit l'un des premiers modèles des relations entre structures et fonctions cérébrales chez l'homme. L'intuition de Brodmann, qui s'est vue fréquemment confirmée par la suite, était qu'à une organisation anatomique donnée correspond une fonction particulière. Brodmann a établi notamment la correspondance entre certaine fonctions cognitives majeures telles que la vision ou le langage, et les zones anatomiques du cerveau sollicitées pour accomplir ces fonctions. Par exemple, l'aire 17 de Brodmann, qui reçoit de l'information d'un noyau du thalamus relié à la rétine, s'est avérée correspondre exactement au cortex visuel primaire. Cependant ces correspondances sont peu ou pas connues quand il s'agit des niveaux inférieurs de la hiérarchie anatomique, et la parcellisation de Brodmann a été fortement remise en question ces dernières années avec l'émergence de nouvelles méthodes d'imagerie neurofonctionnelle [Batrancourt, 2003].

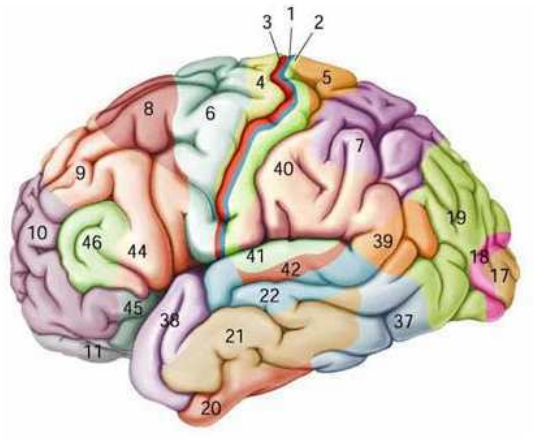


FIG. 5.2 – Découpage cytoarchitectonique de Brodmann

Le tableau 5.2.2 récapitule les différentes ressources explorées. Nous classons les ressources dans deux catégories : les ressources qui permettent un enrichissement conceptuel des futures ontologies (e.g., la CIF, la NBH) et les ressources qui permettent d'enrichir la terminologie utilisée pour désigner les concepts (e.g., UMLS). La FMA structure les organes du corps humain, tant à l'échelle macroscopique que microscopique. Sa composante neuroanatomique combine la NBH et la Terminologia Anatomia (standard international de la terminologie anatomique humaine). Nous avons donc décidé de considérer la NBH comme base pour la partie anatomique du cerveau, librement accessible et incluse en outre dans UMLS, ce qui nous permettra d'enrichir la terminologie de l'ontologie anatomique.

Notons que ces ressources existent dans des formats différents : la NBH est disponible sous la forme d'un fichier Excel, la classification des tests cognitifs (200) disponible au CAC est consignée dans la base CAC réalisée en Access, etc.

5.2.3 Phase de conceptualisation

C'est dans cette phase que nous définissons la structure conceptuelle pour chacune des ontologies à construire. Partant des ressources disponibles et des différents termes qu'elles contiennent, nous identifions une structure conceptuelle de base pour les ontologies, modélisée à travers un diagramme UML.

TAB. 5.1 – Liste des ressources explorées dans le cadre de la construction d'ontologies sur le cerveau

Nom de la ressource	**Préfixe**	**Description**	**Lien**
Base de Données CAC et bilans neuropsychologiques	BDCAC	données neuroanatomiques et neuropsychologiques	-
Neuronames Brain Hierarchy	NBH	nomenclature du cerveau humain et des macaques	http://braininfo.rprc.washington.edu/tools.html
Foundational Model of Anatomy	FMA	modélisation de la structure du corps humain	http://sig.biostr.washington.edu/projects/fm/AboutFM.html
Classification internationale du fonctionnement, de la santé et des handicaps	CIF	langage formalisé et uniformisé sur la santé	http://www3.who.int/icf
Unified Medical Language System	UMLS	vocabulaire généraliste dans le domaine biomédical	http://www.nlm.nih.gov/research/umls/
SNOMED Clinical Terms	SNOMED-CT	terminologie clinique	http://www.snomed.org/snomedct/glossary.html
Medical Subject Heading	MeSH	thésaurus de la base Medline	http://www.nlm.nih.gov/mesh/
Medical Subject Heading Français	MeSHFRE	version française (INSERM) du thésaurus de la base Medline	http://ist.inserm.fr/basismesh/mesh.html
Aires de Brodmann	Brodmann	Découpage cytoarchitectonique de Brodmann	-

L'ontologie des fonctions cognitives

Le diagramme de la figure 5.3 présente un extrait du diagramme UML de l'ontologie des fonctions cognitives du cerveau. *AspectCognitifCerveau* est la racine de l'ontologie. Une fonction

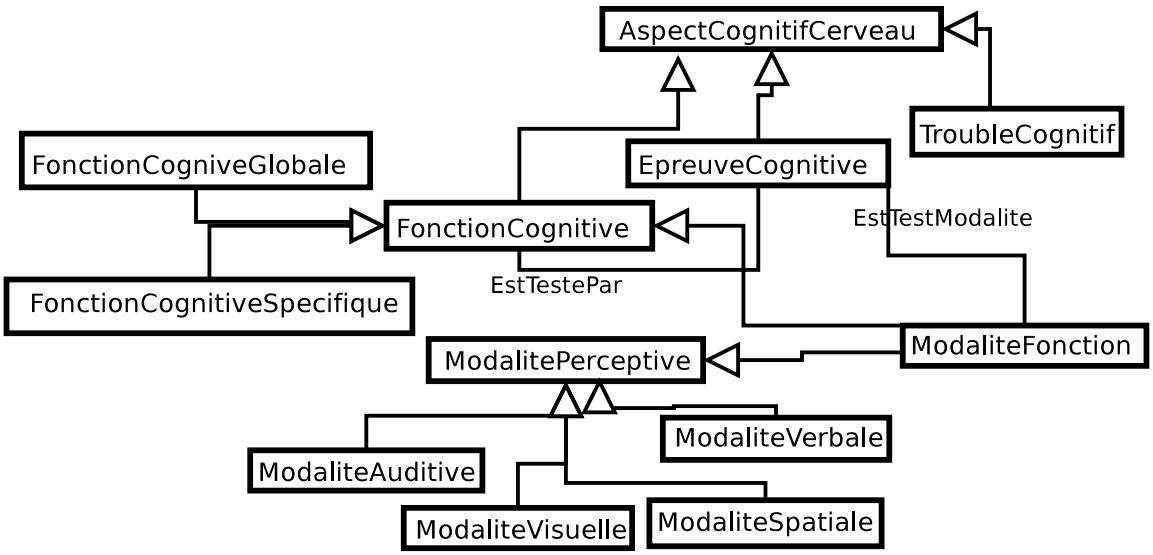


FIG. 5.3 – Diagramme UML partielle pour l'ontologie des fonctions cognitives

cognitive *FonctionCognitive* est évaluée par un certain nombre d'épreuves cognitives ou tests cognitifs (*EpreuveCognitive*). Une épreuve cognitive évalue parfois une fonction cognitive selon une modalité donnée (l'association *EstTestModalite* du diagramme). *ModaliteFonction* représente la modalité d'une fonction donnée (par exemple la modalité verbale de la mémoire). L'association *EstTestModalite* est spécialisée selon les différentes modalités en *EstTestModaliteVerbale*, *EstTestModaliteVisuelle*, etc. Par exemple, l'un des tests de la modalité verbale de la mémoire sémantique est le *Palm Tree Test*.

L'ontologie anatomique

La figure 5.4 présente un extrait du diagramme UML de l'ontologie anatomique du cerveau.

Toutes les relations identifiées ne sont pas représentées. *StructureAnatomiqueCerveau* et *LocalisationAnatomique* spécialisent le concept *StructureAnatomique* représentant tout organe, partie d'un organe, une région ou frontière de régions du corps humain. Une structure anatomique a un type sémantique (concept *TypeSemantique*) qui représente les type sémantiques définis dans UMLS.

L'ontologie pour le support d'identification des performances aux tests

Cette ontologie explicite plus en détail les tests cognitifs en décrivant le type de variable qu'on mesure pour évaluer les performances (note brute, pourcentage), la valeur maximale qu'on peut obtenir au test, etc. La figure 5.5 donne l'extrait correspondant du diagramme UML.

Par exemple la *Dénomination d'images DENO 100* est mesurée par une note brute dont la valeur maximale est 100. Un test cognitif peut avoir une origine identifiée (le ou les auteurs du test).

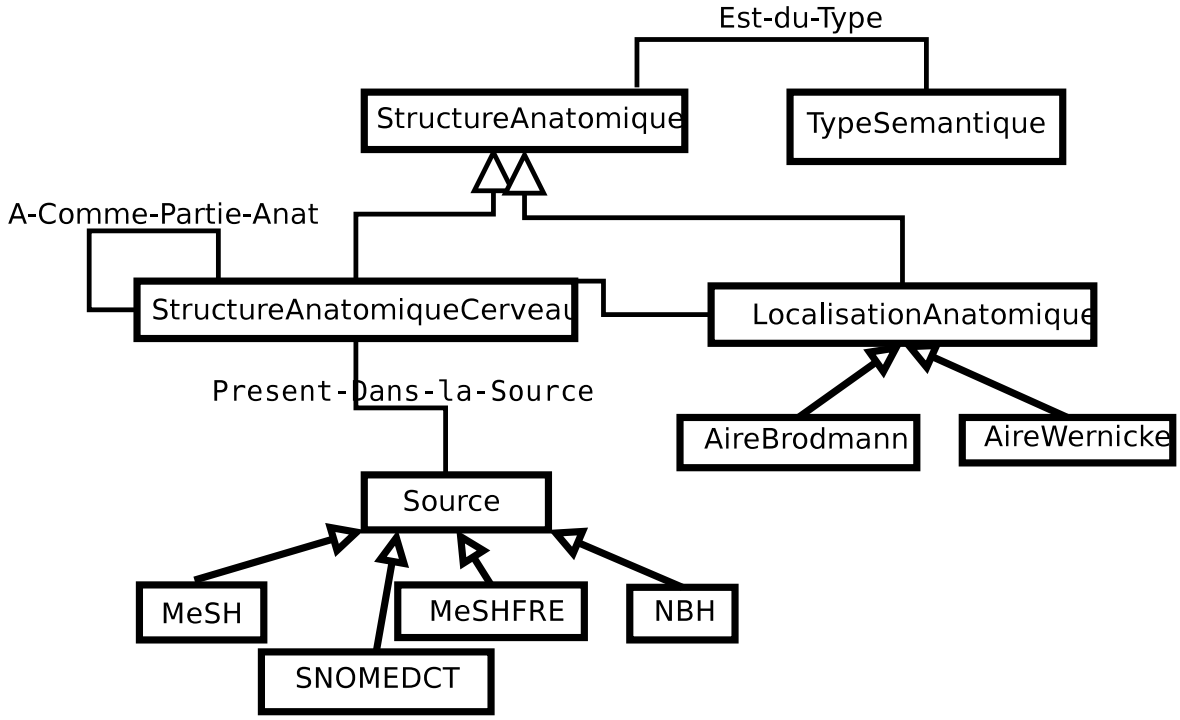


FIG. 5.4 – Extrait du Diagramme UML pour l'ontologie anatomique du cerveau

5.2.4 L'enrichissement conceptuel et terminologique

Une fois obtenue la structure conceptuelle de base de chaque ontologie, il s'agit de l'enrichir. Les concepts identifiés pendant la phase de spécification constituent les concepts majeurs de chaque ontologie. Ces concepts doivent être spécialisés et affinés et les termes qui les dénotent doivent être ajoutés. Un ontologie destinée à un usage de recherche d'information doit contenir un vocabulaire aussi large que possible afin de faciliter l'indexation conceptuelle automatique. On utilise pour l'enrichissement les ressources recensées à la phase d'identification des ressources. Cet enrichissement peut être complété par un enrichissement à partir de textes spécialisés dans le domaine couvert par l'ontologie [Simonet *et al.*, 2006]. Pour l'ontologie des fonctions cognitives nous avons utilisé la classification des experts, formalisée dans la BD CAC et la CIF. La disponibilité d'une version Française de la CIF nous a permis de faire le rapprochement manuel, basé sur le libellé de la fonction, avec les fonctions cognitives de la BD CAC. L'ontologie anatomique présente une particularité, puisque l'enrichissement de la structure conceptuelle s'est faite de façon automatique en utilisant les outils fournis par UMLS pour l'interroger à distance.

Enrichissement de l'ontologie anatomique

Nous avons utilisé comme base d'enrichissement la NBH qui fournit une structuration anatomique du cerveau selon une relation partonomique. Le cortex cérébral, par exemple, qui a comme identifiant 20 et qui est désigné dans la NBH par le terme "Cerebral Cortex", est une composante de la structure 12 de la NBH, désignée par le terme "Telencephalon". La NBH est disponible sous la forme d'un ensemble de tables : la table des structures hiérarchiques dont est extrait l'exemple du tableau ci-dessous), la table des structures auxilliaires (où on peut trouver les différentes aires de Brodmann) et la table des synonymes qui donne pour chaque structure

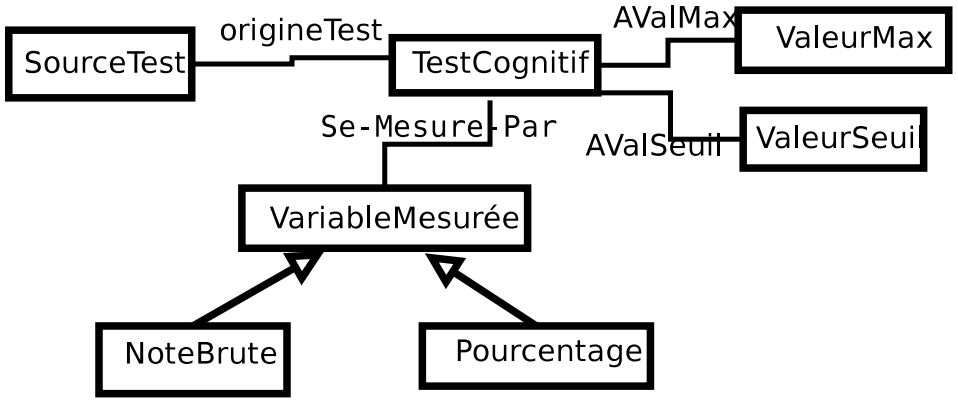


FIG. 5.5 – Diagramme UML pour l'ontologie pour le support d'identification des performances aux tests

TAB. 5.2 – Extrait de la Neuronames Brain Hierarchy

ID	**Label Anglais**	**Label Latin**	**Type**	**ID Parent**
1	Brain	Encephalon	Volumetrique	1
2	Cerebrocerebellar fissure	Fissura cerebrocerebellaris	Surfacique	1
8	Forebrain	Prosencephalon	Volumetrique	1
12	Telencephalon	Telenchephalon	Volumetrique	8
20	Cerebral Cortex	Cortex Cerebri	Volumetrique	12

les termes synonymes, qui lui sont associés et qui indique si la structure est identifiée dans le cerveau humain ou le cerveau des macaques.

A partir des tables fournies par la NBH, nous avons sélectionné et reconstitué uniquement les structures volumétriques (les "Volumetric structures" de la NBH) identifiées dans le cerveau humain (au nombre de 855) et avons exclu les autres structures (classées comme les "Superficial features"). Ces dernières sont des surfaces à 2 dimensions ou des structures à 3 dimensions qui sont adossées au cerveau mais n'en font pas parties. Un exemple de cette catégorie de structures sont les scissures (sillon naturel segmentant le cerveau).

Nous avons ensuite utilisé l'API fournie par UMLS pour son accès à distance[45] pour reconstituer la hiérarchie selon les identifiants de concepts UMLS (CUI). Ces identifiants UMLS permettent d'établir le lien avec les autres sources de vocabulaire comme la SNOMED-CT ou le MeSH français. Nous avons élaboré un algorithme implémenté en Java qui permet d'enrichir, à partir d'UMLS et de la structure hiérarchique obtenue à partir de la NBH, la structure conceptuelle de l'ontologie anatomique traduite en OWL.

L'automatisation de la tâche d'enrichissement conceptuel et terminologique de l'ontologie anatomique à partir de UMLS nous permet d'actualiser régulièrement l'ontologie en fonction des évolutions apportées au niveau du vocabulaire de la NBH et des autres sources connexes de UMLS. La version actuelle de l'ontologie anatomique est basée sur l'édition 2006AA d'UMLS.

[45] UMLSKS API disponible en téléchargement à l'adresse `http://umlsks.nlm.nih.gov/`

5.2.5 Formalisation et implémentation

Cette étape se confond parfois avec l'étape d'enrichissement. C'est le cas notamment de l'ontologie anatomique. Nous avons choisi le langage OWL comme langage de représentation de nos ontologies. Nous avons utilisé l'éditeur Protégé, qui permet de coder directement une ontologie en OWL. OWL s'est imposé naturellement, d'abord parce que c'est le langage recommandé par le W3C pour la représentation des ontologies, ensuite parce que de nombreux outils sont disponibles aujourd'hui dans le cadre du Web Sémantique pour le traitement d'ontologies OWL.

Le diagramme UML de chaque ontologie est alors traduit manuellement et édité sous Protégé.

Par exemple, la forme abrégée de la définition de la mémoire sémantique (*MemoireSemantique*) de l'ontologie des fonctions cognitives est :

$$\begin{aligned} SemanticMemory \sqsubseteq\ & ExplicitMemory \\ & \sqcap \exists TestModaliteVisuelle.(PictureNaming \sqcup ... \sqcup Deno80) \\ & \sqcap \exists TestModaliteVerbale.(TestVoca \sqcup ... \sqcup PalmTreeTest) \\ & \sqcap \ni PreflabelEn.(Semantic\ Memory) \\ & \sqcap \ni PreflabelFr.(Mémoire\ Sémantique) \end{aligned}$$

Cette définition indique que la mémoire sémantique est une sous-classe de la mémoire explicite (*ExplicitMemory*) et est évaluée pour son versant verbal par les tests du vocabulaire (*TestVoca*), du Palm Tree Test (*PalmTreeTest*), etc. Elle est évaluée selon la modalité visuelle par les tests du Picture Naming (PictureNaming), de Dénomination (*Deno80*), etc. La mémoire sémantique a comme terme préféré en Français (*PrefLabelFr*) *Mémoire Sémantique* et comme terme préféré en Anglais (*PrefLabelEn*) *Semantic Memory*. La figure 5.6 présente l'édition de l'ontologie des fonctions cognitives du cerveau.

La figure 5.7 donne une vue de l'ontologie anatomique sous Protégé. Nous avons mis en évidence la structure anatomique *AnatCerveau0004781* (dont la forme compacte est donnée ci-dessous), qui représente le noyau central du cerveau. Sa définition anglaise fournie par le MeSH est reprise par la propriété *rdfs :comment* de OWL. L'aspect multilingue est géré par l'utilisation de la propriété *rdfs :label* de OWL.

$$\begin{aligned} AnatCerveau0004781 \sqsubseteq\ & BrainAnatomicStructure \\ & \sqcap \forall Is-Anatomical-Part.AnatCerveau0039452 \\ & \sqcap \exists Has-Semantic-Type.BodyPartOrgan \\ & \sqcap \exists Present-To-Source.NBH \\ & \sqcap \exists Present-To-Source.MeSHFRE \\ & \sqcap \exists Present-To-Source.SNOMEDCT \\ & \sqcap \exists Present-To-Source.MeSH \\ & \sqcap \ni Has-UMLS-CUI.(C0004781) \end{aligned} \tag{5.1}$$

La structure *AnatCerveau0004781* a comme structure parente unique *AnatCerveau0039452*, qui représente le télencéphale. Elle est désignée par plusieurs termes en français et en anglais. Comme la propriété *Is-Anatomical-Part* a comme propriété inverse *Has-Anatomical-Part*, nous ne déclarons pas explicitement que *AnatCerveau0039452 Has-Anatomical-Part AnatCerveau0004781*, cette information pouvant être déduite.

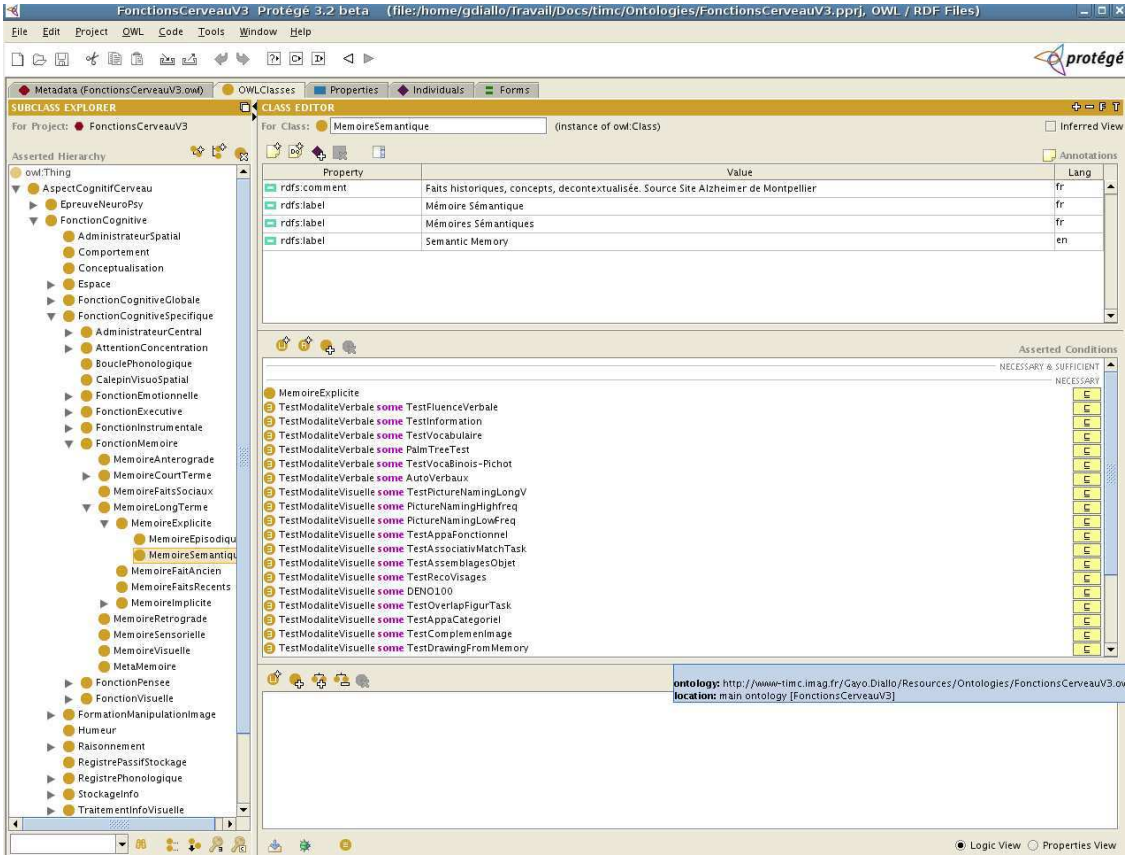


FIG. 5.6 – Edition de l'ontologie des fonctions cognitives sous Protégé

5.2.6 Tests de cohérence de l'ontologie et validation

La cohérence de la hiérarchie de classes des ontologies (la satisfiabilité des classes) a été vérifiée en utilisant le moteur d'inférence Racer 1.9 [Haarslev et Möller, 2001], qu'il est possible de configurer avec l'éditeur Protégé.

Le projet BC^3 a servi de cadre pour une première validation des ontologies anatomiques et fonctionnelles. Outre cette validation, la mise à disposition d'une version OWL de l'ontologie anatomique intéresse D. Bowden de l'Université de Washington (co-auteur de la NBH), ainsi que l'intégration à la NBH de la terminologie française.

Le tableau ci-dessous récapitule les statistiques concernant les ontologies que nous avons explicitées (voir Annexe A pour avoir une vue des hiérarchies).

5.3 Conclusion

Nous avons présenté dans ce chapitre la démarche que nous avons utilisée pour la construction des ontologies sur le cerveau. Compte tenu de la disponibilité de suffisamment de ressources réutilisables pour nos besoins, nous avons adopté une démarche ad hoc de construction. Nous disposons à présent d'ontologies réutilisables dans différents contextes. Ces ontologies nous permettent d'effectuer la caractérisation sémantique de ressources textuelles (articles scientifiques sur le cerveau, bilans neurospsychologiques, etc.). Elles ont aussi servi à tester notre approche de gestion conjointe de plusieurs ontologies (voir chapitre 7). En terme d'évolution de l'onto-

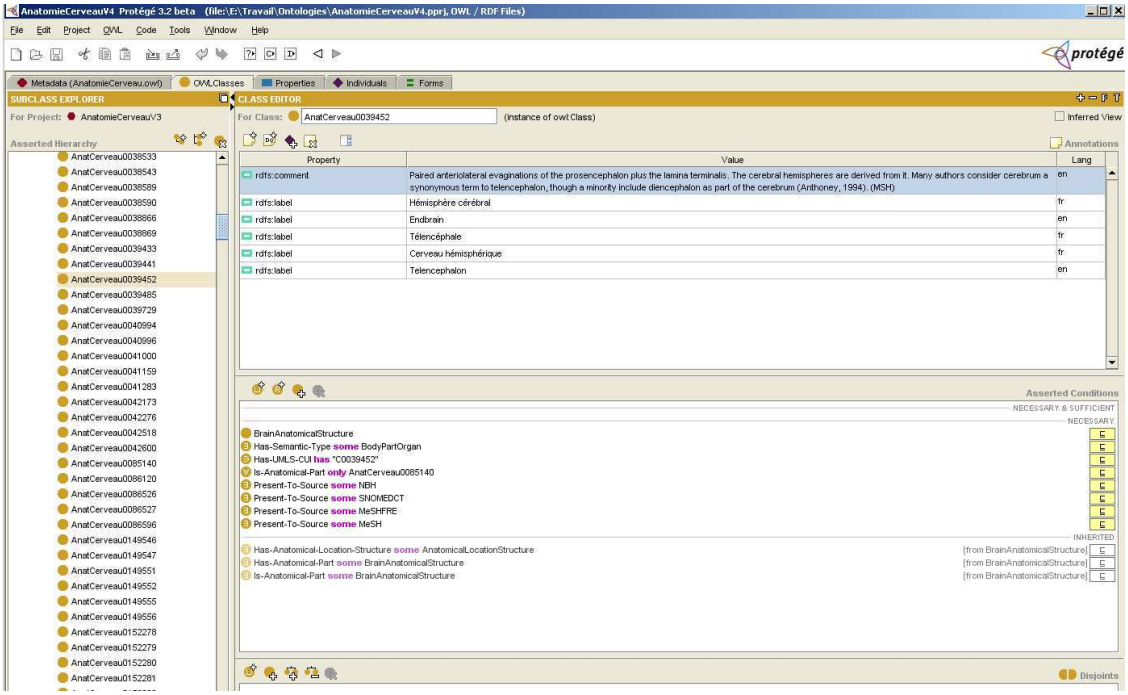


FIG. 5.7 – Ontologie anatomique du cerveau sous Protégé

TAB. 5.3 – Statistiques des ontologies sur le cerveau

Ontologie	**Nombre de classes OWL**	**Nombre de termes (labels)**
Ontologie Anatomique	988	4.880
Ontologie des Fonctions Cognitives	575	898
Ontologie pour les tests	321	407

logie cognitive, nous envisageons d'intégrer la sous-partie de la classification internationale des maladies (CIM-10[46]) consacrée aux maladies neuropsychologiques.

