Définition d'un cadre formel de représentation des Systèmes d'Organisation de la Connaissance

Pierre-Yves Vandenbussche

UNIVERSITÉ PIERRE ET MARIE CURIE - PARIS VI

ÉCOLE DOCTORALE 393

ÉPIDEMIOLOGIE ET SCIENCES
DE L'INFORMATION BIOMÉDICALE

T H È S E

pour obtenir le titre de

Docteur en Sciences

de l'Université Pierre et Marie Curie - Paris VI

Mention : Informatique Biomédicale

Présentée et soutenue par

Pierre-Yves VANDENBUSSCHE

Définition d'un cadre formel de représentation des Systèmes d'Organisation de la Connaissance

Thèse dirigée par Jean CHARLET

préparée à l'INSERM U-872 Paris

Jury :

Rapporteurs :	Jean BÉZIVIN	-	Université de Nantes
	Fabien GANDON	-	INRIA
Directeur :	Jean CHARLET	-	AP-HP, INSERM
Examinateurs :	Bernd AMANN	-	Université Pierre et Marie Curie, LIP6
	Jean DELAHOUSSE	-	Mondeca
	Michel JOUBERT	-	LERTIM, Université de la Méditerranée

Table des matières

Table des figures

Liste des tableaux

Chapitre 1

Introduction

Sommaire

1.1 Contexte

Le travail de recherche présenté dans ce mémoire s'est déroulé en partenariat entre un industriel spécialisé dans la représentation de connaissances, MONDECA [1] et un laboratoire de recherche en Ingénierie des Connaissances