[46] http ://www.med.univ-rennes1.fr/noment/cim10/ (consulté le 5 Novembre 2006)

Chapitre 6

Représentation hybride de sources textuelles

6.1 Introduction

Nous abordons dans ce chapitre notre approche de représentation hybride de sources textuelles permettant la prise en compte explicite de textes (pouvant être multilingues) dans l'infrastructure d'intégration.

L'approche que nous proposons fait l'hypothèse qu'une même source textuelle peut être caractérisée par plusieurs ontologies, chacune donnant un point de vue particulier sur la dite source. Ainsi, une ontologie sur les gènes peut donner un point de vue génétique, une ontologie sur l'anatomie un point de vue anatomique, etc. Toutefois, nous considérons que les différentes ontologies à utiliser traitent de différents aspects tout en donnant des vues complémentaires sur la source. Si les ontologies se recouvrent, elles doivent au préalable être intégrée.

6.2 Définitions préliminaires

Mot un mot $M = s_1, s_2, ..., s_p$ est une suite ordonnée de symboles (caractères et/ou chiffres) autorisés dans une langue donnée et délimitée par un espace ou un signe de ponctuation.

Stem le stem d'un mot (au sens de la définition ci-dessus) ou racine lexicographique commune, est le mot obtenu après troncature des suffixes.

Terme un terme T est une séquence ordonnée finie $m_1, m_2...m_n$ de mots. Dans une ontologie, un terme peut être utilisé pour désigner (comme représentant préféré ou alternatif) un ou plusieurs concepts.

Concepts un concept C est le signifié d'un terme dont on décide de négliger la dimension linguistique.

Entités nommées est une expression générique pour désigner les noms de personnes, de localisation géographiques, les dates, etc.

6.3 Caractérisation de sources textuelles pour une perspective d'intégration

6.3.1 Description générale de l'approche

La caractérisation de sources textuelles est le processus qui permet d'obtenir une représentation exploitable des documents de la source dans la perspective d'une gestion unifiée de données structurées et non structurées.

L'approche que nous préconisons repose sur la construction d'une base suivant un schéma appelé Représentation Sémantique de Documents (RSD), qui permet de représenter toute source textuelle à traiter dans le cadre d'une intégration, selon trois dimensions :

1. une dimension basée sur le contenu intrinsèque des documents, c'est-à-dire les mots simples ;
2. une dimension conceptuelle qui utilise des ontologies et des méta-données comme sources externes de connaissances ;
3. une dimension qui représente les documents selon les entités nommées qu'ils contiennent.

La figure 6.1 détaille l'approche de construction du RSD :

La partie 1) est constituée de trois modules qui effectuent la représentation sémantique proprement dite. Elle comprend :

1. un module *HybrideIndex*, accomplissant la tâche de représentation combinant l'indexation par termes simples (mots) et l'indexation basée sur les concepts (issus d'ontologies) [Diallo *et al.*, 2006a]. Compte tenu en effet de la non complétude des ontologies utilisées en général dans la caractérisation sémantique de documents, nous avons préconisé de garder une représentation mixte (mots et concepts) ;
2. un module *Annotation Manuelle*, qui a à la fois le rôle de fournisseur de méta-données (information de catalogage) pour les documents, et le rôle de correction du résultat de l'indexation automatique basée sur les concepts. Notre approche d'identification de concepts dans les documents n'utilise pas de désambiguïsation automatique. En effet, les termes utilisés pour dénoter les concepts des ontologies sont très souvent des termes composés, sujets à moins d'ambiguïté que les termes simples. Il fallait donc faire le choix entre le coût d'utilisation de la désambiguïsation automatique et celui d'une désambiguïsation manuelle ;
3. un module *RENO* (Reconnaissance des Entités Nommées à l'aide d'une Ontologie), pour l'identification d'entités nommées (EN) dans les documents textuels en utilisant une ontologie. L'identification des EN permet d'effectuer des interrogations de nature précise dans les documents (e.g., retrouver tous les documents dont la valeur de l'entité Entité1 est égale à X). L'implémentation de ce module est motivée par le fait que dans le cas des comptes rendus médicaux (disponibles dans le contexte du projet BC^3), certaines informations essentielles concernant les patients, notamment les résultats des évaluations cognitives, sont consignées dans des fichiers textuels. Pouvoir repérer ces informations permet d'établir un lien avec des données sur les patients, structurées et gérées à travers des bases de données.

La partie 2) est constituée du module de gestion des différentes ontologies manipulées (OTOMANAGE). Ce module fournit les éléments nécessaires pour le chargement, la navigation, la gestion des concepts et des termes d'une ontologie. Il est construit en utilisant l'API Jena [Carroll *et al.*, 2004], développée aux laboratoires HP, qui permet de gérer des ontologies représentées en RDF ou OWL. Il utilise par ailleurs le moteur d'inférence RACER [Haarslev et Möller, 2001] pour la vérification de cohérence des ontologies et faire des inférences.

La partie 3) représente le résultat de la caractérisation sémantique gérée à travers une base de données relationnelle que nous décrivons à la section B.3. L'utilisation du modèle relationnel pour la représentation du RSD permet d'apporter le premier niveau d'homogénéisation avec les sources structurées gérées dans le cadre de la gestion unifiée. Les données peuvent être exportées facilement sous divers formats (XML, RDF, OWL) par l'ajout de modules spécifiques.

La partie 4) représente le module *Recherche Hybride* conçu pour effectuer des recherches mixtes sur le RSD, combinant des termes simples et des concepts d'ontologie.

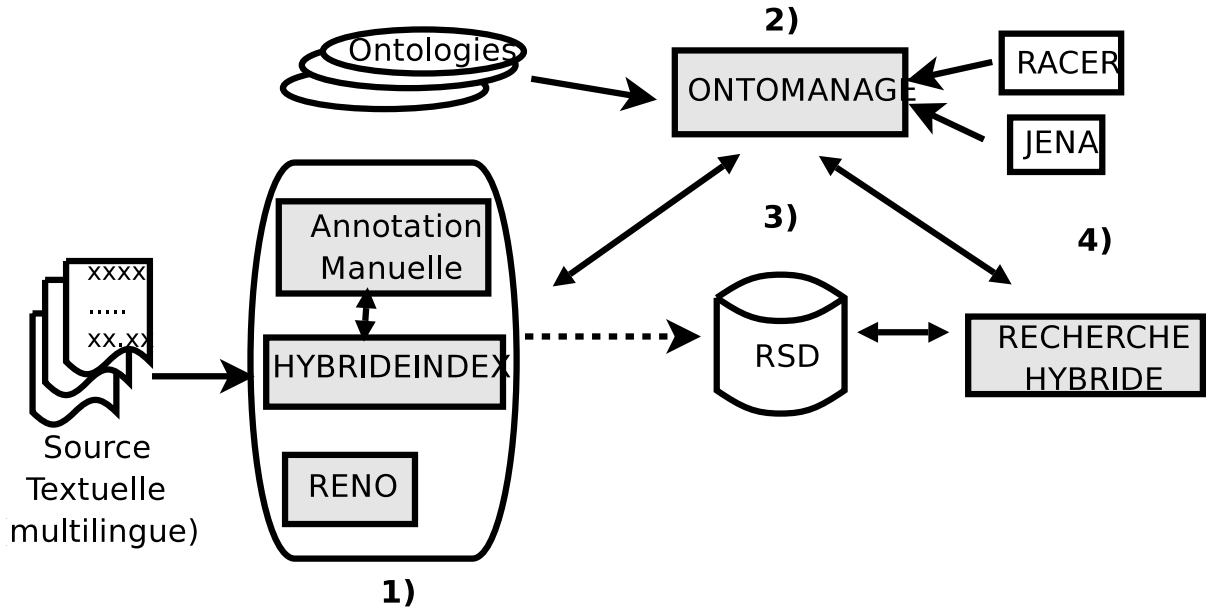


FIG. 6.1 – Architecture de représentation hybride de sources textuelles

Notons qu'avant d'entrer dans le processus décrit ci-dessous, les documents à traiter, s'ils ne sont pas au format texte libre (e.g., PDF, WORD) passent par une phase de conversion (qui n'est pas représentée explicitement ici, cette phase posant des problèmes uniquement d'ordre technique).

Nous détaillons dans les sections suivantes les différentes phases du processus de représentation sémantique de documents à travers ses différents modules. Avant de passer aux modules proprement dits, nous décrivons le modèle utilisé pour le RSD.

6.3.2 Le modèle relationnel RSD

Le RSD est représentée par une base de données relationnelle dont le schéma est présenté à la figure 6.2. Il contient les tables nécessaires pour assurer la gestion des éléments fournis par les différents modules.

Description sommaire du schéma

Une source textuelle représentée par l'entité **corpus** (collection textuelle à traiter) contient un ensemble de documents représenté par l'entité **document**. Un document est caractérisé par son identifiant et un ensemble de métadonnées constituées par les éléments Dublin Core. Il contient un

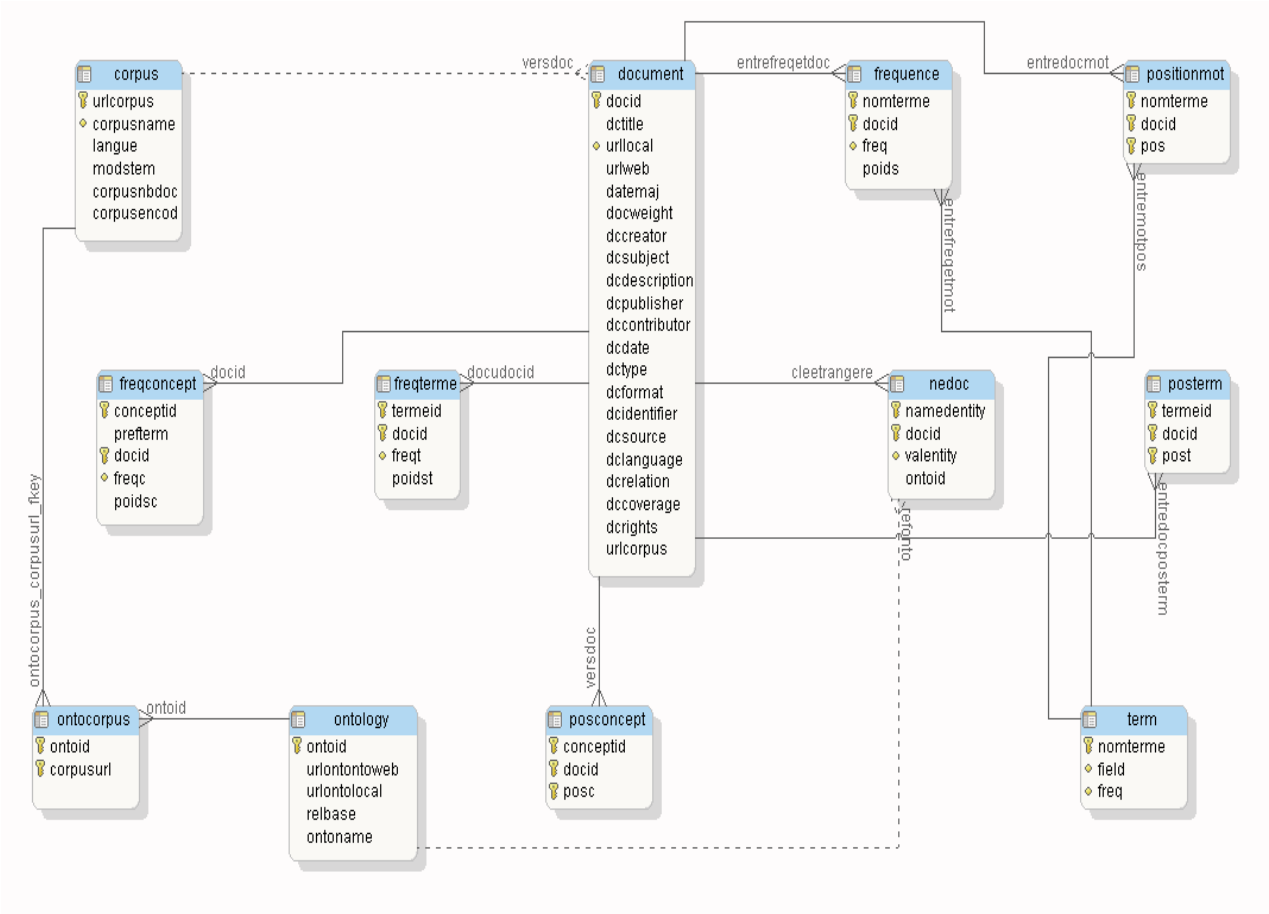


FIG. 6.2 – Schéma relationnel du RSD

ensemble de **mots simples**, de **termes** et de **concepts** issus d'une ontologie. Il peut en outre contenir des **entités nommées** provenant également d'une ontologie, chaque entité nommée ayant une valeur dans le document. Chaque mot, terme ou concept peut avoir plusieurs positions dans un document et leur importance pour le document est dénotée par un poids. Une source textuelle peut être caractérisée par plusieurs ontologies (entité **ontologie**), chaque ontologie étant localisée à une adresse donnée (son URL). Seules les informations concernant l'ontologie sont gérées, l'ontologie dans sa totalité n'est pas stockée au sein du RSD. Le code SQL de création de la base est présentée à l'Annexe B de ce mémoire. Une base est créée pour chaque source textuelle à traiter. Rappelons que la description conceptuelle du RSD est représentée au niveau de l'ontologie globale pour la gestion des sources textuelles.

6.4 Détail des différents modules

6.4.1 HybrideIndex

Le module HybrideIndex procède en deux étapes : la première étape consiste en l'indexation basée sur les mots simples de la source textuelle à traiter et la seconde consiste en l'indexation conceptuelle à partir des ontologies déclarées pour la source.

6.4.2 Indexation basée sur les mots

Le processus d'indexation à base de mots simples est détaillé par la figure 6.3. Ce processus suit l'approche classique d'indexation d'une collection de documents suivant le modèle vectoriel, tout en considérant que les documents en entrée peuvent être multilingues. En entrée du processus nous avons les documents à traiter (1). Ces documents passent par une processus de prétraitement (2) dont le but est d'effectuer un ensemble d'opérations de préparation des documents (tokenisation, purge des mots vides, désuffixation selon la langue). Nous utilisons une liste de mots vides pour chaque langue à traiter. L'algorithme de désuffixation utilisé est l'algorithme de Porter pour l'anglais et une de ses adapations pour le français. Notons que l'opération de désufixation est optionnelle puisque nous fournissons deux modes d'indexation (avec ou sans troncature des mots). Après l'étape (2) nous obtenons une liste de mots constituant le vocabulaire de la source textuelle. C'est ce vocabulaire qui servira à représenter les vecteurs des documents. Le calcul des fréquences et des positions des mots dans chaque document ainsi que de leur poids (suivant la mesure du tf.idf), est effectué à l'étape (3). Nous obtenons enfin la représentation DocMOT stockée dans la base RSD (4). Dans notre approche, la position des

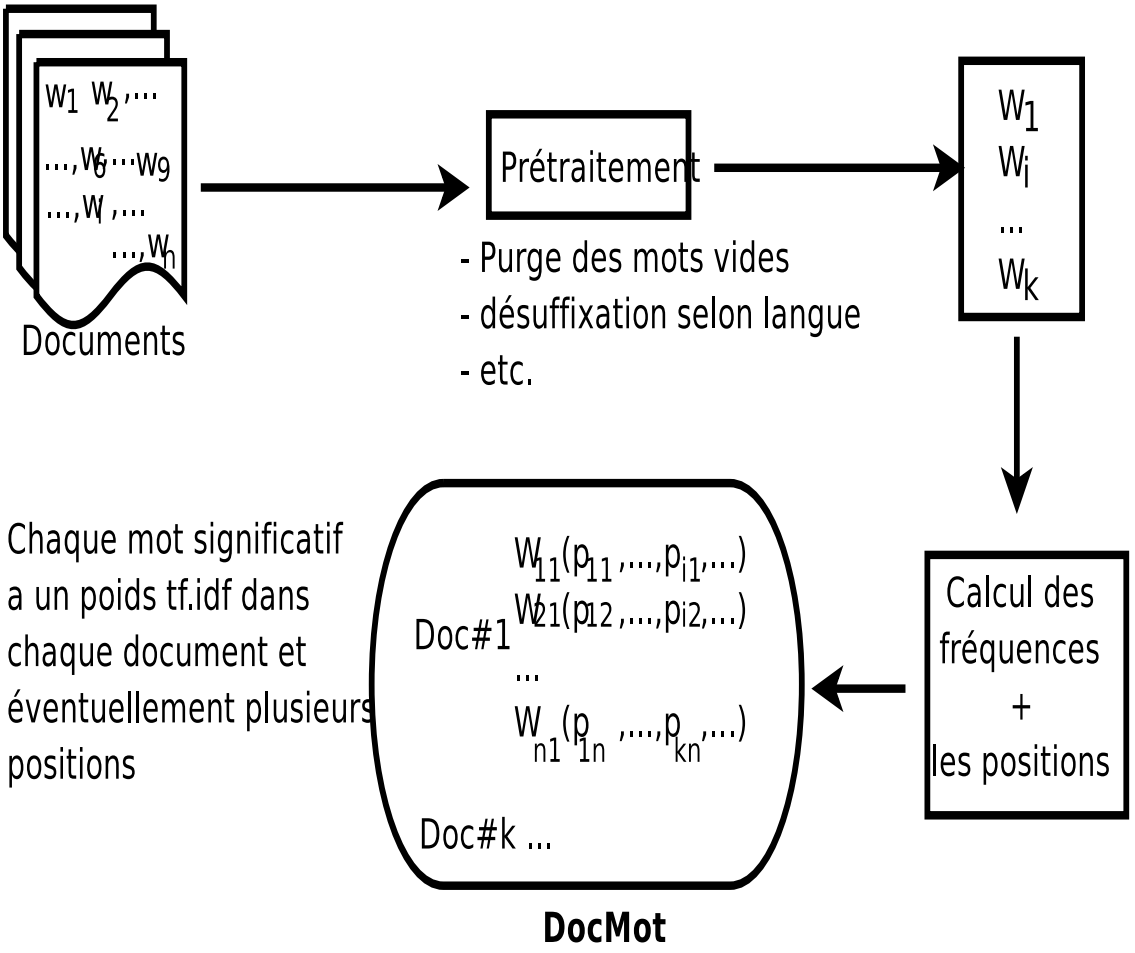


FIG. 6.3 – Caractérisation de documents basée mots

mots est importante car elle permet par la suite de repérer les termes dénotant les concepts dans les documents lors de la phase d'indexation conceptuelle.

6.4.3 Indexation conceptuelle

L'indexation conceptuelle est réalisée en projetant sur les documents de la source textuelle chaque ontologie à utiliser pour la caractérisation sémantique. L'ontologie est projetée sur la représentation des documents obtenue à l'étape précédente. Il existe également une approche symétrique, qui consiste à projeter chaque document à traiter sur l'ontologie. C'est le cas de Baaziz et ses collègues [Baziz *et al.*, 2005], qui utilisent la ressource lexicale WordNet comme ontologie. Dans cette approche, une liste de termes candidats est au préalable extraite des documents, puis les concepts auxquels renvoient ces termes sont identifiés. Comme un terme peut participer à la désignation de plusieurs concepts, une désambiguïsation est nécessaire pour choisir le bon concept selon le contexte.

Le détail de notre approche d'indexation conceptuelle, qui utilise le modèle d'ontologies que nous décrivons à la section suivante, est fourni par la figure 6.4.

Modèle de l'ontologie pour l'indexation conceptuelle

Toute ontologie à utiliser pour l'indexation conceptuelle est représentée suivant un modèle défini de la façon suivante. Une ontologie O est l'ensemble $C, Sub, lex, RelLex$ où :

1. C est l'ensemble de concepts de l'ontologie ;
2. Sub est la relation de subsumption entre concepts ;
3. Lex est le lexique (multilingue) associé aux concepts de l'ontologie ,
4. $RelLex$ est une fonction qui associe un élément de Lex à un concept de l'ontologie ;

Les autres relations associatives entre concepts qui peuvent exister dans une ontologie ne sont pas prises en compte actuellement dans notre approche. L'identification des concepts dans les documents, complétée par les entités nommées, est suffisante pour nos besoins. Ce qui nous intéresse ici est de retrouver des documents relatifs à un ensemble de concepts issus d'une ou plusieurs ontologies.

Chaque concept de l'ontologie est représenté par un ou plusieurs termes (pouvant être multilingues). Un exemple d'un extrait de la définition d'un concept issue de l'ontologie anatomique est donné par le listing ci-dessous. Dans cette définition, seule figure la relation de subsumption.

Listing 6.1 – Extrait de la définition d'un concept de l'ontologie anatomique

```
<owl:Class rdf:ID="AnatCerveau0007461">
...
<rdfs:subClassOf rdf:resource="#BrainAnatomicalStructure"/>
<rdfs:label xml:lang="fr"
            >Noyau intraventriculaire du corps strié</rdfs:label>
        <rdfs:label xml:lang="fr">Noyau caudé</rdfs:label>
        <rdfs:label xml:lang="en">Caudatus</rdfs:label>
        <rdfs:label xml:lang="fr"
            >Noyau intra-ventriculaire du corps strié</rdfs:label>
        <rdfs:label xml:lang="en">Caudate Nucleus</rdfs:label>
        <rdfs:label xml:lang="en"
            >Nucleus caudatus</rdfs:label>

        <rdfs:comment xml:lang="en">
```

```
        Elongated gray mass of the neostriatum located
        adjacent to the lateral ventricle of the brain.
        (MeSH)</rdfs:comment>
</owl:Class>
```

Projeter une ontologie sur un document

Chaque ontologie déclarée pour la source textuelle à traiter (1) est projetée sur la représentation DocMot de la dite source afin d'identifier les concepts de l'ontologie (2). On procède par itération sur chaque document présent dans DocMot. Les concepts eux-mêmes ne sont pas présents explicitement dans les documents, mais à travers les termes qui les dénotent. Il faut à chaque fois identifier le nombre de termes d'un concept (pour la langue du document) présents dans un document donné. Tout d'abord, chaque terme subit la même opération de prétraitement que dans l'étape précédente (DocMot). Ensuite une recherche dans la représentation du document est effectuée afin de vérifier si la suite de mots du terme de l'ontologie y est présente. Si le résultat est positif, le nombre d'occurrences du concept pour le document est incrémentée et la position du concept est notée. A la fin du processus, le poids de chaque concept pour chaque document est calculé afin d'obtenir une représentation de la source textuelle basée sur les concepts (voir Annexe B).

Il arrive que deux concepts soient désignés par deux termes qui se recouvrent. Cette situation peut se présenter lorsque l'un des concepts spécialise l'autre (exemple du cas du concept *Memoire* désigné par le terme "mémoire" et du concept *MemoireTravail* dont un des termes représentants est "mémoire de travail"). L'algorithme peut détecter la présence des ces deux concepts à la même position, même si en réalité il s'agit uniquement du concept *MemoireTravail*. Les positions des concepts sont alors utilisées pour ne garder que les concepts les plus spécifiques de la hiérarchie.

Calcul du poids des concepts Le nombre d'occurences de chaque concept est la somme de ses termes présents dans le document. Pour le calcul du poids de chaque concept nous avons utilisé une mesure que nous avons appelée *cf.idf*, qui est une extension de la mesure du *tf.idf* [Salton et Buckley, 1988]. La mesure du *cf.idf* suit la même logique que celle du *tf.idf*. Elle combine une mesure locale, la fréquence du concept (cf), et une mesure globale (idf) qui permet de relativiser l'importance du concept dans l'ensemble de la source à traiter.

$$W_{ij} = cf_{ij} * idf_i \tag{6.1}$$

Où $cf_{ij} = \sum freq(T_k)$ est la fréquence du concept i dans le document j et $freq(T_k)$ est la fréquence du *kième* terme du concept i dans le document j. Et $idf_i = log\frac{N+1}{df+0.5}$ dénote l'importance relative du concept dans la source textuelle. N est le nombre total de documents et df est le nombre total de documents où le concept i apparaît. Cette approche de pondération est semblable à celle utilisée par Pouliquen et ses collègues dans le cadre de l'indexation de documents médicaux par le thésaurus MeSH [Pouliquen *et al.*, 2002].

D'autres mesures ont été proposées dans la littérature. On peut citer Névéol et ses collègues [Névéol *et al.*, 2006] qui intègrent dans le calcul de la fréquence du concept C (issu du thésaurus MeSH) la fréquence de ses sous-concepts, ou Bazziz [Baziz *et al.*, 2005] qui intègre le nombre de mots du terme dénotant le concept (un concept est équivalent à un noeud du thésaurus WordNet, et dans ce cas il est dénoté par un seul terme). Dans notre cas, c'est à la phase de recherche de

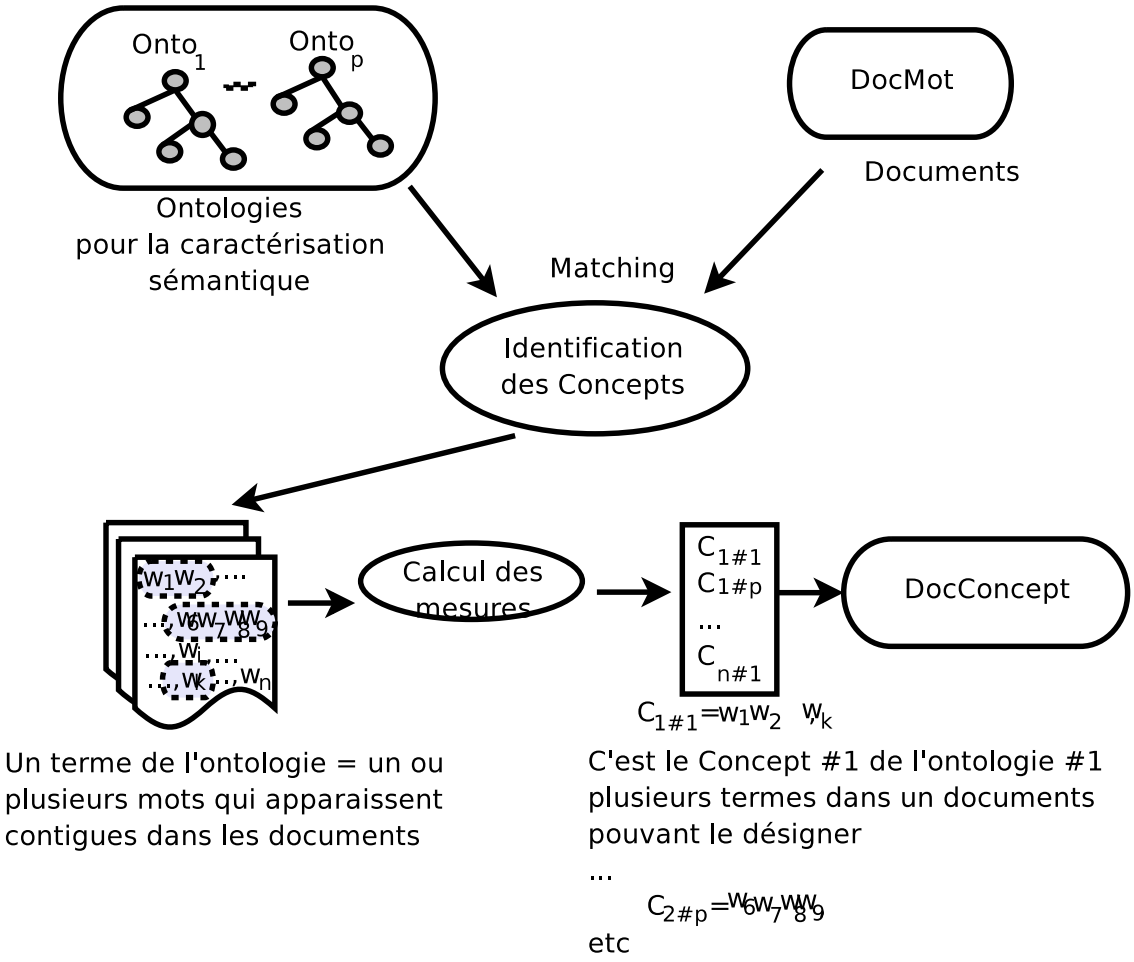


FIG. 6.4 – Schéma synoptique de l'approche de projection d'une ontologie sur un ensemble de documents

documents que peuvent être intégrés les sous-concepts des concepts utilisés dans la requête. Ainsi une recherche sur le concept *Memoire* inclura la recherche sur les concepts *MemoireRetrograde*, *MemoireAnterograde*, etc.

6.4.4 L'annotation manuelle

Dans le cadre de la participation de notre équipe au projet Européen NOESIS (2004-2006, 6^{eme} PCRD)[47], nous avions la tâche de concevoir et implémenter un outil pour l'annotation manuelle de documents avec pour objectif le partage des connaissances et la recherche d'information sur le Web. Le projet Noesis a pour objectif de fournir une plateforme d'aide au diagnostic médical, avec pour domaine d'expérimentation les maladies cardiovasculaires.

Dans le cadre de notre travail de thèse, nous avons réalisé un état de l'art du domaine de l'annotation, puis participé à la phase de conception de l'outil d'annotation. Nous avions aussi pour objectif l'intégration des résultats d'annotation dans notre représentation des documents. L'outil d'annotation a été conçu avec un double objectif : i) permettre à une communauté d'usa-

[47] http ://www.noesis-eu.org

gers de partager leurs expériences et leurs connaissances à travers l'annotation de documents scientifiques ; ii) corriger les erreurs éventuelles qui peuvent résulter d'une indexation sémantique automatique de documents basée sur une ontologie.

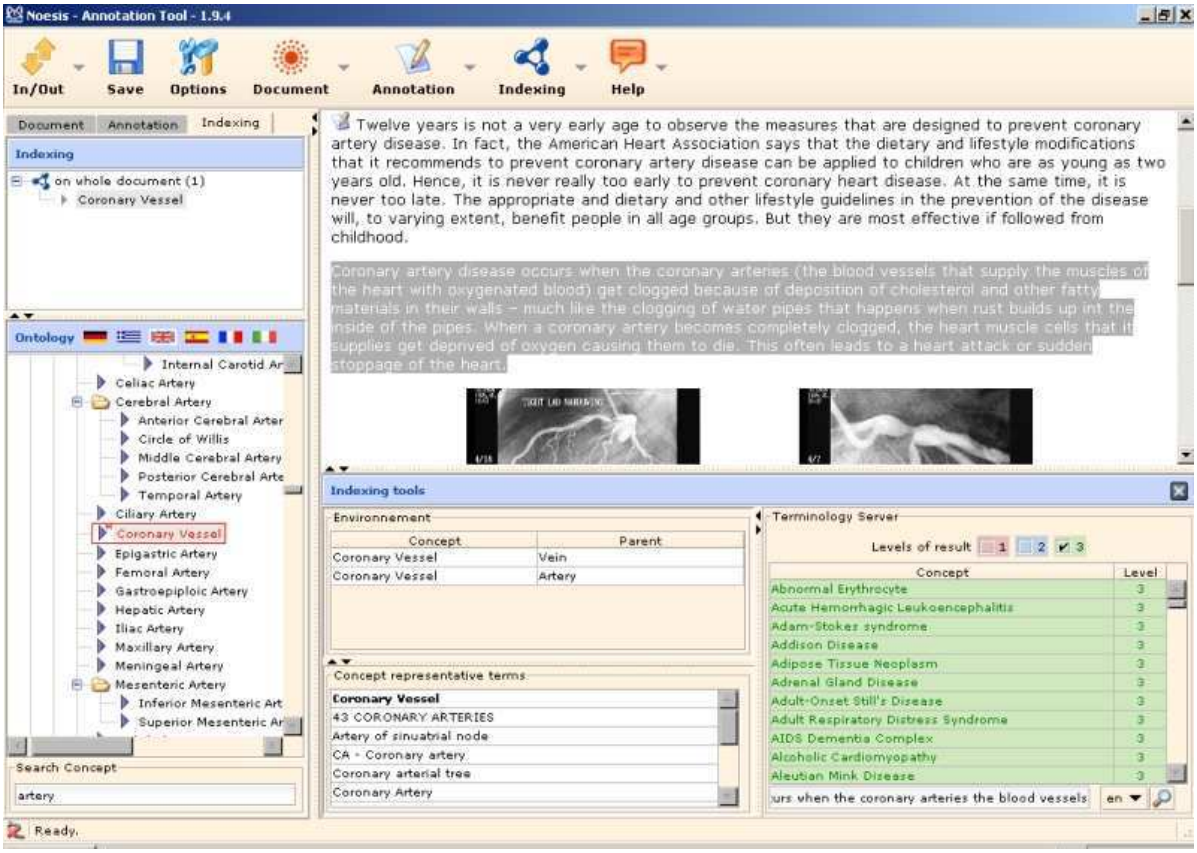


FIG. 6.5 – L'outil d'annotation manuelle de documents

Une des originalités de l'outil d'annotation est l'intégration des trois catégories d'annotation (catalogage, textuelle, sémantique ou conceptuelle) dans un même environnement [Patriarche *et al.*, 2005]. L'utilisation de l'outil dans le cadre de la représentation sémantique de documents se situe à deux niveaux :

1. l'ajout d'annotations de catalogage (métadonnées Dublin Core) sur les documents d'une source textuelle ;

2. la correction de l'indexation conceptuelle automatique effectuée par le module HybrideIndex.

Les annotations produites par l'outil d'annotation sont stockées sous format XML suivant trois DTD spécifiques correspondant respectivement aux annotations de catalogage, textuelle, et sémantique. La base RSD est mise à jour en fonction des résultats des annotations de catalogage et sémantique.

La dernier processus utilisé pour la caractérisation sémantique de documents est le processus d'identification des entités nommées dans la source textuelle.

6.4.5 Identification d'entités nommées

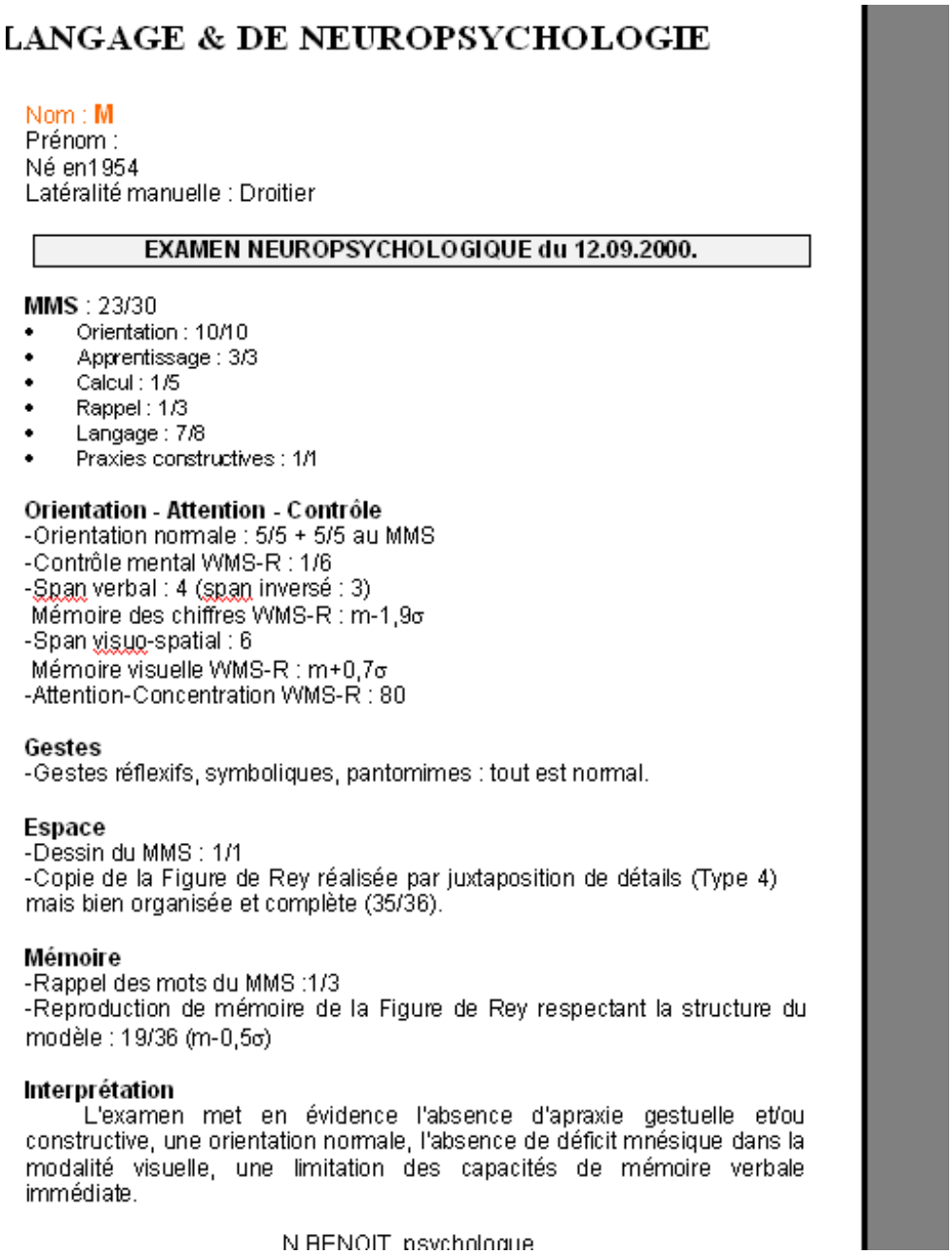
La tâche d'identification des entités nommées fait partie des cinq tâches considérées par le programme MUC [Chinchor, 1997] comme l'extraction d'information [Cunningham, 2005]. L'identification des entités nommées s'occupe du repérage dans les documents des noms de personnes, de lieux géographiques ou d'organisations. La notion d'entités nommées a été étendue aux items désignant des montants, des unités de mesures, des dates, ... [Popov *et al.*, 2003]. Dans le cadre du projet BC^3, L'analyse des comptes rendus médicaux nous a permis de dégager un lien avec les données gérées à travers des bases de données de patients. Les comptes rendus textuels suivent une forme élaborée permettant leur traitement pour l'identification des performances aux tests. Un exemple d'un extrait de compte rendu est donné à la figure 6.6.

Les comptes rendus dont nous disposons sont anonymisés (on ne peut pas identifier un patient à partir d'un compte-rendu qui le concerne). L'identification dans ce cas des noms de patients et des adresses n'était pas possible. Nous avons considéré les résultats des tests cognitifs présents dans les documents. L'extrait présenté à la figure 6.6 indique que lors de l'examen neurospychologique du 12/09/2002, les fonctions cognitives de l'orientation, la mémoire, l'attention, etc. ont été évaluées. Par exemple, la mémoire a été évaluée à travers le test cognitif *Rappel des mots du MMS*. Pour ce test, le patient a obtenu le score de 1 sur un maximum de 3. Pouvoir identifier ce genre de résultats permet d'interroger la base des comptes rendus de façon plus structurée (e.g., trouver les comptes rendus pour lesquels le patient a obtenu un score de X au test du Rappel des mots du MMS).

Nous avons implémenté un algorithme utilisant l'ontologie, qui formalise la connaissance des tests cognitifs, et les expressions régulières pour l'identification de valeurs entières de résultats de tests cognitifs. L'implémentation a été faite à travers le module RENO (Reconnaissance d'Entités Nommées à travers une Ontologie), codé en Java. La figure 6.7 décrit sommairement son principe. Pour chaque document à traiter, un parcours de l'ontologie est effectué. Pour chaque concept on construit un pattern à identifier en combinant chaque terme dénotant le concept avec la règle permettant d'identifier une valeur de test (les valeurs des tests sont présents dans les documents sous la forme X, X :Y, X sur Y, X/Y). Si l'évaluation du pattern sur le document est concluant, on récupère la valeur du test et on crée une entrée dans la base RSD. Ce principe, simple, donne des résultats encourageants. Les comptes-rendus médicaux présentent en effet une certaine régularité. Les valeurs qui ne sont pas identifiées sont souvent dues au fait que dans le document à traiter le test est désigné par des termes qui ne sont pas présents dans l'ontologie. C'est pourquoi un enrichissement terminologique de l'ontologie à partir de la collection de documents à traiter est préconisé [Simonet *et al.*, 2005][Simonet *et al.*, 2006].

6.5 Évaluations préliminaires de l'approche d'indexation conceptuelle

Notre premier objectif ici est de comparer les résultats obtenus par l'approche d'indexation conceptuelle en prenant comme base d'indexation le stem puis le mot. Le second objectif est de comparer les deux modes de représentation d'un document (représentation à base de mots (stem)) et représentation à base de concepts. Nous avons utilisé deux ontologies, l'une sur le cerveau (l'ontologie des fonctions cognitives) pour le traitement de documents en français, l'autre sur la cardiologie [Gedzelman *et al.*, 2005][Simonet *et al.*, 2006] (construite à partir du thésaurus MeSH et enrichi à travers le vocabulaire UMLS) utilisée dans le cadre du projet NOESIS (voir listing 6.2).



LANGAGE & DE NEUROPSYCHOLOGIE

Nom : M
Prénom :
Né en1954
Latéralité manuelle : Droitier

EXAMEN NEUROPSYCHOLOGIQUE du 12.09.2000.

MMS : 23/30
- Orientation : 10/10
- Apprentissage : 3/3
- Calcul : 1/5
- Rappel : 1/3
- Langage : 7/8
- Praxies constructives : 1/1

Orientation - Attention - Contrôle
-Orientation normale : 5/5 + 5/5 au MMS
-Contrôle mental WMS-R : 1/6
-Span verbal : 4 (span inversé : 3)
Mémoire des chiffres WMS-R : m-1,9σ
-Span visuo-spatial : 6
Mémoire visuelle WMS-R : m+0,7σ
-Attention-Concentration WMS-R : 80

Gestes
-Gestes réflexifs, symboliques, pantomimes : tout est normal.

Espace
-Dessin du MMS : 1/1
-Copie de la Figure de Rey réalisée par juxtaposition de détails (Type 4) mais bien organisée et complète (35/36).

Mémoire
-Rappel des mots du MMS :1/3
-Reproduction de mémoire de la Figure de Rey respectant la structure du modèle : 19/36 (m-0,5σ)

Interprétation
L'examen met en évidence l'absence d'apraxie gestuelle et/ou constructive, une orientation normale, l'absence de déficit mnésique dans la modalité visuelle, une limitation des capacités de mémoire verbale immédiate.

N BENOIT psychologue

FIG. 6.6 – Extrait d'un compte rendu neuropsychologique

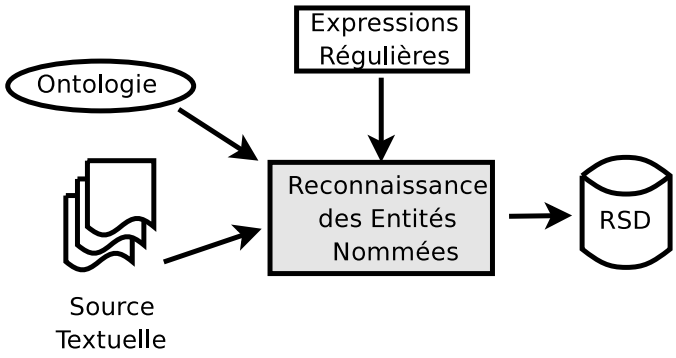


FIG. 6.7 – Processus de reconnaissance des entités nommées

Listing 6.2 – Extrait du concept D001019 (Rupture de l'aorte) de l'ontologie sur la cardiologie

```
<owl:Class rdf:ID="D001019">
<skos:scopeNote rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Tearing of aortic tissue. It may be rupture of an aneurysm or it
     may be due to trauma.</skos:scopeNote>
 <skos:prefLabel xml:lang="en">Aortic Rupture</skos:prefLabel>
 <skos:prefLabel xml:lang="fr">Rupture de l'aorte</skos:prefLabel>
 <skos:altLabel xml:lang="en">Aorta Aneurysm Rupture</skos:altLabel>
 <skos:altLabel xml:lang="en">Aortic aneurysm rupture</skos:altLabel>
 <skos:altLabel xml:lang="fr">Anévrisme aortique rompu</skos:altLabel>
 <skos:altLabel
        xml:lang="fr">Anévrisme rompu de l'aorte</skos:altLabel>
                ....
 <rskos:altLabel
        xml:lang="fr">Rupture d'anévrysme de l'aorte</skos:altLabel>
 <skos:altLabel
        xml:lang="fr">Rupture Anevrysme AORTIQUE</skos:altLabel>
 <rdfs:subClassOf rdf:resource="#D017542"/>
 <rdfs:subClassOf rdf:resource="#D001014"/>
 </owl:Class>
```

6.5.1 Les collections de documents utilisées

Nous avons utilisé une collection de documents en français et deux collections de documents en anglais, indexées respectivement par l'ontologie des fonctions cognitives et par l'ontologie sur la cardiologie.

La première, que nous nommons Cerveau, est un ensemble de documents en français constitué de 45 comptes rendus médicaux issus du projet BC^3 et de 285 documents relatifs au cerveau, que nous avons sélectionnés sur le Web à travers Yahoo! API[48] (en fournissant une liste de termes issus de l'ontologie fonctionnelle du cerveau).

La seconde collection, appelée Cardio est constituée de 430 articles scientifiques en anglais dans le domaine de la cardiologie, obtenus à partir du portail Biomedcentral[49]. Les documents

[48] http ://developper.yahoo.net (consultée le 09 Octobre 2006)
[49] www.biomedcentral.com (consulté la dernière fois le 09 Octobre 2006)

de la collection Cardio ont été sélectionnés à partir de 7 journaux traitant chacun d'un sous domaine de la cardiologie. La sélection des documents dans des journaux différents est motivée par le besoin de constituer une collection qui couvre l'ontologie Cardio. Ainsi nous avons 41 documents du journal "*Cardiovascular diabetology*" ; 94 documents du journal "*BMC cardiovascular disorders*" ; etc.

La troisième collection est constituée d'un sous ensemble de la collection standard OHSUMED [Hersh *et al.*, 1994]. La collection OHSUMED est un ensemble de 345.566 références tirées de la base de données en ligne MEDLINE. Les références sont sélectionnées à partir de 270 journaux dans le domaine médical durant la période 1987-1991. Nous avons sélectionné les 2.182 premiers documents pour un test avec l'ontologie sur la cardiologie.

Les tableaux suivants récapitulent les résultats obtenus dans l'identification des concepts avec les deux modes (stem et mot).

Tab. 6.1 – Indexation conceptuelle basée sur le stem

Stem		
Collection	Concepts identifiés	Nombre de positions
Cerveau	2.630	11.691
Cardio	15.969	115.977
OHSUMED	4.492	9.748

Tab. 6.2 – Indexation conceptuelle basée sur le mot

Mot		
Collection	Concepts identifiés	Nombre de positions
Cerveau	2.346	10.606
Cardio	14.380	107.199
OHSUMED	3.591	7.372

Ces deux tableaux montrent que si l'indexation conceptuelle est faite en partant d'une indexation basée sur le stem on identifie plus de concepts. En examinant d'une part l'ontologie et d'autre part certains documents des collections traitées, on se rend compte que la désuffixation introduit du bruit par rapport à l'ontologie utilisée. La forme désuffixée d'un terme donné de l'ontologie peut correspondre à la forme désuffixée d'un terme de documents sans que les termes eux-mêmes soient identiques.

En plus de cette considération, le nombre total de positions (occurrences) élevé, trouvé dans le cas de désuffixation est aussi dû au fait que dans les ontologies utilisées, on trouve parfois, en plus du terme pour désigner un concept, toutes ses variantes. Dans certains cas la désuffixation des deux termes est équivalente (le vocabulaire des ontologies utilisées est parfois enrichi à partir d'UMLS qui considère toutes les variantes d'un terme).

Les résultats de l'indexation conceptuelle ont été utilisés pour la classification de documents en utilisant les cartes auto-organisatrices de Kohonen (SOM : Self-Organizing Map) [Kohonen, 1984]. L'objectif était de comparer le résultat de la classification de documents (collection Cardio présentée ci-dessus) suivant deux modes de représentation : une représentation basée sur le stem des mots et une représentation basée sur les concepts identifiés à partir d'une ontologie.

L'étude a montré qu'en plus de réduire considérablement le nombre de dimensions pour l'espace de classement (réduction de 25.762 pour la représentation basée sur les stems à 4.315 pour la représentation basée sur les concepts), la représentation basée sur les concepts donne de meilleurs résultats (moins de documents mal classés). Les résultats détaillés de cette étude sont disponible dans [Pham *et al.*, 2007].

Les modules que nous avons présentés ci-dessous ont été implémentés avec le langage Java. Le module qui effectue l'indexation basée sur les termes simples et sur les concepts ainsi que le module de recherche utilisent l'API Lucene de la fondation Apache[50]. Cette API fournit les fonctionnalités nécessaires, adaptables, pour l'indexation et la recherche. Nous avons implémenté la visualisation du résultat de l'indexation basée sur les concepts en langage Python (avec le module MySQLdb pour l'accès au RSD stocké dans une base MySQL). L'ontologie à explorer est représentée à l'aide d'un graphe hyperbolique (nous avons utilisé l'API libre HyperGraph[51]). Une visualisation de l'index pour le corpus Cardio est présentée à la figure 6.8.

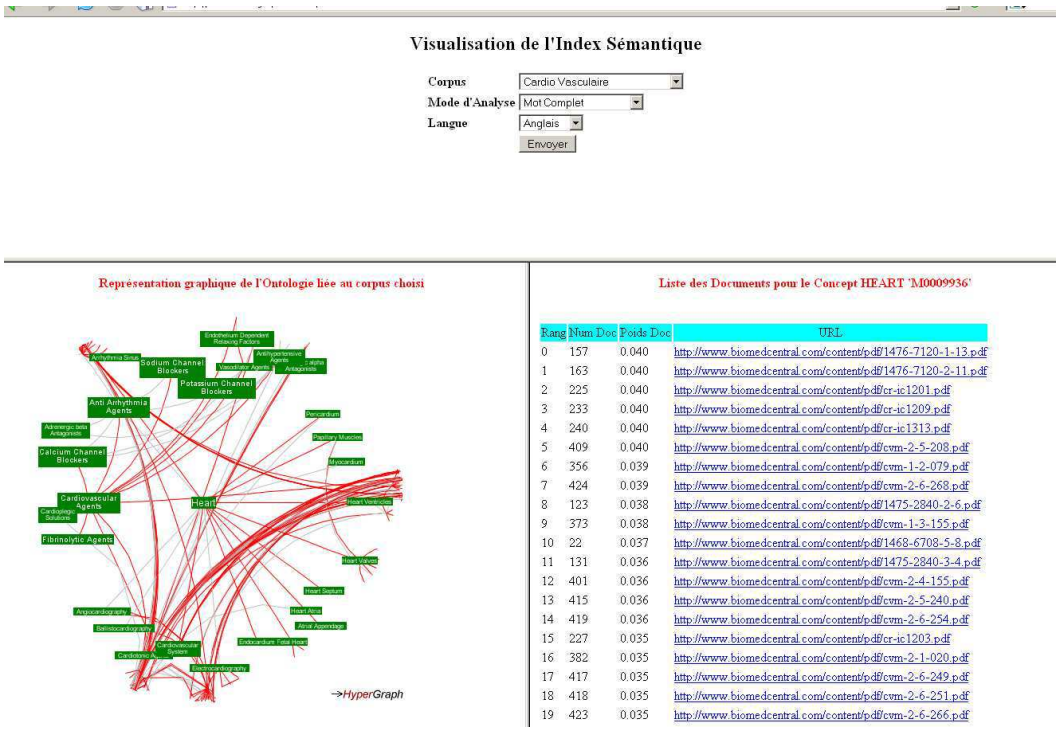


FIG. 6.8 – Visualisation de l'index (indexation basée sur les concepts)

6.6 Conclusion

Nous avons présenté dans cette section la représentation sémantique de documents en vue de leur intégration dans l'infrastructure de gestion unifiée des données structurées et informelles. L'approche d'indexation conceptuelle que nous avons mise en oeuvre n'utilise pas de techniques de désambiguïsation automatique comme c'est le cas par exemple dans le système SemTag [Dill *et al.*, 2003]. En effet, l'outil d'annotation manuelle fait partie intégrante de l'infrastructure et

[50] lucene.apache.org
[51] http ://hypergraph.sourceforge.net/

il est destiné notamment à la tâche de désambiguïsation. Nous avons mentionné que certaines approches utilisent la ressource lexicale WordNet pour la tâche de désambiguïsation de termes anglais. Dans notre cas, l'aspect multilingue (et la gestion des documents français en particulier) est important, et la couverture de WordNet pour notre premier champ d'application, la médecine, reste limitée [Burgun et Bodenreider, 2001]. Pour les langues autres que l'Anglais il reste l'alternative EuroWordnet [Vossen, 1998] facilement accessible, mais qui n'offre pas actuellement le même niveau de service que WordNet.

Nous envisageons une utilisation des résultats fournis par la reconnaissance des entités nommées dans une perspective de fouille de bilans neuropsychologiques [Pisetta *et al.*, 2006].

Nous présentons brièvement au chapitre 7 (consacré au Serveur d'Ontologies) la recherche combinant termes simples et concepts.

Chapitre 7

ServO : Serveur d'Ontologies

7.1 Introduction

Compte tenu de l'utilisation de plusieurs ontologies (parfois de grande taille) dans notre système d'intégration, nous avons été amenés à proposer et implémenter une approche de gestion conjointe de plusieurs ontologies. Nous allons détailler dans ce chapitre notre approche, intitulée **ServO** pour Serveur d'Ontologies. ServO est une approche générique de gestion conjointe de plusieurs ontologies qui s'appuie sur l'utilisation d'une part des techniques de la recherche d'information et d'autre part des technologies du Web Sémantique (WS). Les ontologies utlisées dans le cadre du WS sont pour une grande part représentées dans les deux langages recommandés par le W3C, RDF et OWL, langages qui ont fait l'objet d'une présentation dans la partie consacrée à l'état de l'art. Notre serveur d'ontologies a été conçu afin de gérer des ontologies décrites dans ces langages.

7.2 Principales motivations

L'approche de gestion unifiée que nous avons proposée intègre la prise en compte de plusieurs types d'ontologies (chapitre 4). Nous énumérons ci-dessous quelques éléments de motivation pour la proposition de ServO :

1. le processus de représentation hybride des documents, qui aboutit au *RSD*, nécessite l'usage de plusieurs ontologies (comme sources externes de connaissances), fournissant chacune une vue particulière sur la source à traiter. Pour effectuer une recherche, la première intuition amène à saisir naturellemment comme requête une liste de termes en langue naturelle, alors que les documents sont indexés par des concepts. Il faut alors permettre une reformulation de la requête en fournissant les "bons" concepts des ontologies associés aux termes en langue naturelle ;
2. avant d'utiliser une ontologie dans un processus de caractérisation sémantique, on peut l'examiner afin de savoir si elle est pertinente par rapport à la source à caractériser. Cet examen peut se faire manuellement si l'ontologie en question contient peu de concepts. Cependant, si elle contient des centaines, voire des milliers de concepts (dénotés chacun par plusieurs termes), comme c'est souvent le cas dans le domaine médical, une assistance automatique est nécessaire ;
3. la tâche de construction d'une nouvelle ontologie peut nécessiter la réutilisation de plusieurs ressources (vocabulaires contrôlés, ontologies) déjà existantes. Ces ressources sont souvent

disponibles dans les langages RDF/OWL. Une aide dans le choix des ressources peut se faire via un environnement qui permet rapidement d'interroger le contenu des ressources.

4. il existe aujourd'hui des outils qui permettent la recherche d'ontologies, par exemple Swoogle [Ding *et al.*, 2004] et OLS (Ontology Lookup Service) [Cote *et al.*, 2006]. Swoogle, dont le développement a commencé en 2004 à l'Université de Washington, est un exemple de moteur de recherche d'ontologies généraliste sur le Web (le choix du nom de l'outil n'est pas fortuit). Ce cadre limite sa personnalisation dans un contexte spécifique (ajout facilité de nouvelles ressources – l'indexation de toute nouvelle ressources nécessite une approbation et certain délais –, recherche plus fine par la spécification de certaines propriétés, etc.). OLS a été conçu dans le cadre du projet PRIDE (PRoteomics IDEntifications database)[52] pour gérer les ontologies et vocabulaires contrôlés disponibles dans le format OBO (Open Biomedial Ontologies), l'objectif étant de fournir un accès centralisé aux ontologies OBO. Cet objectif d'une part, et la non prise en compte d'ontologies au format OWL/RDF d'autre part, limitent considérablement son utilisation dans un cadre plus général.

Nous avons proposé une approche de gestion de plusieurs ontologies décrites au format OWL/RDF, permettant leur gestion dynamique à travers une base unique et offrant un support aux activités suivantes :

1. **l'indexation d'ontologies**, en permettant facilement d'ajouter de façon automatique de nouvelles ressources ;
2. **la recherche d'ontologies et la recherche de concepts dans les ontologies**, fonctionnalités réutilisables dans l'activité de construction d'une nouvelle ontologie. Plus généralement, ces fonctionnalités permettent d'aider un ingénieur de la connaissance ou un spécialiste de l'intégration de données, à retrouver des ontologies d'intérêt (concernant un domaine précis) par rapport à son activité. Le résultat d'une recherche de concepts peut être utilisé dans la reformulation de requêtes dans le cadre de l'interrogation de sources indexées conceptuellement ;
3. l'assistance dans la mise en correspondance d'ontologies, en permettant de retrouver les éléments de différentes ontologies à faire correspondre.

7.3 Présentation générale du principe

Il y a une certaine analogie entre le problème de recherche d'ontologies et le problème de recherche d'information classique, que nous exploitons en adoptant une démarche de RI adaptée aux ontologies. Une ontologie RDF/OWL est un *document sémantique* qui peut être vu comme une collection à traiter (dans le sens de la RI). Les constituants de la collection à traiter sont alors les classes (qui traduisent les concepts de l'ontologie), les métadonnées et les instances éventuelles de l'ontologie. Dans une ontologie décrite en RDF/OWL, un certain nombre d'attributs et de relations sont définis. Ces attributs et relations peuvent être affectés à une ou plusieurs classes. Les attributs ont comme domaine de valeur les types de données primitifs comme les entiers, les chaînes de caractères, etc., tandis que les relations permettent de lier les classes entre elles. Le diagramme de la figure 7.1 décrit la représentation simplifiée d'une ontologie.

Dans la suite, le terme ontologie désigne une ontologie décrite dans les langages OWL/RDF et pour simplifier le terme OWL sera utilisé à la place de RDF/OWL.

Nous détaillons ci-dessous les classes du diagramme :

[52] http ://www.ebi.ac.uk/pride/ (consulté la dernière fois le 3 octobre 2006)

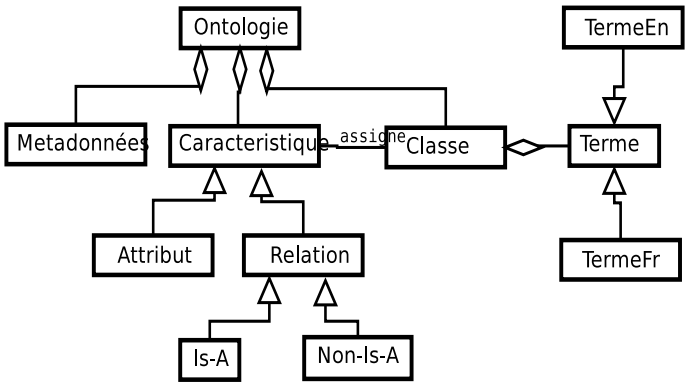


FIG. 7.1 – Diagramme simplifié de représentation d'une ontologie

1. **Ontologie** représente une ontologie à gérer dans la base ;
2. **Métadonnées** représente, pour chaque ontologie, l'ensemble de ses métadonnées ;
3. **Caracteristique** représente les éléments qui servent à caractériser une ontologie. Elle a comme composants les attributs et les relations ;
4. **Classe** représente toute classe OWL (traduisant les concepts des ontologies) à gérer dans la base d'ontologies ;
5. **Terme** représente les termes (multilingues) dénotant toute classe d'une ontologie ;
6. **TermeFr** représente les termes d'un concept (classe OWL/RDF) exprimés en français ;
7. **TermeEn** représente les termes d'un concept exprimés en anglais ;
8. **Attribut** représente les attributs définis pour chaque ontologie à gérer ; un arrtibut dénote la relation entre un concept et un type de données de base (entier, chaîne de caractères, etc.) ;
9. **Relation** désigne les relations entre classes définies dans une ontologie. Une relation a un domaine de valeur et domaine d'arrivée. Une relation de nom donné peut être affectée à plusieurs concepts.
10. **Is-A** représente la relation de subsumption entre concepts d'une ontologie ;
11. **Non-Is-A** représente tous les autres types de relations sémantiques entre concepts d'une ontologie.

7.4 Les modules de ServO

L'architecture de ServO comprend trois couches principales : une couche de gestion des ontologies, une couche recherche d'information et une couche pour le stockage. ServO utilise le module OntoManage pour la gestion des ontologies. La couche RI utilise l'API Lucene d'Apache[53]. La figure 7.2 présente l'architecture de ServO.

Le module **Indexation** s'occupe d'indexer les différentes ontologies selon le modèle que nous décrivons à la section 7.5. Deux niveaux d'indexation sont définis : le niveau *Indexation Métadonnées* ou niveau d'indexation globale, qui permet d'indexer les métadonnées d'une ontologie, et le niveau *Indexation Classes*, qui permet d'indexer les classes d'une ontologie.

[53] http ://lucene.apache.org/

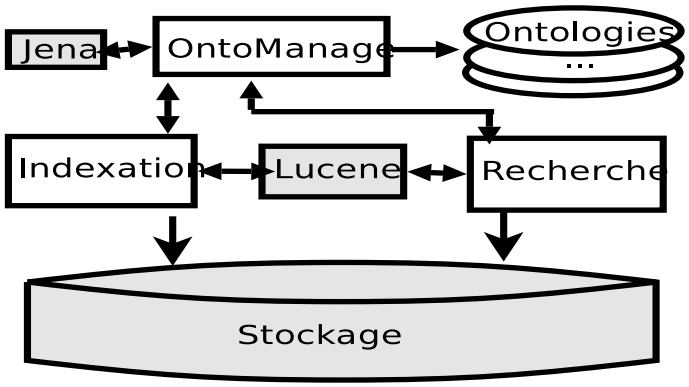


FIG. 7.2 – Architecture du serveur d'ontologies

Le module **Recherche** s'occupe de la recherche sur l'index construit à partir du module Indexation. Il offre plusieurs possibilités de recherche que nous détaillons à la section 7.7.

Le module **OntoManage** a en charge la gestion d'une ontologie décrite en RDF/OWL. Il implémente les fonctions nécessaires à la manipulation d'une ontologie : lister les classes de l'ontologie, les termes (éventuellement filtré par langue) d'une classe donnée, les relations et attributs associés à une classe, etc. Il permet en outre l'exploration de l'ontologie en offrant la possibilité de navigation sous-classes/super-classes jusqu'à un niveau donné.

Trois modes de stockage de l'index sont possibles : stockage de l'index en mémoire vive, stockage dans un répertoire temporaire et enfin stockage dans un répertoire pérenne. Cette diversité offre plusieurs modes d'utilisation de ServO selon le contexte.

7.5 Modèle de recherche d'ontologies

Un modèle de recherche spécifie comment les documents et les requêtes doivent être représentés, mais aussi les détails de la fonction de recherche à utiliser. Il détermine en outre la notion de pertinence. La pertinence peut être binaire (cas du modèle booléen) ou continue (liste triée de résultats). ServO doit permettre de poser des requêtes par combinaison booléenne de termes, d'attributs ou de relations. Nous avons adopté une approche qui combine l'approche booléenne et l'approche vectorielle (présentée dans la première partie de ce mémoire) pour déterminer la pertinence entre les requêtes et les éléments de la base d'ontologies (d'où le choix de l'API Lucene pour l'implémentation).

Dans le modèle vectoriel, chaque document ou requête est représenté par un vecteur dans un espace où chaque dimension est associée à un terme d'indexation. Il s'agit dans un premier temps de déterminer quels vont être les termes d'indexation dans le cadre de la gestion de plusieurs ontologies. On peut faire l'analogie suivante. La gestion de plusieurs ontologies peut être assimilée à la gestion de plusieurs collections de documents, chaque ontologie étant une collection de documents, avec toutefois une particularité : l'ontologie elle-même, comme objet indexé, peut faire l'objet d'une recherche et donc peut être retournée en réponse à une requête. Nous avons défini deux niveaux d'indexation : le niveau global, qui indexe les métadonnées de chaque ontologie, et le niveau des classes, qui indexe les éléments d'une ontologie à gérer. L'index est donc constitué par deux ensembles de vecteurs : un premier ensemble, constitué par le vocabulaire des métadonnées, et un second ensemble, constitué par tous les termes utilisés dans les ontologies. Une requête sera aussi représentée dans cet espace vectoriel. Le score entre

requêtes et documents est calculé suivant la mesure dite du cosinus (implémentée par défaut par l'API Lucene).

La figure 7.3 présente la structure d'un index ontologique. Un index est constitué d'une ou plusieurs ontologies. Chaque ontologie est composée d'un ensemble de métadonnées et d'un ensemble de concepts. Les ensembles de métadonnées et de concepts sont constitués de plusieurs champs qui sont les unités d'indexation de base du modèle. Un champ a un nom et un contenu. Le nom peut être utilisé lors de la phase de recherche pour préfixer les termes de requêtes, donnant ainsi une possibilité de filtrage du domaine de recherche. Chaque vecteur contient des éléments

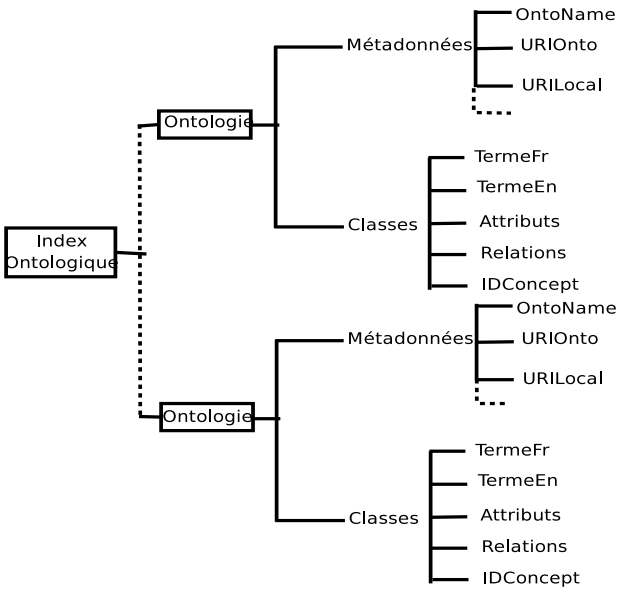


FIG. 7.3 – Structure d'un index ontologique

pondérés tenant compte des champs qui ont été définis. La similarité entre une requête et une élément de l'index ontologique sera calculée par la mesure de l'angle entre le vecteur de la requête et le vecteur de l'élément ontologique.

7.6 Stratégie d'indexation

Comme le montre la figure 7.3, pour indexer une ontologie suivant les deux niveaux définis, plusieurs champs sont utilisés. Chaque champ est traité différemment selon son type. Ainsi certains champs tels que les différents termes associés à une classe subissent un processus de désuffixation (stemming) en utilisant l'algorithme de Porter ou son adaptation pour la langue française, d'autres, comme les identifiants, sont indexés tels quels.

Nous détaillons ci-dessous le processus d'indexation au niveau des métadonnées et au niveau des classes.

7.6.1 Métadonnées comme *documents* à indexer

Les métadonnées donnent les informations descriptives d'une ontologie. Elles permettent, une fois indexées, d'effectuer des recherches dont le résultat est un ensemble trié d'ontologies. Le tableau 7.6.1 présente une liste non exhaustive des éléments de métadonnées utilisés pour

Métadonnée	Description	Exemple
OntoName	Désignation de l'ontologie	Ontologie de l'anatomie du cerveau
URIOnto	URI de l'ontologie (adresse Web)	`http://www-timc.imag.fr/GD/AnatomieCerveau.owl`
URILocal	URL local de l'ontologie	`file://c:/ontos/anatcerveau.owl`
OntoNameSpace	Espace de nommage de l'ontologie	`http://www-timc.imag.fr/GD/anatomieCerveau.owl#`

TAB. 7.1 – Exemples de métadonnées utilisées dans le cadre de l'indexation

la description d'une ontologie. OntoName est la désignation en langue naturelle de l'ontologie (e.g., Fonctions Cerveau, Breast Cancer ontology). Les métadonnées de l'ontologie sont obtenues à travers le module OntoManage.

7.6.2 Les classes de l'ontologie OWL comme *documents* à indexer

Chaque classe d'une ontologie est considérée comme un document à indexer. Une classe (ayant un identifiant unique qui est son URI) est donc représentée sous sa forme plate en considérant les différents champs définis ci-dessus pour chaque classe. Une classe d'une ontologie peut être dénotée par plusieurs termes (dans plusieurs langues) et est caractérisée par un ensemble d'attributs et un ensemble de relations. Les attributs et les relations sont soit déclarés explicitement pour la classe, soit hérités d'une classe parente. Afin de permettre le calcul de différents niveaux de similarité entre classes, deux modes d'indexation d'une classe sont possibles : une indexation en mode partiel et une indexation en mode étendu.

1. Le mode d'indexation partiel ne prend en compte que l'identifiant de la classe et des termes multilingues. Ce mode est indiqué en particulier pour traiter des ontologies terminologiques définies antérieurement ;
2. le mode complet considère la description entière de la classe, c'est-à-dire en plus de l'identifiant et des termes multilingues, les attributs et les relations sont pris en compte. Dans ce cas, deux possiblités d'ajout des attributs et relations sont mises en oeuvre : une description comprenant uniquement les attributs explicitement définis pour la classe et une description prenant en compte les attributs et relations définis au niveau des classes parentes.

La figure 7.4 donne l'exemple d'une classe OWL représentant le concept de l'ontologie anatomique du cerveau dont l'identifiant est *AnatCerveau0007759* (Cortex Cérébelleux) ayant comme ensemble de relations (*Is-Anatomical-Part*, *Present-To-Source*, *Has-Semantic-Type*, etc.) et comme attribut Has-UMLS-CUI. L'indexation partielle de cette classe OWL prendra en compte uniquement l'identifiant *AnatCerveau007759* et les termes français (Ecorce cérébelleuse, Cortex cérébelleux) et anglais (Cerebellar cortex). L'indexation étendue prendra en compte les attributs et relations associés à la classe.

Pour ajouter un concept de l'ontologie dans l'index d'ontologies, nous utilisons l'algorithme *addConcept2Index* dont le pseudo-code est présenté dans le listing 7.1. Les booléens *full* et *parent* indiquent respectivement si la classe doit être indexée dans sa totalité (termes, attributs, relations) et si les relations et attributs hérités doivent être considérés. L'algorithme construit le document à indexer en concaténant champ par champ les éléments à indexer. Ainsi, pour la classe de la figure 7.4, le champs TermeFr aura comme chaîne à indexer "Ecorce cérébelleuse

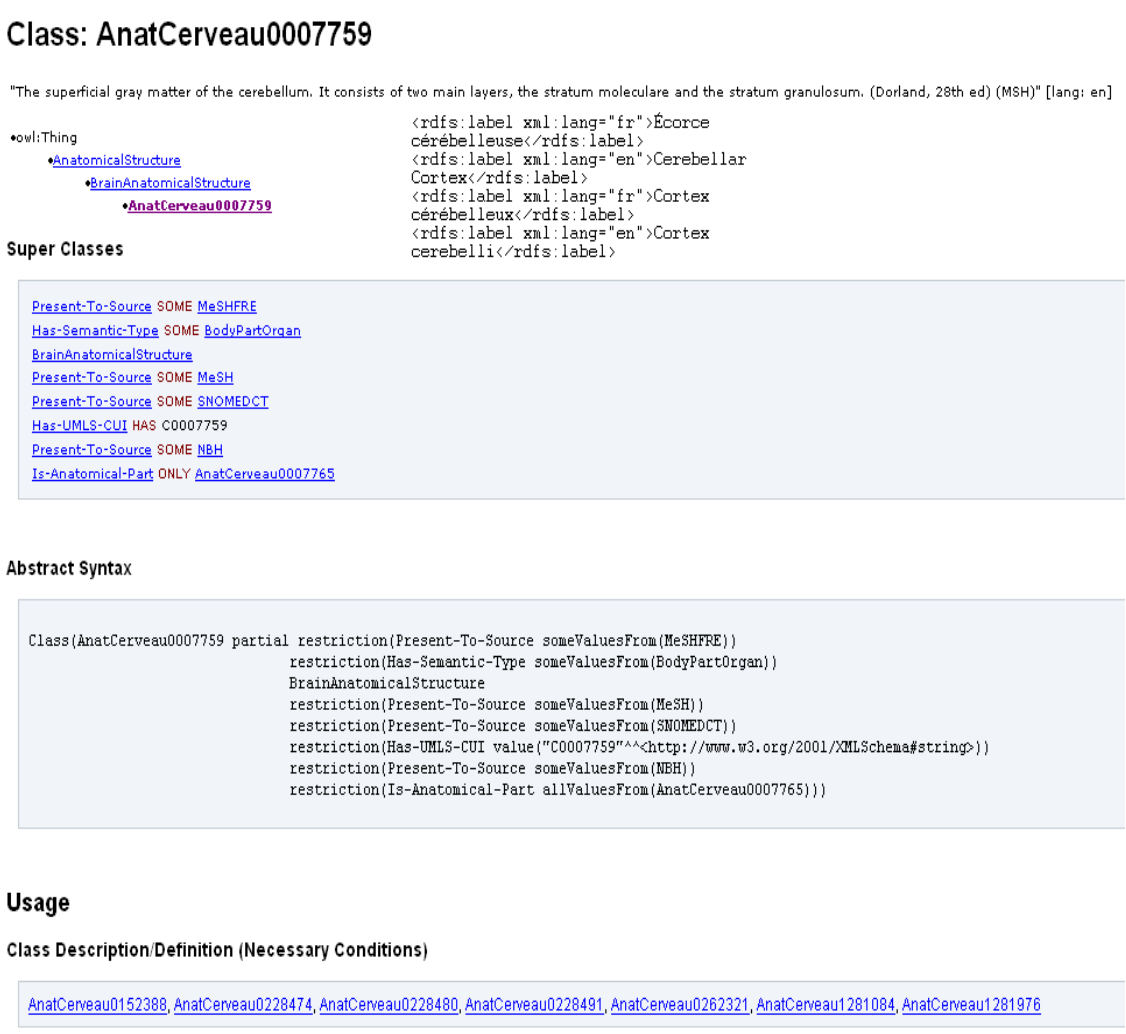


FIG. 7.4 – Exemple de définition d'une classe OWL de l'ontologie anatomique

Cortex cérébelleux" avant désuffixation.

Listing 7.1 – Pseudo-algorithme d'indexation d'un concept (classe OWL)

```
addConcept2Index(Concept C,Index I,boolean full,boolean parent)
        relation = ""; attribut = ""; TermEn = ""; TermFr = "";
        DocumentIndex d = new Document(); //le document à ajouter
        foreach R as relation in C.getrelation(parent) do
                relation += R.getlocalname();
                // recuperer le nom local de la relation
        fin
        foreach A as attribut in C.listattribut(parent) do
                attribut += A.getlocalname();
                 // recuperer le nom local de l'attribut
        fin
        /* Traitement des termes Français */
```

```
        tfr = c.listterm(fr);
        foreach terme in tfr do
                TermeFr += " " + terme;
                // constituer une chaîne de tous les termes français
        fin
                /* Traitement des termes Anglais */
        ten = c.listterm(en);
        foreach terme in ten do
                TermeEn += " " + terme;
                // constituer une chaîne de tous les termes anglais
        fin

        ChampsFr = new Champs("TermFr", TermFr);
        ChampsEn = new Champs("TermEn", TermEn);
        /* Chaque champs est  desufixé suivant la langue */

        if (full) // indexation de tous les éléments du concepts
                ChampsAttribut=
                        new champs("Attribut", attribut);
                ChampsRelation =
                        new champs(Relation", relation);

        finIf

        ChampsURIConcept = new champs("URIConcept", C.geturi());
        d.addchamps(); // ajouter tous les champs dans le document
        I.add(d); // ajouter enfin le document dans l'index
Fin
```

7.7 Interrogation de la base d'ontologies

Les algorithmes d'interrogation tiennent compte des différents niveaux d'indexation offerts et du niveau de connaissances de l'utilisateur sur les ontologies gérées par la base d'ontologies. L'interrogation peut se faire de deux manières : i) une interrogation simple basée sur une liste de mots-clés en spécifiant éventuellement un filtre basé sur la langue de recherche ; ii) une interrogation complexe qui permet d'une part de spécifier explicitement les champs sur lesquels la recherche doit s'effectuer (en les combinant par des opérateurs booléens), et d'autre part de limiter la portée de la recherche dans une langue donnée. La disponibilité de ces deux niveaux d'interrogation a un double avantage :

1. permettre à un utilisateur néophyte, qui ne sait pas exactement quel est le contenu de la base d'ontologies à interroger, d'effectuer une interrogation à la Google ;
2. permettre à un utilisateur expérimenté, qui a une certaine connaissance du contenu de la base d'ontologies, de spécifier une requête de façon plus élaborée. Ce mode est utile lorsqu'il s'agit de considérer que l'élément de la requête est un concept dont il faut chercher les concepts proches dans la base (cas de recherche de similarité entre concepts d'ontologies).

7.7.1 Interrogation en mode simple

L'interrogation en mode simple permet de faire des interrogations sans se soucier des champs sur lesquels va porter la recherche. Par défaut, le système reformule la requête de l'utilisateur en la propageant sur les différents champs de recherche disponibles.

7.7.2 Interrogation en mode complexe

Ce mode d'interrogation est conçu pour une utilisation avec des connaissances a priori sur le contenu de la base d'ontologies. On spécifie de façon explicite les champs de recherche (termes, attributs , relations).

ServO est disponible sous la forme d'une API Java, réutilisable dans différents contextes (dans une application locale, application web, ...). Nous avons implémenté un prototype sous la forme d'une application WEB[54] déployable sur un serveur Tomcat d'Apache[55], pour d'une part fournir une utilisation en ligne de certaines fonctionnalités de ServO et d'autre part pouvoir effectuer une recherche sur la représentation fournie par le RSD (voir chapitre 6) combinant les termes simples et les concepts. Comme ServO offre la possibilité d'obtenir les concepts d'ontologies à partir de termes en langue naturelle, nous avons fourni un service supplémentaire permettant d'interroger la base en ligne de litérature biomédicale MedLine[56].

7.8 Description du prototype

7.8.1 Architecture

L'architecture suit le motif de conception MVC (Modèle-Vue-Contrôleur) [Reenskaug *et al.*, 1996] qui sépare le modèle de données, l'interface utilisateur et la logique de contrôle (voir figure 7.5). Le modèle, qui gère le comportement de l'application, gère la base d'ontologies et se charge de l'accès aux données à travers les classes implémentées avec ServO. La vue correspond à l'interface avec laquelle l'utilisateur interagit. Elle est représentée par un ensemble de vues qui se chargent de l'affichage des interfaces d'interrogation et des résultats fournis par le modèle. Le contrôleur, représentée par une servlet Java, prend en charge la gestion des événements de synchronisation pour mettre à jour la vue ou le modèle. Il se charge de répartir les différentes requêtes qui arrivent.

La base d'ontologies gérée actuellement est constituée des ontologies anatomiques et fonctionnelles sur le cerveau, d'une classification orientée patient sur le cancer du sein [Messai *et al.*, 2006] et de l'ontologie NOESIS du domaine cardiovasculaire. Toutes ces ressources ont des concepts dénotés par des termes multilingues. Les versions indexées actuellement de ces ressources totalisent 2.832 concepts et 19.293 termes. Leur indexation a pris 1.23 minutes (utilisation d'un PC Pentium 4 avec 512 MO de RAM).

7.8.2 Interrogation de la base d'ontologies

Interface d'interrogation simple

Cette interface permet d'interroger la base en donnant une suite de mots-clés à chercher soit dans les termes, soit dans les attributs, soit dans les relations (figure 7.6). L'utilisateur a la

[54] accessible à l'adresse https ://kafka.imag.fr/servonto/servo
[55] http ://tomcat.apache.org/
[56] Accessible à travers PubMed à l'adresse `www.ncbi.nlm.nih.gov/entrez/`

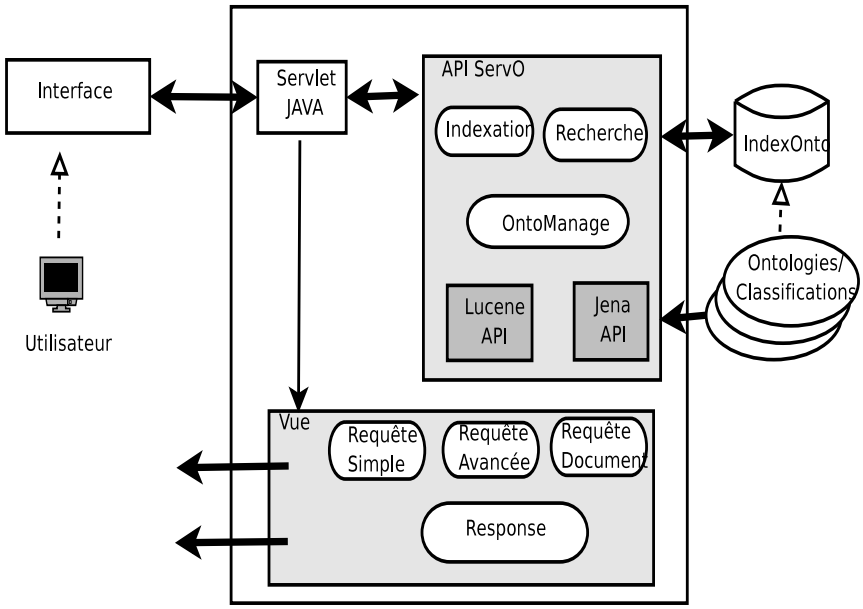


FIG. 7.5 – Architecture du prototype

possibilité de choisir l'option "Ontologie" afin d'effectuer une recherche dont le résultat est une liste triée d'ontologies. Les termes entrés peuvent être liés entre eux par des opérateurs booléens. Un exemple de requête peut être "mémoire sémantique OR parietal lobe".

FIG. 7.6 – Interface d'accueil pour l'interrogation simple

Interface d'interrogation avancée

Cette interface implémente la recherche en mode avancé offerte par ServO (voir 7.7). La possibilité est donnée de

1. limiter la recherche à une langue donnée (cas des ontologies ayant une terminologie multilingue) ;

2. spécifier un degré de similarité *sim*. Tous les résultats en-dessous de cette valeur ne sont pas présentés ;
3. spécifier de façon explicite les mots-clés de chaque champ.

Notons que pour les deux interrogations, simple et avancée, il est possible d'imposer la présence de certains termes dans les résultats ou d'augmenter le poids de certains termes dans la requête.

FIG. 7.7 – Interface d'interrogation avancée de ServO

Affichage des résultats

Les résultats sont affichés suivant leur degré de similarité avec la requête de l'utilisateur (voir 7.8). L'annexe C fournit un exemple de résultats obtenus avec ServO.

Une description détaillée de chaque ontologie ou concept résultat est fournie grâce à l'utilisation de l'API OWLDoc[57] disponible comme plugin dans Protégé. Cette API permet de générer une documentation d'une ontologie décrite en RDF/OWL.

7.8.3 Interrogation de la base en ligne MedLine

PubMed[58] est un service de la National Library of Medicine (NLM) qui inclut plus de 16 millions de citations de MedLine et des articles de journaux dans le domaine biomédical. MEDLINE est une des plus importantes sources de références bibliographiques pour les biologistes, les médecins et autres professionels de santé. Bien que l'interface d'accès à la base MEDLINE fournie à travers PubMed est relativement facile d'utilisation, effectuer des recherches sur la base en utilisant une langue autre que l'anglais produit des résultats médiocres. En effet, les ressources MedLine sont indexées par le vocabulaire fourni par le thésaurus MeSH (anglais). Pour remédier à cette insuffisance, CisMef[59], portail francophone de catalogage et d'indexation de ressources médicales francophones, propose dans son interface un accès aux ressources PubMed par traduction des termes français saisis par l'utilisateur.

[57] http ://www.co-ode.org/downloads/owldoc/ (consulté le 5 Novembre 2006)
[58] www.ncbi.nlm.nih.gov/entrez
[59] www.chu-rouen.fr/cismef/

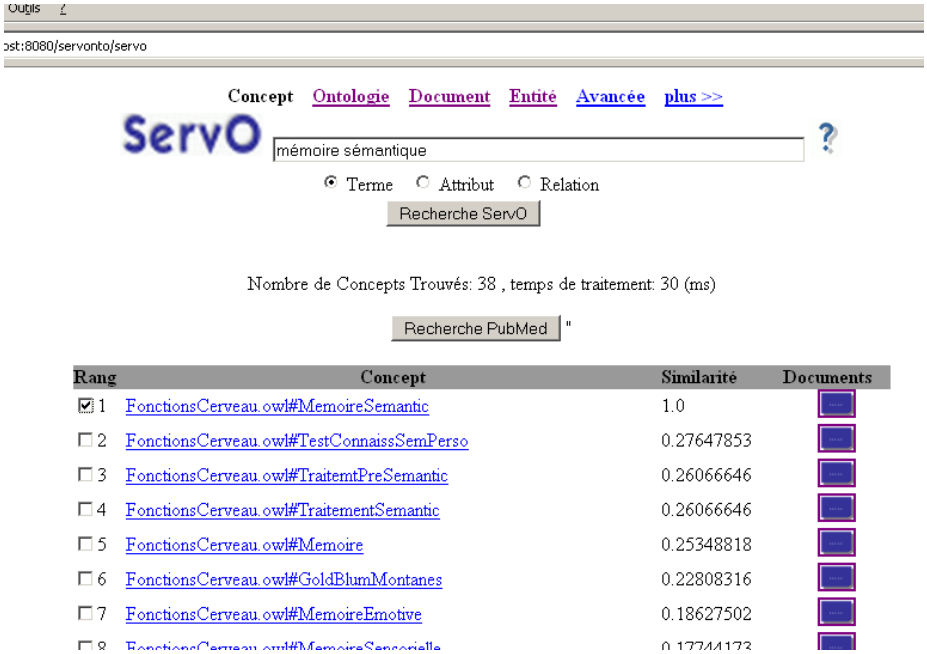


FIG. 7.8 – Affichage trié des résultats

Grâce au lien entre termes et concepts que ServO peut effectuer, nous proposons une interrogation de MEDLINE basée sur les termes en anglais des concepts jugés les plus pertinents (par l'utilisateur). On utilise le service Web Eutils fourni par le NCBI[60].

7.8.4 Interrogation combinant termes simples et concepts

L'utilisation de ServO nous a permis de proposer une interrogation d'une collection de documents caractérisée par une ou plusieurs ontologies. Pour cela, l'utilisateur peut choisir un degré de similarité minimale. La requête de l'utilisateur est alors envoyée sur la base d'ontologies. La liste de concepts trouvés est utilisée pour la reformulation de la requête de l'utilisateur à envoyer dans la base de documents. Par défaut, le degré de similarité est fixé à 1 (degré maximal). Cette technique permet de répondre aux problèmes de couverture des ontologies par rapport à la collection qu'elles caractérisent. Les premiers tests que nous avons faits en utilisant les bilans neurospsychologiques ont montré des résultats encourageants, mais la base de tests est trop petite pour tirer des conclusions significatives.

7.9 Conclusion

Nous avons présenté dans ce chapitre notre approche de gestion conjointe de plusieurs ontologies utilisant des techniques de la recherche d'information et des technologies du Web Sémantique. Cette approche, que nous avons appelée ServO pour Serveur d'Ontologies, permet à la fois la recherche d'ontologies et la recherche d'éléments d'ontologies. L'indexation d'une ontologie peut se faire soit en mode strict (description exacte des concepts) soit en mode étendu en utilisant les inférences possibles sur l'ontologie afin de recupérer les propriétés héritées des concepts.

[60] http ://www.ncbi.nlm.nih.gov/ (consulté le 5 Novembre 2006)

ServO est actuellement utilisé pour la gestion d'ontologies dans le domaine du cerveau, du cancer du sein et de la cardiologie, à travers le prototype Web implémenté. Nous avons montré qu'il est possible d'utiliser ServO pour l'interrogation de la base de littérature médicale en ligne MEDLINE.

Des extensions restent toutefois à faire. L'implémentation actuelle de ServO ne prend pas en compte les instances éventuellement présentes dans les ontologies. La gestion des instances de l'ontologie peut être utile dans un processus de recherche de similarités entre ontologies.

Lors d'une recherche, la version actuelle de ServO retourne une liste triée de résultats suivant le poids de chaque ontologie (ou élément d'ontologie) par rapport à une requête posée. Nous envisageons d'améliorer la présentation des résultats en fournissant par exemple en réponse à une requête une reconstruction sous forme hiérarchique de liste de concepts, issus d'une même ontologie, obtenue après une interrogation en texte libre.

Nous préconisons de rendre disponibles les fonctionnalités de recherche de ServO à travers des services Web, ce qui rendra les ontologies gérées par ServO accessibles par des applications distantes.

Conclusion

Les travaux présentés dans ce mémoire se situent dans le contexte général de la gestion unifiée de données hétérogènes, et plus particulièrement la gestion unifiée basée sur une approche de médiation. Les données manipulées au sein des organisations (bases de données, documents techniques, ...) sont gérées par divers systèmes d'informations (SI), conçus pour fonctionner de manière autonome. Il est essentiel, pour une prise de décision efficace, d'avoir une vision globale des différentes données de ces SI. Cependant, la diversité des données pose le problème de leur hétérogénéité, question centrale dans le cadre d'une gestion unifiée.

Nous avons proposé une solution à ce problème, basée sur l'utilisation d'ontologies. Les ontologies, comme spécification explicite d'une conceptualisation, offrent une solution possible à l'hétérogénéité en représentant explicitement et sémantiquement, aussi bien le modèle de données global que le modèle et le contenu des sources.

Les travaux sur la gestion unifiée s'occupent en général de l'intégration de sources de données ayant un modèle de données commun, aucune étape de prétraitement n'est alors à envisager. Cependant, quand il s'agit de prendre en compte de façon explicite des sources de données textuelles, il devient nécessaire d'effectuer une phase de prétraitement, qui consiste en une structuration de ces sources. L'avènement du Web sémantique et la prolifération d'ontologies et autres vocabulaires contrôlés sont un atout que nous avons exploité dans le cadre de notre travail.

Dans la section suivante, nous résumons les principaux apports de nos travaux.

1 Réalisations

1.1 Architecture de gestion unifiée avec prise en compte explicite de sources textuelles

Nous avons défini une architecture de gestion unifiée utilisant les technologies du Web Sémantique, avec la prise en compte explicite de sources textuelles. L'architecture est basée sur l'utilisation d'ontologies, d'une part pour la représentation du schéma global du système intégré et des sources de données, et d'autre part pour la description du contenu des sources textuelles. Grâce à la définition d'un schéma générique de représentation d'une source relationnelle, la représentation "ontologique" des sources à intégrer se fait de manière semi-automatique à l'aide des techniques de réingénierie.

1.2 Représentation de sources textuelles

Nous avons proposé une approche de caractérisation de sources textuelles combinant les termes simples, les concepts issus de plusieurs ontologies et les entités nommées éventuellement identifiées dans ces sources. Le résultat de la caractérisation est géré à travers une base de

données relationnelle, permettant une interrogation structurée, et l'export éventuel de l'index dans d'autres formats.

Les modules que nous avons développés à ce niveau seront utilisés dans le cadre du travail de thèse de Radja Messai, sur la mise en place d'un portail sémantique sur le cancer du sein destiné aux patients. Le portail permettra l'indexation et à la recherche sémantique de ressources relatives au cancer, à l'aide de l'ontologie construite à cet effet. Le module d'indexation et de recherche ainsi que le serveur d'ontologies seront les points d'entrée pour ce travail.

Une évaluation sur des corpus standards de la représentation combinant les termes simples et les concepts est envisagée et reste à faire.

1.3 Ontologies sur le cerveau

Dans le cadre de notre participation au projet BC^3 nous avons été amenés à formaliser la connaissance sur le cerveau au travers d'ontologies. Ces ontologies modélisent le cerveau sur ses aspects anatomique et fonctionnel. Elles sont disposibles en OWL, le langage recommandé par le W3C pour représenter des ontologies, et elles peuvent être réutilisées dans différents contextes.

1.4 Indexation et recherche d'ontologies

Le multiplicité des ontologies gérées dans le cadre de notre approche de gestion unifiée, et de façon générale, la prolifération de vocabulaires contrôlés et autres ontologies décrites dans les langages OWL/RDF, nous a amenés à proposer une approche de gestion conjointe de plusieurs ontologies. Cette approche s'inspire des travaux menés dans le domaine de la RI pour indexer et rechercher des ontologies ou des éléments d'ontologies.

Le serveur d'ontologies ServO permet une indexation rapide de vocabulaires contrôlés et d'ontologies. Il sert de support à l'interrogation dans le cadre de la gestion unifiée. Il offre des fonctionnalités étendues de recherche d'ontologies ou d'éléments d'ontologies. A ce titre, il joue le rôle d'assistant pour la sélection d'ontologies d'intérêt, dans le cadre d'une tâche de construction d'une nouvelle ontologie.

2 Perspectives

De nombreuses perspectives tant à caractère théorique que pratique peuvent être envisagées. Nous présentons dans cette section celles qui nous semblent les plus prometteuses.

2.1 Classification de comptes rendus

L'implémentation actuelle du module pour la reconnaissance d'entités nommées ne considère que les performances aux tests exprimées sous la forme d'une note brute. Il est souhaitable de généraliser l'identification des entités nommées afin notamment de pouvoir utiliser ces résultats pour la tâche de classification de comptes rendus médicaux qui servira de support à l'aide à la décision pour les praticiens.

2.2 Interrogation de bases relationnelles à travers les technologies du Web Sémantique

Le travail sur l'interrogation de bases relationnelles à travers un schéma OWL mérite d'être poursuivi. La traduction de requêtes posées sur le schéma OWL (VirtualSource.owl) vers le langage SQL permettra une interrogation d'une base de données relationnelle à travers les langages

du Web Sémantique. Nous nous focalisons sur le langage SPARQL [Perez *et al.*, 2006], qui a reçu le statut de "Recommendation Candidate" et qui sera très probablement le langage d'interrogation standard du Web Sémantique. Une fois ce travail finalisé et le module d'interrogation du système global implémenté, une évaluation globale du système d'intégration sera possible.

Un prolongement naturel de notre travail sur la gestion des bases de données et des documents textuels serait la prise en compte de données semi-structurées. Une étude sur l'intégration de données XML, basée sur le système de bases de données et de connaissances OSIRIS [Simonet et Simonet, 1995], est actuellement en cours dans le cadre du travail de thèse de Houda Ahmad. L'intégration des données XML nécessitera alors d'exprimer le méta-schéma d'OSIRIS en OWL.

2.3 Modélisation des connaissances sur le cerveau

Le travail sur la modélisation des connaissances sur le cerveau afin d'aboutir à terme à une ontologie anatomo-fonctionnelle mérite d'être poursuivi.

Dans un premier temps, nous envisageons d'enrichir le vocabulaire des ontologies construites, en particulier le vocabulaire français de l'ontologie anatomique du cerveau, qui servira de terminologie française à la NBH, non existante actuellement (en collaboration avec Pr. Douglas Bowden de l'Université de Washington). Par ailleurs, les ontologies ont été construites dans une optique de caractérisation de sources textuelles. Afin de pouvoir caractériser d'autres types de données non structurées telles que des images, nous envisageons d'étendre l'ontologie anatomique en considérant notamment des relations de nature topologique entre les structures anatomiques du cerveau. Pour l'ontologie cognitive, nous envisageons d'intégrer la sous-partie de la CIM-10 consacrée aux maladies neuropsychologiques.

Dans un second temps, il s'agira de formaliser une démarche d'intégration des ontologies anatomiques et fonctionnelles, à travers la découverte des liens anatomo-fonctionnels. Ces liens, valués, pourront être actualisés régulièrement en fonction de la découverte de nouvelles connaissances (apprentissage sur les comptes rendus médicaux, bases de données de patients).

2.4 Découverte et typage de relations entre ontologies

Il est envisagé d'améliorer les algorithmes implémentés dans le serveur d'ontologies afin de disposer, à terme, d'un environnement pour la decouverte et le typage de relations entre ontologies. Les résultats de ce travail pourront trouver plusieurs applications dans le domaine de l'ingénierie d'ontologies et la génération de correspondances entre schémas.

Avec la version actuelle du serveur d'ontologies, il n'est pas possible de savoir, parmi les mots-clés utilisés pour l'interrogation de la base d'ontologies, quels sont ceux qui n'ont pas participé au résultat, c'est-à-dire ceux qui ne sont pas présents dans la base. Pour l'interrogation d'un index basé sur la représentation combinant termes simples et concepts d'ontologies, tous les termes de la requête d'origine sont combinés avec les concepts pertinents retournés par le serveur en guise de re-formulation de requête. Il serait intéressant d'améliorer le serveur afin d'en faire un analyseur "intelligent" de requêtes.

Par ailleurs, afin de rendre disponibles à des applications distantes les ontologies gérées à travers le serveur d'ontologies, la mise à disposition, sous la forme de services Web, de certaines fonctionnalités de recherche du serveur est envisagée.

Troisième partie

Annexe

Annexe A

Ontologies relatives au cerveau

L'ensemble des ressources relatives à nos travaux est présenté à l'adresse `http://www-timc.imag.fr/Gayo.Diallo/Resources/`

A.1 Example de définition complète en OWL d'un concept de l'ontologie anatomique

Listing A.2 – AnatCerveau0007759 "Écorce cérébelleuse"

```
<owl:Class rdf:about="#AnatCerveau0007759">
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:allValuesFrom rdf:resource="#AnatCerveau0007765"/>
        <owl:onProperty>
          <owl:TransitiveProperty rdf:about="#Is-Anatomical-Part"/>
        </owl:onProperty>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:someValuesFrom>
          <owl:Class rdf:about="#SNOMEDCT"/>
        </owl:someValuesFrom>
        <owl:onProperty>
          <owl:ObjectProperty rdf:about="#Present-To-Source"/>
        </owl:onProperty>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:label xml:lang="fr">Écorce cérébelleuse</rdfs:label>
    <rdfs:label xml:lang="en">Cerebellar Cortex</rdfs:label>
    <rdfs:label xml:lang="fr">Cortex cérébelleux</rdfs:label>
    <rdfs:subClassOf rdf:resource="#BrainAnatomicalStructure"/>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:someValuesFrom rdf:resource="#NBH"/>
```

```
        <owl:onProperty>
          <owl:ObjectProperty rdf:about="#Present-To-Source"/>
        </owl:onProperty>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty>
          <owl:ObjectProperty rdf:about="#Present-To-Source"/>
        </owl:onProperty>
        <owl:someValuesFrom rdf:resource="#MeSHFRE"/>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty>
          <owl:ObjectProperty rdf:about="#Has-Semantic-Type"/>
        </owl:onProperty>
        <owl:someValuesFrom>
          <owl:Class rdf:about="#BodyPartOrgan"/>
        </owl:someValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:comment xml:lang="en">The superficial gray matter of
        the cerebellum. It consists of two main layers, the stratum
        moleculare and the stratum granulosum. (Dorland, 28th ed)
        (MSH)</rdfs:comment>
    <rdfs:label xml:lang="en">Cortex cerebelli</rdfs:label>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty>
          <owl:DatatypeProperty rdf:about="#Has-UMLS-CUI"/>
        </owl:onProperty>
        <owl:hasValue rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
        >C0007759</owl:hasValue>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:someValuesFrom rdf:resource="#MeSH"/>
        <owl:onProperty>
          <owl:ObjectProperty rdf:about="#Present-To-Source"/>
        </owl:onProperty>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>
```

A.2 Ontologie des fonctions cognitives

La hiérarchie de l'ontologie des fonctions cognitives est visualisée à l'aide du plugin Jambalaya de Protégé (voir figure A.1).

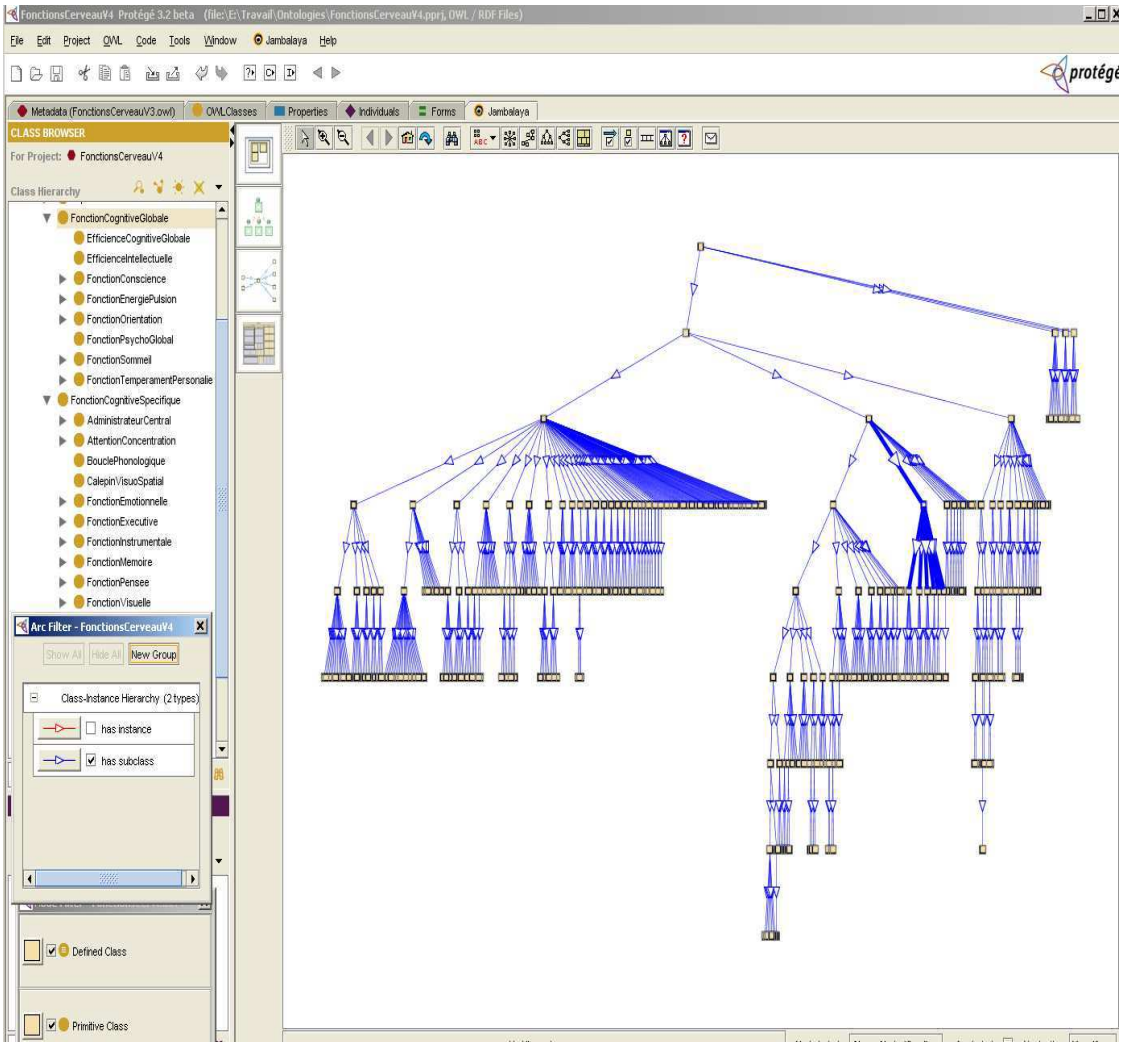


FIG. A.1 – Hiérarchie des classes OWL de l'ontologie des fonctions cognitives (obtenue à l'aide du plugin Jambalaya de Protégé)

A.3 Ontologie pour les tests cognitifs

La hiérarchie de l'ontologie est visualisée à l'aide du plugin Jambalaya de Protégé (voir figure A.2).

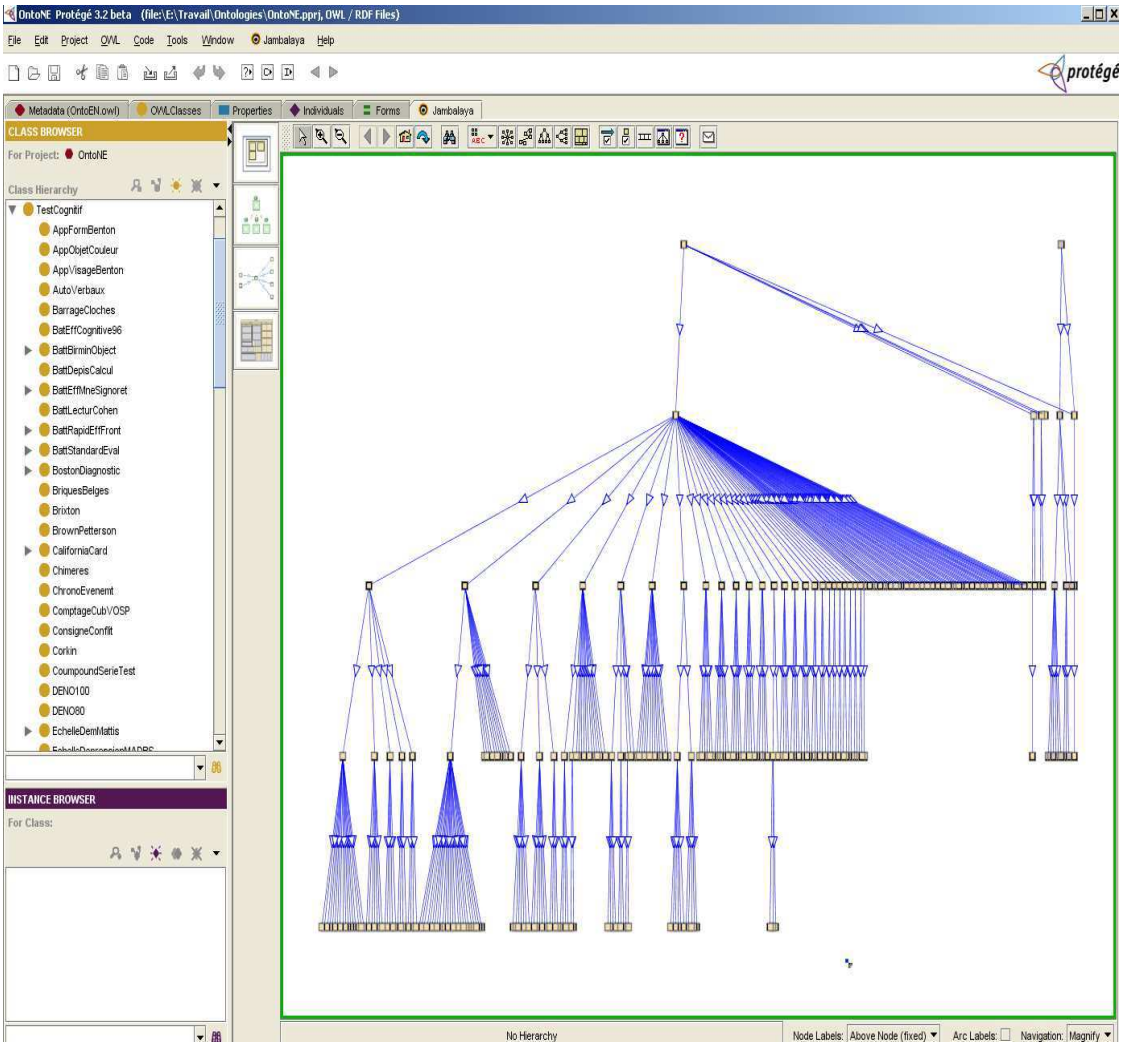


FIG. A.2 – Hiérarchie des classes OWL de l'ontologie pour le tests cognitifs (obtenue à l'aide du plugin Jambalaya de Protégé)

Annexe B

Schéma du RSD et exemple de résultats d'indexation conceptuelle

B.1 Schéma relationnel de la représentation sémantique de documents instanciée pour chaque source textuelle

Le RSD peut être stocké dans une base PostgreSQL ou une base MySQL. Nous présentons ci-dessous le code SQL de création de la base pour PostgreSQL (les contraintes n'y figurent pas).

Listing B.3 – RSD sous PostgreSQL

```
--
-- PostgreSQL database dump
--
-- Started on 2006-11-19 17:46:09 Paris, Madrid

SET client_encoding = 'UTF8';
SET check_function_bodies = false;
SET client_min_messages = warning;

COMMENT ON SCHEMA public IS 'Standard public schema';

SET search_path = public, pg_catalog;

SET default_tablespace = '';

SET default_with_oids = false;

CREATE TABLE corpus (
    urlcorpus character varying(255) NOT NULL,
    corpusname character varying(50) NOT NULL,
    langue character varying(15),
    modstem character varying(5),
    corpusnbdoc integer,
    corpusencod character varying(20)
);
```

```
CREATE TABLE document (
    docid integer NOT NULL,
    dctitle character varying(150) DEFAULT 'sans titre'::character varying,
    urllocal character varying(250) NOT NULL,
    urlweb character varying(250),
    datemaj date DEFAULT ('now'::text)::date,
    docweight character varying(8) DEFAULT 0,
    dccreator character varying(150),
    dcsubject character varying(250),
    dcdescription character varying(250),
    dcpublisher character varying(40),
    dccontributor character varying(40),
    dcdate date,
    dctype character varying(40),
    dcformat character varying(40),
    dcidentifier character varying(150),
    dcsource character varying(150),
    dclanguage character varying(20),
    dcrelation character varying(150),
    dccoverage character varying(250),
    dcrights character varying(250),
    urlcorpus character varying(250)
);

CREATE TABLE freqconcept (
    conceptid character varying(60) NOT NULL,
    prefterm character varying(150),
    docid integer NOT NULL,
    freqc integer DEFAULT 0 NOT NULL,
    poidsc numeric(5,4) DEFAULT 0
);

CREATE TABLE freqterme (
    termeid character varying(150) NOT NULL,
    docid integer NOT NULL,
    freqt integer DEFAULT 0 NOT NULL,
    poidst numeric(5,4) DEFAULT 0
);

CREATE TABLE frequence (
    nomterme character varying(40) NOT NULL,
    docid integer NOT NULL,
    freq integer DEFAULT 0 NOT NULL,
    poids numeric(5,4) DEFAULT 0
);

CREATE TABLE nedoc (
```

```
    namedentity character varying(60) NOT NULL,
    docid integer NOT NULL,
    valentity smallint NOT NULL,
    ontoid smallint
);

CREATE TABLE ontocorpus (
    ontoid smallint NOT NULL,
    corpusurl character varying(250) NOT NULL
);

CREATE TABLE ontology (
    ontoid smallint NOT NULL,
    urlontontoweb character varying(250),
    urlontolocal character varying(250),
    relbase character varying(15),
    ontoname character varying
);

CREATE TABLE posconcept (
    conceptid character varying(40) NOT NULL,
    docid integer NOT NULL,
    posc integer DEFAULT -1 NOT NULL
);

CREATE TABLE positionmot (
    nomterme character varying(40) NOT NULL,
    docid integer NOT NULL,
    pos integer DEFAULT -1 NOT NULL
);

CREATE TABLE posterm (
    termeid character varying(150) NOT NULL,
    docid integer NOT NULL,
    post integer DEFAULT -1 NOT NULL
);

CREATE TABLE term (
    nomterme character varying(40) NOT NULL,
    field character varying(20) NOT NULL,
    freq integer DEFAULT 0 NOT NULL
);

REVOKE ALL ON SCHEMA public FROM PUBLIC;
REVOKE ALL ON SCHEMA public FROM postgres;
GRANT ALL ON SCHEMA public TO postgres;
GRANT ALL ON SCHEMA public TO PUBLIC;
```

```
-- Completed on 2006-11-19 17:46:10 Paris, Madrid
-- PostgreSQL database dump complete
--
```

B.2 Exemple de résultats dans le RSD

La figure B.1 montre un extrait du résultat de l'indexation conceptuelle du corpus Cardio avec l'ontologie NOESIS sur la cardiologie. La colonne freqc indique le nombre d'occurrences du concept dans le document et poidsc représente la mesure cf.idf, indication de l'importance du concept dans le document, rapportée à l'ensemble des documents.

←T→	conceptid	prefterm	docid	freqc	poidsc
	M0011293	Infarct	14	25	0.287
	M0029049	Three dimensional ultrasound imaging of heart	14	22	0.260
	M0029050	Echocardiography Four Dimensional	14	22	0.260
	M0006970	Echocardiography Two Dimensional	14	22	0.244
	M0374430	Stress Echocardiographies Dobutamine	14	9	0.199
	M0014344	Heart Muscles	14	57	0.142
	M0026024	Endoluminal Repair	14	19	0.135
	M0011734	Ischemia (disorder)	14	18	0.129
	M0014340	Myocardial Infarcts	14	39	0.111
	M0009975	Ventricle of heart	14	16	0.093
	M0005049	Constriction Pathological	14	9	0.077
	M0001187	Arteriographies	14	9	0.072
	M0024875	Ventricular Functions	14	5	0.067
	M0000495	Adrenergic beta Blockers	14	4	0.061
	M0023774	Reperfusion Therapy	14	4	0.061
	M0024876	Ventricular Functions Left	14	4	0.060
	M0001560	Aortocoronary Bypass	14	6	0.057
	M0005191	Heart Disease Coronary	14	31	0.049
	M0027729	Dysfunction Left Ventricular	14	3	0.049
	M0001180	ANGINAL SYNDROME	14	7	0.047
	M0028090	Dysfunction Ventricular	14	3	0.047
	M0005197	Coronary Vessels	14	4	0.045
	M0008434	fibrinolytic agent	14	3	0.044
	M0001182	Angina at Rest	14	4	0.039
	M0008002	Step Tests	14	2	0.036
	M0001731	Artery NOS	14	17	0.031

FIG. B.1 – Exemple de résultats du RSD

B.3 Schéma générique de représentation d'une base de données relationnelle

Le schéma OWL est accessible à l'adresse `http://www-timc.imag.fr/Gayo.Diallo/Resources/Ontologies/VirtualSource/`

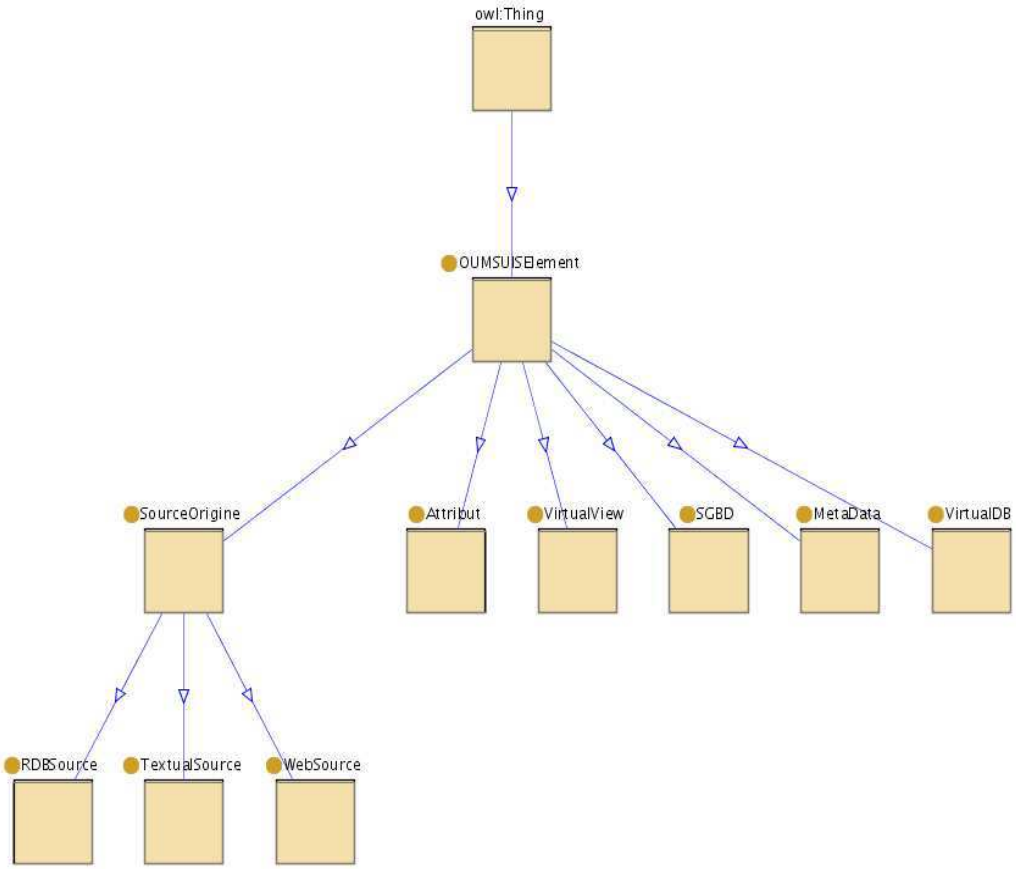


FIG. B.2 – Hiérarchie des classes de VirtualSource.owl (obtenue à l'aide du plugin Jambalaya de Protégé)

Annexe C

Requêtes sur le serveur d'ontologies

C.1 Requêtes sur le serveur d'ontologies

Comparaison du résultat de deux requêtes sur le serveur :

1. Requête 1 : trouver tous les concepts avec les termes "mémoire épisodique à long terme"
2. Requête 2 : trouver tous les concepts avec les termes "mémoire +épisodique à long terme" avec obligatoirement la présence du mot "épisodique" (+)

C.1.1 Requête 1

La requête 1 est traitée en 79 ms et le serveur donne une liste triée de 40 concepts

C.1.2 Requête 2

La requête 2 est traitée en 13 ms et le serveur donne un (1) seul concept en résultat "MemoireEpisodique"

FIG. C.1 – Requête 1

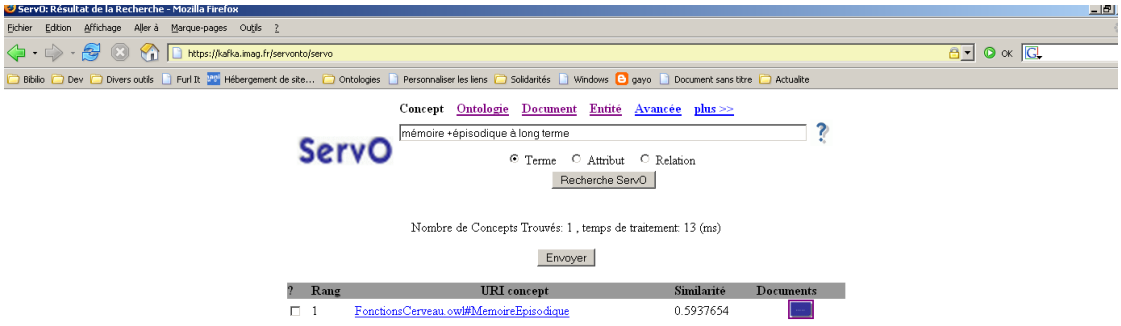


FIG. C.2 – Requête 2

Bibliographie

[Abiteboul *et al.*, 2002a] Abiteboul, S., Benjelloun, O. et Milo, T. (2002a). Web services and data integration. *In* Ling, T. W., Dayal, U., Bertino, E., Ng, W. K. et Goh, A., éditeurs : *International Conference on Web Information Systems Engineering*, pages 3–6. IEEE Computer Society.

[Abiteboul *et al.*, 2002b] Abiteboul, S., Cluet, S., Ferran, G. et Rousset, M.-C. (2002b). The xyleme project. *Computer Networks*, 39(3):225–238.

[Abrial, 1974] Abrial, J. R. (1974). Data semantics. *J. W. Klimbie and K. L. Koeman, editors, Data Base Management, North-Holland Publ. Co., Amsterdam*, pages 1–59.

[An *et al.*, 2006] An, Y., Mylopoulos, J. et Borgida, A. (2006). Building semantic mappings from databases to ontologies. *In Proceedings, The Twenty-First National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference, July 16-20, 2006, Boston, Massachusetts, USA*.

[Arenas *et al.*, 2003] Arenas, M., Kantere, V., Kementsietsidis, A., Kiringa, I., Miller, R. et Mylopoulos, J. (2003). The hyperion project : From data integration to data coordination. *SIGMOD Record*, 32(3).

[Arens *et al.*, 1994] Arens, Y., Chee, C. Y., Hsu, C.-N., , In, H. et Knoblock, C. A. (1994). Query processing in an information mediator. *In Proceedings of the ARPA/Rome Laboratory Knowledge-Based Planning and Scheduling Initiative Workshop, Tucson, AZ*.

[Arens *et al.*, 1997] Arens, Y., Hsu, C.-N. et Knoblock, C. A. (1997). Query processing in the SIMS information mediator. *In* Huhns, M. N. et Singh, M. P., éditeurs : *Readings in Agents*, pages 82–90. Morgan Kaufmann, San Francisco, CA, USA.

[Aussenac-Gilles *et al.*, 2000] Aussenac-Gilles, N., Biebow, B. et Szulman, S. (2000). Revisiting ontology design : A methodology based on corpus analysis. *In 12th European Workshop on Knowledge Acquisition, Modeling and Management*, pages 172–188.

[Bachimont, 2000] Bachimont, B. (2000). Engagement sémantique et engagement ontologique : conception et réalisation d'ontologies en ingénierie des connaissances. *Ingénierie des connaissances, Evolutions récentes et nouveaux défis*.

[Barillot *et al.*, 2003] Barillot, C., Amsaleg, L., Aubry, F., Bazin, J., Benali, H., y. Cointepas, Corouge, I., Dameron, O., Dojat, M., Garbary, C., Gibaud, B., Gros, P., Kinkingnehum, S., Malandain, G., Matsumoto, J., Papadopoulos, D., Pelegrini-Isaac, M., Richard, N. et Simon, E. (2003). Neurobase : Management of distributed knowledge abd data bases in neuroimaging. *Neuroimage*, 19(2):726.

[Barish et Knoblock, 2003] Barish, G. et Knoblock, C. A. (2003). Learning value predictors for the speculative execution of information gathering plans. pages 3–9.

[Batrancourt, 2003] Batrancourt, B. (2003). *Modélisation et outils logiciels pour une connaissance anatomo-fonctionnelle du cerveau humain*. Thèse de doctorat, Université Lyon 1.

[Bayardo, Jr. *et al.*, 1997] Bayardo, Jr., R. J., Bohrer, W., Brice, R., Cichocki, A., Fowler, J., Helal, A., Kashyap, V., Ksiezyk, T., Martin, G., Nodine, M., Rashid, M., Rusinkiewicz, M., Shea, R., Unnikrishnan, C., Unruh, A. et Woelk, D. (1997). InfoSleuth : Agent-based semantic integration of information in open and dynamic environments. 26,2:195–206.

[Baziz *et al.*, 2005] Baziz, M., Boughanem, M. et Aussenac-Gilles, N. (2005). A Conceptual Indexing Approach for the TREC Robust Task . *In* Voorhees, E. M. et Buckland, L. P., éditeurs : *The Fourteenth Text REtrieval Conference Proceedings (TREC 2005) , Gaithersburg, Maryland, 15/11/05-18/11/05*. NIST. Pages de la publication : cdrom.

[Bechhofer *et al.*, 2001] Bechhofer, S., Horrocks, I., Goble, C. et Stevens, R. (2001). OilEd : A reason-able ontology editor for the semantic Web. *Lecture Notes in Computer Science*, 2174:396– ? ?

[Belkin et Croft, 1992] Belkin, N. J. et Croft, W. B. (1992). Information filtering and information retrieval : Two sides of the same coin ? *Commun. ACM*, 35(12):29–38.

[Beneventano *et al.*, 2001] Beneventano, D., Bergamaschi, S., Guerra, F. et Vincini, M. (2001). The momis approach to information integration. pages 194–198.

[Benjamins et Fensel, 1998] Benjamins, R. et Fensel, D. (1998). The ontological engineering initiative-ka. *Proceedings of the 1st International Conference on Formal Ontologies in Information Systems, FOIS'98, Trento, Italy*, pages 287–301.

[Benjelloun *et al.*, 2005] Benjelloun, O., Garcia-Molina, H., Jonas, J., Su, Q. et Widom, J. (2005). Swoosh : A generic approach to entity resolution. Rapport technique, Stanford University.

[Bernaras *et al.*, 1996] Bernaras, A., Laresgoiti, I. et Corera, J. (1996). Building and reusing ontologies for electrical network applications. *Proceedings of the 12th European Conference on Artificial Intelligence (ECAI)*, pages 298–302.

[Berners-Lee *et al.*, 2000] Berners-Lee, T., Hendler, J. et Lassila, O. (2000). Semantic web. *Scientific America*, 1(1):68–88.

[Bernstein *et al.*, 2002] Bernstein, P., Giunchiglia, F., Kementsietsidis, A., Mylopoulos, J., Serafini, L. et Zaihrayeu, I. (2002). Data management for peer-to-peer computing : A vision. *Workshop on the Web and Databases, WebDB 2002*.

[Biebow et Szulman, 1999] Biebow, B. et Szulman, S. (1999). Terminae : A linguistic-based tool for the building of a domain ontology. *11th European Workshop, Knowledge Acquisition, Modeling and Management (EKAW' 99), Dagstuhl Castle, Germany*, pages 49–66.

[Bizer, 2003] Bizer, C. (2003). D2r map-a database to rdf mapping languaage. *WWW2003, The Twelfth Onternational World Wide Web Conference, Budapest, Hugary*.

[Blair et Maron, 1985] Blair, D. et Maron, M. (1985). An evaluation of retrieval effectiveness for a full-text document-retrieval system. *Commun. of the ACM*, 28:289–299.

[Blazquez *et al.*, 1998] Blazquez, M., Fernandez, M., Garcia-Pinar, J. et Gomez-Perez, A. (1998). A. building ontologies at the knowledge level using the ontology design environment. *Knowledge Acquisition of Knowledge-Based Systems Workshop (KAW) 1998*.

[Bloehdorn *et al.*, 2005] Bloehdorn, S., Cimiano, P., Hotho, A. et Staab, S. (2005). An ontology-based framwork for text miningi. *In LDV Forum GLDV Journal for Computational Linguistics and Language Technology*, 20 (1):87–112.

[Bodenreider, 2004] Bodenreider, O. (2004). The unified medical language system (umls) : integrating biomedical terminology. 32(Database-Issue):267–270.

[Bontcheva, 2004] Bontcheva, K. (2004). Open-source tools for creation, maintenance, and storage of lexical resources for language generation from ontologies. *Fourth International Conference on Language Resources and Evaluation (LREC'2004). Lisbon, Portugal.*

[Borgida *et al.*, 1989] Borgida, A., Brachman, R. J., McGuinness, D. L. et Resnick, L. A. (1989). CLASSIC : a structural data model for objects. *In ACM SIGMOD International Conference on Management of Data, Portland Oregon, May-June*, pages 58–67.

[Bowden et Dubach, 2005] Bowden, D. et Dubach, M. (2005). Neuroanatomical nomenclature and ontology. *In Databasing the Brain.* John Wiley & Sons.

[Boyd *et al.*, 2002] Boyd, M., McBrien, P. et Tong, N. (2002). The automed schema integration repository. *In Proceedings of BNCOD02, Springer Verlag LNCS*, 2405:42–45.

[Brachman *et al.*, 1991] Brachman, R. J., McGuinness, D. L., Patel-Schneider, P. F., Resnick, L. A. et Borgida, A. (1991). *Principles of Semantic Networks*, chapitre Living with CLASSIC : How and When to Use a KL-ONElike Language, pages 401–456. Morgan Kaufmann, San Mateo, CA.

[Brodmann, 1909] Brodmann, K. (1909). Vergleichende lokalisationslehre der großhirnrinde in ihren prinzipien dargestellt auf grund des zellenbaues. Leipzig : Barth JA.

[Buckley *et al.*, 1995] Buckley, C., Salton, G., Allan, J. et Singhal, A. (1995). Automatic query expansion using smart : Trec-3. *D. K. Harman (ed.), Proceedings of the 3rd Text REtrieval Conference (TREC-3) (NIST SP 500-225).*

[Burgun et Bodenreider, 2001] Burgun, A. et Bodenreider, O. (2001). Comparing terms, concepts and semantic classes in wordnet and the unified medical language system. *Proceedings of the NAACL'2001 Workshop, "WordNet and Other Lexical Resources : Applications, Extensions and Customizations"*, pages 77–82.

[Calì *et al.*, 2004] Calì, A., Calvanese, D., Giacomo, G. D. et Lenzerini, M. (2004). Data integration under integrity constraints. *In Inf. Syst.*, volume 29, pages 147–163.

[Callan *et al.*, 1992] Callan, J. P., Croft, W. B. et Harding, S. M. (1992). The INQUERY retrieval system. pages 78–83.

[Carpineto et Romano, 1998] Carpineto, C. et Romano, G. (1998). Effective reformulation of boolean queries with concept lattices. pages 83–94.

[Carroll *et al.*, 2004] Carroll, J. J., Dickinson, I., Dollin, C., Reynolds, D., Seaborne, A. et Wilkinson, K. (2004). Jena : implementing the semantic web recommendations. *Alternate Track Papers & Posters)Proceedings of the 13th international conference on World Wide Web - Alternate Track Papers & Posters, WWW 2004, New York, NY, USA*, pages 74–83.

[Castano et Antonellis, 1999] Castano, S. et Antonellis, V. D. (1999). Artemis : Analysis and reconciliation tool environment for multiple information sources. *SEBD*, pages 341–356.

[Castano *et al.*, 2001] Castano, S., Antonellis, V. D. et di Vimercati, S. D. C. (2001). Global viewing of heterogeneous data sources. *IEEE Transactions on Knowledge and Data Engineering*, 13(2):277–297.

[Catarci et Lenzerini, 1993] Catarci, T. et Lenzerini, M. (1993). Representing and using interschema knowledge in cooperative information systems. *Journal of Intelligent and Cooperative Information Systems*, pages 55–62.

[Chandrasekaran *et al.*, 1998] Chandrasekaran, B., Josephson, J. et Benjamins, V. R. (1998). The ontology of tasks and methods. *Knowledge Acquisition Workshop, KAW98.*

[Chaudhri *et al.*, 1998] Chaudhri, V. K., Farquhar, A., Fikes, R., Karp, P. D. et Rice, J. (1998). Okbc : A programmatic foundation for knowledge base interoperability. pages 600–607.

[Chiaramella et Defude, 1987] Chiaramella, Y. et Defude, B. (1987). A prototype of an intelligent system for information retrieval : Iota. *Information Processing & Management*, 23(4):285–303.

[Chinchor, 1997] Chinchor, N. (1997). Muc-7 named entity task definition, dry run version, version 3.5. *Documentation for the Seven Message Understanding Conference.*

[Choi *et al.*, 2006] Choi, N., Song, I.-Y. et Han, H. (2006). A survey on ontology mapping. *SIGMOD Rec.*, 35(3):34–41.

[Cimiano et Voelker, 2005] Cimiano, P. et Voelker, J. (2005). Text2onto - a framework for ontology learning and data-driven change discovery. *Proceedings of the 10th International Conference on Applications of Natural Language to Information Systems (NLDB'2005).*

[Cleverdon, 1967] Cleverdon, C. (1967). The cranfield tests on index language devices. *Aslib Proceedings*, 19(6):173–193.

[Cleverdon *et al.*, 1966] Cleverdon, C., Mills, J. et Keen, E. (1966). Factors determining the performance of indexing systems. *Aslib-Cranfield Project, Cranfield*, 2 vols.

[Cody *et al.*, 1995] Cody, W. F., Haas, L. M., Niblack, W., Arya, M., Carey, M. J., Fagin, R., Flickner, M., Lee, D., Petkovic, D., Schwarz, P. M., II, J. T., Roth, M. T., Williams, J. H. et Wimmers, E. L. (1995). Querying multimedia data from multiple repositories by content : the garlic project. pages 17–35.

[Consortium, 2001] Consortium, G. O. (2001). Creating the gene ontology resource : design and implementation. *Genome Res*, 11(8):1425–33.

[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*, 21(1):20–27.

[Corcho *et al.*, 2003] Corcho, Ó., Fernández-López, M., Gómez-Pérez, A. et López-Cima, A. (2003). Building legal ontologies with methontology and webode. pages 142–157.

[Corcho et Perez, 2000] Corcho, O. et Perez, A. G. (2000). Evaluating knowledge representation and reasoning capabilities of ontology specification languages. *In Proc. of ECAI-00 Workshop on Applications of Ontologies and Problem-Solving Methods.*

[Cote *et al.*, 2006] Cote, R. G., Jones, P., Apweiler, R. et Hermjakob, H. (2006). The ontology lookup service, a lightweight cross-platform tool for controlled vocabulary queries. *BMC Bioinformatics*, 7(1).

[Croft, 1993] Croft, W. B. (1993). Knowledge-based and statistical approaches to text retrieval. *IEEE Expert : Intelligent Systems and Their Applications*, 08(2):8–12.

[Cunningham, 2005] Cunningham, H. (2005). Information Extraction, Automatic. *Encyclopedia of Language and Linguistics, 2nd Edition, Elsevier.*

[Dameron *et al.*, 2003] Dameron, O., Burgun, A., Morandi, X. et Gibaud, B. (2003). Ontologie stratifiée de l'anatomie du cortex cérébral : application au maintien de la cohérence. *In Journée Web Sémantique Médical 2003, Rennes, France.*

[de Laborda et Conrad, 2005] de Laborda, C. P. et Conrad, S. (2005). Relational.owl : a data and schema representation format based on owl. *In APCCM '05 : Proceedings of the 2nd Asia-Pacific conference on Conceptual modelling*, pages 89–96, Darlinghurst, Australia, Australia. Australian Computer Society, Inc.

[Decker *et al.*, 2000] Decker, S., Mitra, P. et Melnik, S. (2000). Framework for the semantic web : An rdf tutorial. 4(6):68–73.

[Deerwester *et al.*, 1990] Deerwester, S. C., Dumais, S. T., Landauer, T. K., Furnas, G. W. et Harshman, R. A. (1990). Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391–407.

[Diallo *et al.*, 2003] Diallo, G., Batrancourt, B., Bernhard, D. et Simonet, M. (2003). Vers une ontologie anatomo-fonctionnelle du cerveau. *In Journée Web Sémantic Médical WSM'03, Rennes France.*

[Diallo *et al.*, 2006a] Diallo, G., Simonet, M. et Simonet, A. (2006a). An approach to automatic ontology-based annotation of biomedical texts. *Advances in Applied Artificial Intelligence, 19th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2006, Annecy, France, June 27-30, 2006, Proceedings*, Lecture Notes in Artifical Intelligence, 4031, Springer:1024–1033.

[Diallo *et al.*, 2006b] Diallo, G., Simonet, M. et Simonet, A. (2006b). Bringing together structured and unstructured sources : the OUMSUIS approach. *In Proceedings of OTM Confederated International Workshops and Posters, Knowledge Systems in Bioinformatics (KSinBit'06) in conjunction with OTM'06, Montpellier, France, LNCS Springer-Verlag.*

[Dill *et al.*, 2003] Dill, S., Eiron, N., Gibson, D., Gruhl, D., Guha, R., Jhingran, A., Kanungo, T., Rajagopalan, S., Tomkins, A., Tomlin, J. A. et Zien, J. Y. (2003). Semtag and seeker : bootstrapping the semantic web via automated semantic annotation. pages 178–186.

[Ding *et al.*, 2004] Ding, L., Finin, T., Joshi, A., Pan, R., Cost, R. S., Peng, Y., Reddivari, P., Doshi, V. et Sachs, J. (2004). Swoogle : a search and metadata engine for the semantic web. pages 652–659.

[Dong et Li, 2002] Dong, Y. et Li, M. (2002). Tempplet : A new method for domain-specific ontology design. *EDCIS'2002*, pages 90–103.

[Elmagarmid *et al.*, 1999] Elmagarmid, A., Rusinkiewicz, M. et Sheth, A. (1999). Management of heterogeneous and autonomous database systems. *Morgan Kaufmann, San Francisco.*

[Euzenat, 1995] Euzenat, J. (1995). Building consensual knowledge bases : context and architecture. *Towards very large knowledge bases (actes 2nd international conference on building and sharing very large-scale knowledge bases (KBKS), Enschede (NL), (10-13 avril) 1995) IOS press, Amsterdam (NL)*, pages 143–155.

[Euzenat, 1996] Euzenat, J. (1996). Corporate memory through cooperative creation of knowledge bases and hyper-documents. *Actes 10th knowledge acquisition workshop, Banff (CA), 9-14 novembre*, pages 1–18.

[Farquhar *et al.*, 1995] Farquhar, A., Fikes, R., Pratt, W. et Rice, J. (1995). Collaborative ontology construction for information integration. *Stanford University.*

[Fellbaum, 1999] Fellbaum, C. (1999). *WordNet : An electronic lexical database.* Cambridge, Massashussets : MIT Press.

[Fellbaum *et al.*, 2006] Fellbaum, C., Hahn, U. et Smith, B. (2006). Towards new information resources for public health from wordnet to medical wordnet. *Journal of Biomedical Informatics*, 39(3):321–332.

[Fensel *et al.*, 1999] Fensel, D., Angele, J., Decker, S., Erdmann, M., Schnurr, H.-P., Staab, S., Studer, R. et Witt, A. (1999). On2broker : Semantic-based access to information sources at the www. *WebNet*, pages 366–371.

[Fensel *et al.*, 2001] Fensel, D., van Harmelen, F., Horrocks, I., McGuinness, D. L. et Patel-Schneider, P. F. (2001). Oil : An ontology infrastructure for the semantic web. *In IEEE Intelligent Systems*, volume 16, pages 38–45.

[Fernandez *et al.*, 1997] Fernandez, M., Gomez-Perez, A. et Juristo, N. (1997). Methontology : From ontological art towards ontological engineering. *Spring Symposium Series*, pages 33–40.

[Fortuna *et al.*, 2006] Fortuna, B., Grobelnik, M. et Mladenic, D. (2006). System for semi-automatic ontology construction. *The 3rd Annual European Semantic Web Conference (ESWC2006) Budva, Montenegro, LNCS 4011, Springer-Verlag.*

[Friedman *et al.*, 1999] Friedman, M., Levy, A. et Millstein, T. (1999). Navigational plans for data integration. *Proc. of the 16th National Conference on Artificial Intelligence AAAI*, pages 67–73.

[Fuhr, 1992] Fuhr, N. (1992). Probabilistic models in information retrieval. *Comput. J.*, 35(3): 243–255.

[Furnas *et al.*, 1988] Furnas, G., Deerwester, S., Dumais, S., Landauer, T., Harshman, R., Streeter, L. et Lochbaum, K. (1988). Information retrieval using a singular value decomposition model of latent semantic structure. *The 11th International Conference on Research and Development in Information Retrieval , Grenoble, France : ACM Press*, pages 65–480.

[Gabrilovich et Markovitch, 2005] Gabrilovich, E. et Markovitch, S. (2005). Feature generation for text categorization using world knowledge. *Proceedings of the 19th International joint Conference in Artificial Intelligence*, pages 1048–1053.

[Gamper *et al.*, 1999] Gamper, J., Nejdl, W. et Wolpers, M. (1999). Combining ontologies and terminologies in information systems. *Proc. 5th International Congress on Terminology and Knowledge Engineering, Innsbruck, Austria.*

[Garcia-Molina *et al.*, 1997] Garcia-Molina, H., Papakonstantinou, Y., Quass, D., Rajaraman, A., Sagiv, Y., Ullman, J. et Widom, J. (1997). The tsimmis approach to mediation : Data models and languages. *Journal of Intelligent Information Systems*, 8:117–132.

[Gedzelman *et al.*, 2005] Gedzelman, S., Simonet, M., Bernhard, D., Diallo, G. et Palmer, P. (2005). Building an ontology of cardio-vascular diseases for concept-based information retrieval. *IEEE Proceedings of Computers in Cardiology*, 32:255–258.

[Genesereth et Fikes, 1992] Genesereth, M. et Fikes, R. (1992). Knowledge interchange format - version 3 - reference manual. Rapport technique, Stanford University, Logic Group, Logic-92-1, Stanford, CA.

[Genesereth *et al.*, 1997] Genesereth, M. R., Keller, A. M. et Duschka, O. M. (1997). Infomaster : An information integration system. pages 539–542.

[Gennari *et al.*, 2003] Gennari, J., Musen, M. A., Fergerson, R. W., Grosso, W. E., Crubezy, M., Eriksson, H., Noy, N. F., et Tu, S. W. (2003). The evolution of prot ?g ? : An environment for knowledge-based systems development. *International Journal of Human-Computer Studies*, 58 :1:89–123.

[Goasdoué *et al.*, 2000] Goasdoué, F., Lattès, V. et Rousset, M.-C. (2000). The use of carin language and algorithms for information integration : The picsel system. *In Int. J. Cooperative Inf. Syst.*, volume 9, pages 383–401.

[Goh *et al.*, 1999] Goh, C. H., Bressan, S., Madnick, S. et Siegel, M. (1999). Context interchange : new features and formalisms for the intelligent integration of information. *ACM Trans. Inf. Syst.*, 17(3):270–293.

[Gomez-Perez et Rojas, 1999] Gomez-Perez, A. et Rojas, M. (1999). Ontological reengineering and reuse. *In European Knowledge Acquisition Workshop (EKAW)*.

[Gonzalo *et al.*, 1998] Gonzalo, J., Verdejo, F., Chugur, I. et Cigarran, J. (1998). Indexing with wordnet synsets can improve text retrieval. *Proceedings of the COLING/ACL '98 Workshop on Usage of WordNet for NLP*, pages 38–44.

[Gregory Karvounarakis, 2001] Gregory Karvounarakis, Vassilis Christophides, D. P. S. A. (2001). Querying rdf descriptions for community web portals. *17emes Journees, BDA'2001, Agadir, Maroc*, pages 133–144.

[Gruber, 1995] Gruber, T. (1995). Towards principles for the design of ontologies used for knowledge sharing. *International Journal of Human-Computer Studies*, 43 (5/6):907–028.

[Gruber, 1992] Gruber, T. R. (1992). Ontolingua : A mechanism to support portable ontologies. tech. report. Rapport technique, Stanford University, Knowledge Systems Laboratory.

[Gruber, 1993] Gruber, T. R. (1993). A translation approach to portable ontologies. *Knowledge Acquisition*, 5(2):199–220.

[Grüninger et Fox, 1995] Grüninger, M. et Fox, M. S. (1995). Methodology for the design and evaluation of ontologies. *IJCAI Workshop on Basic Ontological Issues in Knowledge Sharing, Montreal, Canada*.

[Guarino, 1998a] Guarino, N. (1998a). Formal ontologies and information systems. *In FOIS'98, Trento, Italy, IO Press*.

[Guarino, 1998b] Guarino, N. (1998b). Some ontological principles for designing upper level lexical resources. *CoRR*, cmp-lg/9809002.

[Guarino, 1999] Guarino, N. (1999). Ontoseek : Content-based acess to the web. *IEEE Intelligent Systems*, pages 70–80.

[Gupta *et al.*, 2000] Gupta, A., Ludascher, B. et Martone, M. E. (2000). Knowledge-based integration of neuroscience data sources. *In Statistical and Scientific Database Management*, pages 39–52.

[Haarslev et Möller, 2001] Haarslev, V. et Möller, R. (2001). RACER system description. *Lecture Notes in Computer Science*, 2083:701– ? ?

[Haas *et al.*, 2001] Haas, L. M., Schwarz, P. M., Kodali, P., Kotlar, E., Rice, J. E. et Swope, W. C. (2001). Discoverylink : A system for integrated access to life sciences data sources. *IBM Systems Journal*, 40(2).

[Halevy, 2001] Halevy, A. Y. (2001). Answering queries using views : A survey. *VLDB Journal : Very Large Data Bases*, 10(4):270–294.

[Halevy, 2005] Halevy, A. Y. (2005). Why your data won't mix. *ACM Queue*, 3(8):50–58.

[Harman, 1992] Harman, D. (1992). Overview of the first text retrieval conference (trec-1). pages 1–20.

[Heflin *et al.*, 2003] Heflin, J., Hendler, J. A. et Luke, S. (2003). Shoe : A blueprint for the semantic web. *In* Fensel, D., Hendler, J. A., Lieberman, H. et Wahlster, W., éditeurs : *Spinning the Semantic Web : Bringing the World Wide Web to Its Full Potential [outcome of a Dagstuhl seminar]*, pages 29–63. MIT Press.

[Hersh *et al.*, 1994] Hersh, W., Buckley, C., Leone, T. et Hickam, D. (1994). Ohsumed : An interactive retrieval evaluation and new large test collection for research. *Proceedings of the 17th Annual ACM SIGIR Conference*, pages 192–201.

[Hillman, 2001] Hillman, D. (2001). Using dublin core. *DCMI Recommendation. Disponible à http ://dublincore.org/documents/usageguide/.*

[Horrocks, 1998] Horrocks, I. (1998). The FaCT system. *Lecture Notes in Computer Science*, 1397:307– ? ?

[Horrocks et Sattler, 2003] Horrocks, I. et Sattler, U. (2003). Decidability of shiq with complex role inclusion axioms. *IJCAI'2003*, pages 343–348.

[Hurson et Bright, 1991] Hurson, A. R. et Bright, M. W. (1991). Multidatabase systems : An advanced concept in handling distributed data. *Advances in Computers*, 32:149–200.

[Ingwersen, 1992] Ingwersen, P. (1992). *Information Retrieval Interaction.* Taylor Graham Publishing.

[InterOp, 2006] InterOp, I. (2006). State of the art and state of the practice including initial possible research orientations, d8.1. Rapport technique, InterOp, Interoperability Research for Networked Enterprises Applications and Software.

[Ives *et al.*, 2004] Ives, Z. G., Halevy, A. Y., Mork, P. et Tatarinov, I. (2004). Piazza : mediation and integration infrastructure for semantic web data. *J. Web Sem.*, 1(2):155–175.

[Jarrar et Meersman, 2002] Jarrar, M. et Meersman, R. (2002). Formal ontology engineering in the dogma approach. *Proceedings of the International Conference on Ontologies, Databases and Applications of Semantics (ODBase 02)*, LNCS 2519:1238 – 1254.

[Jing et Croft, 1994] Jing, Y. et Croft, W. B. (1994). An association thesaurus for information retrieval. *Proceedings of RIAO-94, 4th International Conference "Recherche d'Information Assistee par Ordinateur'"*, pages 146–160.

[Jones et Jackson, 1970] Jones, K. S. et Jackson, D. M. (1970). The use of automatically obtained keyword classification for information retrieval. *Information Storage and Retrieval*, 5.

[Jones et Needham, 1968a] Jones, K. S. et Needham, R. (1968a). Automatic term classification and retrieval. *Information Processing and Management*, 4:91 – 100.

[Jones et Needham, 1968b] Jones, K. S. et Needham, R. M. (1968b). Automatic term classifications and retrieval. *Information Storage and Retrieval*, 4(2):91–100.

[Kabel *et al.*, 2004] Kabel, S., de Hoog, R., Wielinga, B. J. et Anjewierden, A. (2004). The added value of task and ontology-based markup for information retrieval. *Journal of the American Society for Information Science and Technology*, 55(4):348–362.

[Kahan *et al.*, 2002] Kahan, J., Koivunen, M.-R., Prud'hommeaux, E. et Swick, R. R. (2002). Annotea : an open rdf infrastructure for shared web annotations. *In Computer Networks*, volume 39, pages 589–608.

[Kalfoglou et Schorlemmer, 2005] Kalfoglou, Y. et Schorlemmer, M. (2005). Ontology mapping : The state of the art. *In* Kalfoglou, Y., Schorlemmer, M., Sheth, A., Staab, S. et Uschold, M., éditeurs : *Semantic Interoperability and Integration*, numéro 04391 de Dagstuhl Seminar Proceedings. Internationales Begegnungs- und Forschungszentrum fuer Informatik (IBFI), Schloss Dagstuhl, Germany. <http ://drops.dagstuhl.de/opus/volltexte/2005/40> [date of citation : 2005-01-01].

[Karp, 1996] Karp, P. D. (1996). A strategy for database interoperation. *Journal of Computational Biology*, 2(4):573–583.

[Kaski *et al.*, 1998] Kaski, S., Honkela, T., Lagus, K. et Kohonen, T. (1998). Websom–self-organizing maps of document collections. *Neurocomputing*, 21:101–117.

[Kazai *et al.*, 2003] Kazai, G., Gövert, N., Lalmas, M. et Fuhr, N. (2003). The inex evaluation initiative. *Intelligent Search on XML Data, Applications, Languages, Models, Implementations, and Benchmarks*, pages 279–293.

[Kifer *et al.*, 1995] Kifer, M., Lausen, G. et Wu, J. (1995). Logical foundations of object-oriented and frame-based languages. *Journal of the ACM*, 42(4):741–843.

[Kim, 2005] Kim, H. H. (2005). Ontoweb : Implementing an ontology based web retrieval system. *JASIST*, 56(11):1167–1176.

[Kirk *et al.*, 1995] Kirk, T., Levy, A. Y., Sagiv, Y. et Srivastava, D. (1995). The Information Manifold. *Information Gathering from Heterogeneous, Distributed Environments.*

[Kirkpatrick, 1988] Kirkpatrick, B. (1988). Roget's thesaurus of english words and phrases. *concise edition (ed.), ISBN 0-14-051422-8.*

[Kiryakov *et al.*, 2003] Kiryakov, A., Popov, B., Ognyanoff, D., Manov, D., Kirilov, A. et Goranov, M. (2003). Semantic annotation, indexing, and retrieval. *2nd International Semantic Web Conference (ISWC2003), 20-23 October 2003, Florida, USA. LNAI Springer-Verlag Berlin Heidelberg*, 2870:484–499.

[Klein, 2002] Klein, M. (2002). Ontology versioning and change detection on the web. *International Conference on Knowledge Engineering and Knowledge Management (EKAW02). 2002. Siguenza, Spain.*

[Kobayashi et Takeda, 1999] Kobayashi, M. et Takeda, K. (1999). Information retrieval on the web : Selected topics.

[Kohonen, 1984] Kohonen, T. (1984). Self-organization and associative memory. *Springer-Verlag, Berlin, 1st edition.*

[Kohonen *et al.*, 2000] Kohonen, T., Kaski, S., Lagus, K., Salojarvi, J., Honkela, J., Paatero, V. et Saarela, A. (2000). Self organization of a massive document collection. *IEEE Transactions on Neural Networks, Special Issue on Neural Networks for Data Mining and Knowledge Discovery*, 11 :3:574–585.

[Kwon *et al.*, 1994] Kwon, O.-W., Kim, M.-C. et Choi, K.-S. (1994). Query expansion using domain adapted, weighted thesaurus in an extended boolean model. *CIKM'94*, pages 140–146.

[Lancaster, 1968] Lancaster, F. (1968). Information retrieval systems : Characteristics, testing and evaluation. *Wiley, New York.*

[Lancaster et Warner, 1993] Lancaster, F. et Warner, A. (1993). Information retrieval today. *Information Resources Press, Arlington, VA.*

[Landers et Rosenberg., 1982] Landers, T. et Rosenberg., R. L. (1982). An overview of multibase. *H.-J. Schneider, editor, Second International Symposium on Distributed Data Bases (DDB 1982)*, pages 153–184.

[Lassila et Swick, 1999] Lassila, O. et Swick, R. R. (1999). Resource description framework (rdf) model and syntax specification. *World Wide Web Consortium W3C Recommendation 22 February 1999 / W3C /.*

[Lee *et al.*, 1993] Lee, J. H., Kim, W. Y., Kim, M.-H. et Lee, Y.-J. (1993). On the evaluation of boolean operators in the extended boolean retrieval framework. *Proceedings of the 16th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval. Pittsburgh, PA, USA, June 27 - July 1*, pages 291–297.

[Lei *et al.*, 2006] Lei, Y., Uren, V. et Motta, E. (2006). Semsearch : A search engine for the semantic web. *15th International Conference on Knowledge Engineering and Knowledge Management Managing Knowledge in a World of Networks (EKAW 2006), Podebrady, Czech Republic*.

[Lenat et Guha, 1990] Lenat, D. B. et Guha, R. V. (1990). Building large knowledge based systems. *Reading, Massachusetts : Addison Wesley*.

[Lenzerini, 2002] Lenzerini, M. (2002). Data integration : A theoritical perspective. *Proc. of the 21st ACM SIGACT SIGMOD SIGART Symp. on Principles of Database Systems (PODS 2002)*, pages 233–246.

[Levy et Rousset, 1995] Levy, A. et Rousset, M.-C. (1995). Carin : A representation language integrating rules and description logics. *Proc. Int'l Description Logics, 1995*.

[Levy *et al.*, 1995] Levy, A. Y., Mendelzon, A. O., Sagiv, Y. et Srivastava, D. (1995). Answering queries using views. pages 95–104.

[Litvak *et al.*, 2005] Litvak, M., Last, M. et Kisilevich, S. (2005). Improving classification of multi-lingual web documents using domain ontologies. *KDO05 The Second International Workshop on Knowledge Discovery and Ontologies. Porto, Portugal October 7th*.

[Lopez, 2001] Lopez, F. (2001). Overview of methodologies for building ontologies. *IJCAI-99 Workshop on Ontologies and Problem-Solving Methods : Lessons Learned and Future Trends.Intelligent Systems*, CEUR Publications 1999, 16(1).

[MacGregor, 1991] MacGregor, R. (1991). Inside the loom description classifier. *SIGART Bulletin*, 2(3):88–92.

[Maedche et Staab, 2000] Maedche, A. et Staab, S. (2000). Semi-automatic engineering of ontologies from text. *Proceedings of the Twelfth International Conference on Software Engineering and Knowledge Engineering (SEKE'2000)*.

[Maron, 1961] Maron, M. E. (1961). Automatic indexing : An experimental inquiry. *J. ACM*, 8(3):404–417.

[McBride, 2004] McBride, B. (2004). The resource description framework (rdf) and its vocabulary description language rdfs. *Handbook on Ontologies*, pages 51–66.

[McBrien et Poulovassilis, 2003] McBrien, P. et Poulovassilis, A. (2003). Data integration by bi-directional schema transformation rules. pages 227–238.

[McGuinness, 1998] McGuinness, D. L. (1998). Ontological issues for knowledge-enhanced search. *Guarino, N., editor, Proceedings of the First International Conference on Formal Ontology in Information Systems, Trento, Italy,IOS Press.*, pages 302–316.

[McGuinness *et al.*, 2002] McGuinness, D. L., Fikes, R., Hendler, J. A. et Stein, L. A. (2002). Daml+oil : An ontology language for the semantic web. *IEEE Intelligent Systems*, 17(2)(5):72–80.

[McGuinness et van Harmelen, 2004] McGuinness, D. L. et van Harmelen, F. (2004). Owl web ontology language overview. *World Wide Web Consortium, Recommendation REC-owl-features-20040210*.

[McQueen, 1967] McQueen, J. (1967). Some methods for classification and analysis of multivariate observations. *Computer and Chemistry*, 4:257–272.

[Mena *et al.*, 2000] Mena, E., Illarramendi, A., Kashyap, V. et Sheth, A. P. (2000). Observer : An approach for query processing in global information systems based on interoperation across pre-existing ontologies. 8(2):223–271.

[Messai *et al.*, 2006] Messai, R., Zeng, Q. T., Mousseau, M. et Simonet, M. (2006). Building a bilingual french-english patient-oriented terminology for breast cancer. *In MEDNET'06, Toronto, Canada.*

[Miles et Brickley, 2005] Miles, A. et Brickley, D. (2005). Skos core guide. Rapport technique, 2nd W3C Public Working Draft 2 November 2005.

[Milo *et al.*, 2003] Milo, T., Abiteboul, S., Amann, B., Benjelloun, O. et Ngoc, F. D. (2003). Exchanging intensional xml data. *Proc. of SIGMOD*, pages 289–300.

[Minsky, 1968] Minsky, M. (1968). Semantic information processing. *MIT Press, Cambridge, MA.*

[Minsky, 1975] Minsky, M. (1975). A framework for representing knowledge. *The Psychology of Computer Vision, Winston, Patrick, ed. New York : Mc Graw Hill*, pages 211–77.

[Nardi et Brachman, 2003] Nardi, D. et Brachman, R. J. (2003). An introduction to description logics. pages 1–40.

[Neal *et al.*, 1998] Neal, P., Shapiro, L. G. et Rosse, C. (1998). The digital anatomist structural abstraction : a scheme for the spatial description of anatomical entities. *J Am Med Inform Assoc. AMIA'98 Symp. Suppl.*, pages 423–427.

[Nejdl *et al.*, 2002a] Nejdl, W., Wolf, B., Qu, C., Decker, S., Sintek, M., Naeve, A., Nilsson, M., Palmér, M. et Risch, T. (2002a). Edutella : a p2p networking infrastructure based on rdf. pages 604–615.

[Nejdl *et al.*, 2002b] Nejdl, W., Wolf, B., Staab, S. et Tane, J. (2002b). Edutella : Searching and annotating resources within an rdf-based p2p network.

[Noy et McGuinness, 2001] Noy, N. F. et McGuinness, D. L. (2001). Ontology development 101 : A guide to creating your first ontology. Rapport technique, Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880.

[Noy et Musen, 2004] Noy, N. F. et Musen, M. A. (2004). Ontology versioning in an ontology management framework. 19(4):6–13.

[Névéol *et al.*, 2006] Névéol, A., Rogozan, A. et Darmoni, S. (2006). Automatic indexing of online health resources for a french quality controlled gateway. *Information Processing and Management*, 42 :3:695–709.

[Ognyanov et Kiryakov, 2002] Ognyanov, D. et Kiryakov, A. (2002). Tracking changes in rdf(s) repositories. *EKAW '02 : Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management. Ontologies and the Semantic Web*, pages 373–378.

[Oliver *et al.*, 1999] Oliver, D. E., Shahar, Y., Shortliffe, E. H. et Musen, M. A. (1999). Representation of change in controlled medical terminologies. *Artificial Intelligence in Medicine*, 15(1):53–76.

[Pan et Heflin, 2003] Pan, Z. et Heflin, J. (2003). Dldb : Extending relational databases to support semantic web queries. *In* Volz, R., Decker, S. et Cruz, I. F., éditeurs : *PSSS'2003*, volume 89 de *CEUR Workshop Proceedings*. CEUR-WS.org.

[Parent et Spaccapietra, 1996] Parent, C. et Spaccapietra, S. (1996). Intégration de bases de données : Panorama des problémes et des approches. *Ing. des Syst. d'Info.*, 4(3).

[Patriarche *et al.*, 2005] Patriarche, R., Gedzelman, S., Diallo, G., Bernhard, D., Bassolet, C.-G., Ferriol, S., Girard, A., Mouries, M., Palmer, P. et Simonet, M. (2005). A tool for textual and conceptual annotation of documents. *Proceedings of E-challenges 2005, Ljublijana (Slovenia).*

[Perez *et al.*, 2006] Perez, J., Arenas, M. et Gutierrez, C. (2006). Semantics and complexity of sparql. pages 30–43.

[Petrini et Risch, 2004] Petrini, J. et Risch, T. (2004). Processing queries over rdf views of wrapped relational databases. *1st International Workshop on Wrapper Techniques for Legacy Systems, WRAP 2004, Delft, Holland.*

[Pham *et al.*, 2007] Pham, M.-H., Bernhard, D., Diallo, G., Messai, R. et Simonet, M. (2007). *SOM-based Clustering of Multilingual Documents Using an Ontology.* Data Mining with Ontologies : Implementations, Findings and Framewoks - Idea Group Inc. (to appear).

[Pinto *et al.*, 1999] Pinto, H., Prez, A. et Martins, J. (1999). Some issues on ontology integration. *Proceedings of the IJCAI-99 Workshop on Ontologies and Problem-Solving Methods (KRR5).*

[Pisanelli *et al.*, 1998] Pisanelli, D. M., Gangemi, A. et Steve, G. (1998). An ontological analysis of the umls metathesaurus. *AMIA 98 Annual Symposium.*

[Pisetta *et al.*, 2006] Pisetta, V., Hacid, H. et Zighed, D. A. (2006). Multi-catégorisation de textes juridiques et retour de pertinence. pages 235–246.

[Popov *et al.*, 2003] Popov, B., Kiryakov, A., Kirilov, A., Manov, D., Ognyanoff, D. et Goranov, M. (2003). Kim : A semantic annotation platform. *2nd International Semantic Web Conference (ISWC2003), 20-23 October 2003, Florida, USA. LNAI, Springer-Verlag Berlin Heidelberg*, 2870:834–849.

[Pottinger et Levy, 2000] Pottinger, R. et Levy, A. Y. (2000). A scalable algorithm for answering queries using views. *In* Abbadi, A. E., Brodie, M. L., Chakravarthy, S., Dayal, U., Kamel, N., Schlageter, G. et Whang, K.-Y., éditeurs : *VLDB 2000, Proceedings of 26th International Conference on Very Large Data Bases, September 10-14, 2000, Cairo, Egypt*, pages 484–495. Morgan Kaufmann.

[Pouliquen *et al.*, 2002] Pouliquen, B., Delamarre, D. et Beux, P. L. (2002). Indexation de textes m ?dicaux par extraction de concepts, et ses utilisations. *JADT2002 : 6 ?me Journ ?es Internationales d'Analyse statistique de Donn ?es Textuelles.*

[P ?rez-Rey *et al.*, 2006] P ?rez-Rey, D., Maojo, V., Garc ?a-Remesal, M., Alonso-Calvo, R., Billhardt, H., Martin-S ?nchez, F. et Sousa, A. (2006). Ontofusion : ontology-based integration of genomic and clinical databases. *Computers in Biology and Medicine*, 38 :(7-8):712–30.

[Quillian, 1968] Quillian, M. R. (1968). Semantic memory. *M. Minsky, editor, Semantic Information Processing, MIT Press*, pages 227–270.

[Rahm et Bernstein, 2001] Rahm, E. et Bernstein, P. A. (2001). A survey of approaches to automatic schema matching. *VLDB Journal : Very Large Data Bases*, 10(4):334–350.

[Rajashekar et Croft, 1995] Rajashekar, T. B. et Croft, W. B. (1995). Combining automatic and manual index representations in probabilistic retrieval. *JASIS*, 46(4):272–283.

[Rasmussen, 1992] Rasmussen, E. M. (1992). Clustering algorithms. *In Information Retrieval : Data Structures & Algorithms*, pages 419–442.

[Rector *et al.*, 1996] Rector, A., Rogers, J. et Pole, P. (1996). The galen high level ontology. *In Fourteenth International Congress of the European Federation for Medical Informatics, MIE-96, Copenhagen, Denmark.*

[Rector et Bechhofer, 1997] Rector, A. L. et Bechhofer, S. (1997). The grail concept modelling language for medical terminology. *Artif Intell Med*, 9(2):139–71.

[Reenskaug *et al.*, 1996] Reenskaug, T., Wolde, P. et Lehne, O. (1996). *Working With Objects : The Ooram Software Engineering Method.* F Manning/Prentice Hall, Princeton USA.

[Richardson et Smeaton, 1995] Richardson, R. et Smeaton, A. F. (1995). Using WordNet in a knowledge-based approach to information retrieval. Rapport technique, Dublin, Ireland.

[Rijsbergen *et al.*, 1981] Rijsbergen, C. J., Harper, D. et Porter, M. F. (1981). The selection of good search terms. *Information Processing and Management*, 17:77–91.

[Robertson et Jones, 1976] Robertson, S. et Jones, K. S. (1976). Relevance weighting of search terms. *Journal of the American Society for Information Science*, 27:129–46.

[Robertson *et al.*, 1995] Robertson, S., Walker, S. et Hancock-Beaulieu, M. (1995). Large test collection experiments on an operational, interactive system : Okapi at trec. *Information Processing and Management*, 31:345–360.

[Robertson, 1997] Robertson, S. E. (1997). The probability ranking principle in ir. pages 281–286.

[Rocchio, 1971] Rocchio, J. (1971). Relevance feedback in information retrieval. *In The Smart Retrieval System — Experiments in Automatic Document Processing.* Prentice-Hall, Englewood, Cliffs, New Jersey.

[Roche, 2001] Roche, C. (2001). The "specific-difference" principle : a methodology for building consensual and coherent ontologies. *IC-AI'2001 2000 : Las Vegas , USA.*

[Roche, 2003] Roche, C. (2003). The differentia principle as a cornerstone for ontology. *Knowledge Management and Philosophy, Workshop in WM 2003 Conference, Luzern.*

[Roche, 2005] Roche, C. (2005). Terminologie et ontologie. *LAROUSSE - Revue*, 157:1–11.

[Rosse et Mejino, 2003] Rosse, C. et Mejino, J. (2003). A reference ontology for bioinformatics : the foundational model of anatomy. *J Biomed Inform.*, 36:478–500.

[Roth *et al.*, 1996] Roth, M. T., Arya, M., Haas, L., Carey, M., Cody, W., Fagin, R., Schwarz, P., Thomas, J. et Wimmers, E. (1996). The garlic project. *In Proc. of the ACM SIGMOD Conf. on Management of Data, Montreal, Canada, June 1996.*

[Rousset *et al.*, 2006] Rousset, M.-C., Adjiman, P., Chatalic, P., Goasdoué, F. et Simon, L. (2006). Somewhere in the semantic web. *In* Wiedermann, J., Tel, G., Pokorný, J., Bieliková, M. et Stuller, J., éditeurs : *SOFSEM 2006 : Theory and Practice of Computer Science, 32nd Conference on Current Trends in Theory and Practice of Computer Science, Merín, Czech Republic, January 21-27, 2006, Proceedings*, pages 84–99. Springer.

[Rousset et Reynaud, 2003] Rousset, M.-C. et Reynaud, C. (2003). Picsel and xyleme : Two illustrative information integration agents. *In* Klusch, M., Bergamaschi, S., Edwards, P. et Petta, P., éditeurs : *Intelligent Information Agents - The AgentLink Perspective*, pages 50–78. Springer.

[Roussey *et al.*, 2001a] Roussey, C., Calabretto, S. et Pinon, J.-M. (2001a). A multilingual information system based on knowledge representation. pages 98–111.

[Roussey *et al.*, 2001b] Roussey, C., Calabretto, S. et Pinon, J.-M. (2001b). A new conceptual graph formalism adapted for multilingual information retrieval purposes. pages 92–101.

[Salton, 1968] Salton, G. (1968). *Automatic Information Organization and Retrieval.* McGraw Hill Text.

[Salton, 1971] Salton, G. (1971). A comparison between manual and automatic indexing methods. *Journal of the American Documentation*, 20(1).

[Salton et Buckley, 1988] Salton, G. et Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 24:513–523.

[Salton *et al.*, 1983a] Salton, G., Buckley, C. et Yu, C. (1983a). An evaluation of term dependence models in information retrieval. *LNCS*, 146:151–173.

[Salton *et al.*, 1983b] Salton, G., Fox, E. et Wu, H. (1983b). Extended boolean information retrieval. *Communications of the ACM*, 26(11):1022–1036.

[Salton et McGill, 1986] Salton, G. et McGill, M. J. (1986). Introduction to modern information retrieval. *McGraw-Hill, Inc., New York, NY,*.

[Sanderson, 1994] Sanderson, M. (1994). Word sense disambiguation and information retrieval. pages 142–151.

[Saracevic, 1970] Saracevic, T. (1970). *Introduction to Information Science*, chapitre The concept of "relevance" in information science : A historical review, pages 111–151. R.R. Bowker, New York.

[Savoy, 2005] Savoy, J. (2005). Bibliographic database accees using free-text and controlled vocabulary : an evaluation. *Information Processing and Management*, 41:873–890.

[Sebastiani, 2002] Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Comput. Surv.*, 34(1):1–47.

[Sheth et Kashyap, 1992] Sheth, A. P. et Kashyap, V. (1992). So far (schematically) yet so near (semantically). *In DS-5*, pages 283–312.

[Sheth et Larson, 1990] Sheth, A. P. et Larson, J. A. (1990). Federated database systems for managing distributed, heterogeneous, and autonomous databases. *ACM Comput. Surv.*, 22(3):183–236.

[Simonet et Simonet, 1995] Simonet, A. et Simonet, M. (1995). The p-type model : From databases to knowledge bases. *In KRDB.*

[Simonet et Simonet, 1999] Simonet, A. et Simonet, M. (1999). The ISIS methodology for object database conceptual modelling. *In Poster E/R 99, 18th Conference on Conceptual Modeling, Paris.*

[Simonet *et al.*, 2005] Simonet, M., Bernhard, D., Diallo, G., Gedzelman, S. et Patriarche, R. (2005). An environement for ontology design and enrichment from texts. *Proceedings of SWAP'05.*

[Simonet *et al.*, 2003] Simonet, M., Diallo, G., Palmer, P. et Simonet, A. (2003). Ontologies pour l'organisation des connaissances - vers une ontologie anatomo-fonctionnelle du cerveau. *Santé et Systémique (Hermès-Lavoisier)*, 7(1):44–75.

[Simonet *et al.*, 2006] Simonet, M., Patriarche, R., Bechlioulis, A., Bernhard, D., Diallo, G., Messai, R. et Gedzelman, S. (2006). Multilingual enrichment of an ontology of cardio-vascular diseases. *Computers in Cardiology (CINC'06) Proceedings.*

[Sirin et Parsia, 2004] Sirin, E. et Parsia, B. (2004). Pellet : An owl dl reasoner.

[SMALL, 1973] SMALL, H. (1973). Co-citation in the scientific literature : A new measure of the relationship between two documents. *J. Amer. Soc. Inform. Sci.*, pages 265–269.

[Smith, 1995] Smith, B. (1995). Formal ontology, common sense and cognitive science. *International Journal of Human-Computer Studies*, 43:641–667.

[Smith, 1998] Smith, B. (1998). Applied ontology : A new discipline is born. *Philosophy Today,*, 29:5–6. Preprinted in no ?sLetter (Bufalo) June 1998, pp. 3, 11.

[Solar et Doucet., 2002] Solar, G. V. et Doucet., A. (2002). M ?diation de donn ?es : solutions et probl ?mes ouverts. *Actes des deuxi ?mes assises nationales du GdRI3.*

[Sowa, 1984] Sowa, J. (1984). Conceptual structures : Information processing in mind and machine. *Addison Wesley.*

[Sowa, 1998] Sowa, J. F. (1998). Conceptual graph standard and extension. *ICCS 1998*, pages 3–14.

[Sowa, 1999] Sowa, J. F. (1999). *Knowledge Representation : Logical, Philosophical, and Computational Foundations.* Brooks Cole Publishing Co, Pacific Grove, CA.

[Stephen Harris, 2005] Stephen Harris, N. S. (2005). Sparql query processing with conventional relational database systems. *Proceedings of Web information Systems Engineering - WISE 2005 Workshops, New York, NY, USA*, pages 235–244.

[Stevens *et al.*, 2000] Stevens, R., Baker, P. G., Bechhofer, S., Ng, G., Jacoby, A., Paton, N. W., Goble, C. A. et Brass, A. (2000). Tambis : Transparent access to multiple bioinformatics information sources. *Bioinformatics*, 16(2)(2):184–186.

[Stojanovic *et al.*, 2002] Stojanovic, L., Maedche, A., Motik, B. et Stojanovic, N. (2002). User-driven ontology evolution management. pages 285–300.

[Studer *et al.*, 1998] Studer, R., Benjamins, V. R. et Fensel, D. (1998). Knowledge engineering : Principles and methods. *Data Knowl. Eng.*, 25(1-2):161–197.

[Sure *et al.*, 2002] Sure, Y., Angele, J. et Staab, S. (2002). Ontoedit : Guiding ontology development by methodology and inferencing. pages 1205–1222.

[Tenopir, 1985] Tenopir, C. (1985). Full text database retrieval performance. *Online Review*, 9 :2:149–164.

[ul Murshed et Ali, 2005] ul Murshed, A. et Ali, M. E. (2005). Evaluation and ranking of ontology construction tools. Rapport technique Technical Report DIT-05-013, University of Trento.

[Ullman, 2000] Ullman, J. D. (2000). Information integration using logical views. *Theor. Comput. Sci.*, 239(2):189–210.

[Uschold et Grüninger, 1996] Uschold, M. et Grüninger, M. (1996). Ontologies : principles, methods, and applications. *Knowledge Engineering Review*, 11(2):93–155.

[Uschold et King, 1995] Uschold, M. et King, T. (1995). Towards a methodology for building ontologies. *In Proceedings IJCAI-95, Workshop on Basic Ontological Issues in Knowledge Sharing, Canada.*

[Vallet *et al.*, 2005] Vallet, D., Fernandez, M. et Castells, P. (2005). An ontology-based information retrieval model. *In 2nd European Semantic Web Conference (ESWC 2005). Heraklion, Greece, May 2005. Springer Verlag Lecture Notes in Computer Science.*

[van Heijst *et al.*, 1997] van Heijst, G., Schreiber, A. et Wielinga, B. (1997). Using explicit ontologies in kbs development. *Int. J. of Human-Computer Studies*, 46(2/3):183–292.

[van Rijsbergen, 1977] van Rijsbergen, C. (1977). A theoretical basis for the use of co-occurrence data in information retrieval. *Journal of Documentation*, 33:106–119.

[van Rijsbergen, 1979] van Rijsbergen, C. J. (1979). Information retrieval, 2nd edition. *Dept. of Computer Science, University of Glasgow.*

[Visser *et al.*, 1999] Visser, P. R. S., Beer, M. D., Bench-Capon, T. J. M., Diaz, B. M. et Shave, M. J. R. (1999). Resolving ontological heterogeneity in the KRAFT project. *In Database and Expert Systems Applications*, pages 668–677.

[Voorhees, 1993] Voorhees, E. M. (1993). Using wordnet to disambiguate word sense for text retrieval. *ACMSIGIR '93, Pittsburg, PA, USA*, pages 171–180.

[Vossen, 1998] Vossen, P. (1998). Eurowordnet : A multilingual database with lexical semantic networks. *Kluwer Academic publishers.*

[Wache *et al.*, 2001] Wache, H., Vögele, T., Visser, U., Stuckenschmidt, H., Schuster, G., Neumann, H. et Hübner, S. (2001). Ontology-based integration of information — a survey of existing approaches. pages 108–117.

[Wang *et al.*, 2003] Wang, B. B., McKay, I. B., Abbass, H. A. et Barlow, M. (2003). A comparative study for domain ontology guided feature extraction. *Proceedings of the 26th Australian Computer Science Conference (ACSC-2003), Adelaide, Australia,*, pages 69–78.

[Welty et Nancy, 1999] Welty, C. et Nancy, I. (1999). Using the right tools : enhancing retrieval from marked-up documents. *J. Computers and the Humanities*, 33(10):59–84.

[Wiederhold, 1997] Wiederhold, G. (1997). Mediators in the architecture of future information systems. *In* Huhns, M. N. et Singh, M. P., éditeurs : *Readings in Agents*, pages 185–196. Morgan Kaufmann, San Francisco, CA, USA.

[Wielinga *et al.*, 1992] Wielinga, B. J., Schreiber, A. et Breuker, J. (1992). Kads : A modelling approach to knowledge engineering. *Knowledge Acquisition, Special issue "The KADS Approach to Knowledge Engineering"*, 4(1):5–53.

[Willet, 1988] Willet, P. (1988). Recent trends in hierarchical document clustering : a critical review. *Information Processing and Management*, 24(5):577– 597.

[Winograd, 1972] Winograd, T. (1972). Understanding natural language. *Academic Press, New York.*

[Wu et Buchmann, 1997] Wu, M.-C. et Buchmann, A. P. (1997). Research issues in data warehousing. pages 61–82.

[Wu *et al.*, 2003] Wu, S. H., Tsai, T. H. et Hsu, W. L. (2003). Text categorization using automatically acquired domain ontology. *Proceedings of IRAL2003 Workshop on Information Retrieval with Asian Languages, Sapporo, Japan,.*

[Yang et Garcia-Molina, 2002] Yang, B. et Garcia-Molina, H. (2002). Improving search in peer-to-peer networks. *Proceedings of the 22 nd International Conference on Distributed Computing Systems (ICDCS'02), Vienna, Austria.*

[Yang, 1999] Yang, Y. (1999). An evaluation of statistical approaches to text categorization. *Information Retrieval*, 1, 1-2:69–90.

[Yarowsky, 1992] Yarowsky, D. (1992). Word-sense disambiguation using statistical models of Roget's categories trained on large corpora. pages 454–460.

[Zdobnov *et al.*, 2002] Zdobnov, E. M., Lopez, R., Apweiler, R. et Etzold, T. (2002). The ebi srs server - recent developments. *Bioinformatics*, 18(2):368–373.

[Ziegler et Dittrich, 2004] Ziegler, P. et Dittrich, K. R. (2004). Three decades of data integration - all problems solved ? *In* Jacquart, R., éditeur : *18th IFIP World Computer Congress (WCC 2004), Volume 12, Building the Information Society*, volume 156 de *IFIP International Federation for Information Processing*, pages 3–12, Toulouse, France, August 22-27. Kluwer.

Résumé

Les systèmes d'information des organisations contiennent des données de diverses natures, dispersées dans une grande variété de sources. La gestion unifiée permet d'offrir un accès uniforme et transparent à cet ensemble hétérogène de sources. Nous nous intéressons à l'intégration de données structurées (bases de données relationnelles) et de données non structurées (sources textuelles, pouvant être multilingues) et particulièrement à la prise en compte de sources textuelles dans une infrastructure de gestion unifiée.

L'approche que nous proposons repose sur l'utilisation des technologies du Web sémantique et de différents types d'ontologies. Les ontologies servent d'une part à définir le schéma global d'intégration (ontologie globale) et les différentes sources à intégrer. Les ontologies qui représentent les sources à intégrer sont appelées schémas virtuels de sources ou ontologies locales (obtenues par un processus de rétroingénierie). D'autre part, les ontologies permettent d'effectuer une représentation hybride de chaque source textuelle qui combine des informations de catalogage, les vecteurs de termes, les vecteurs de concepts et, de façon optionnelle, les entités nommées ; tous ces éléments étant identifiés dans chaque document de la source. Nous avons par ailleurs élaboré une approche de gestion conjointe de plusieurs ontologies à travers un serveur d'ontologies qui sert notamment de support à l'interrogation.

Un premier domaine d'application de notre travail a été la gestion de données dans le domaine du cerveau. Nous avons construit ou enrichi des ontologies pour l'organisation des connaissances dans ce domaine, utilisées notamment pour la caractérisation sémantique de sources.

Mots-clés ontologies, recherche d'informations, intégration de données, web sémantique, annotation de documents, cerveau

Abstract

Organizations' information systems contain different kinds of data, dispersed in several sources. The purpose of the managing heterogeneous data is to offer a transparent access to this set of sources. We are interested in the management of structured (relational databases) and unstructured ((multilingual) textual sources) data. We especially describe an approach for taking textual sources into account in an integration system.

The approach we propose is based on the use of Semantic Web technologies and different kinds of ontologies. Ontologies are used to define the global schema (global ontology) and the sources to be integrated (local ontologies). Local ontologies are obtained in a semi-automatic way using reverse engineering techniques. Ontologies are also used for the hybrid representation of textual sources. The hybrid representation combines cataloguing information, vectors of terms and concepts and optionally named entities identified in documents. We have designed and implemented an ontology server to manage multiple ontologies and support queries.

A first application domain of our work has been the brain field. We have developed or enriched ontologies for brain knowledge management and semantic characterization.

Keywords ontologies, information retrieval, data integration, semantic web, documents annotation, brain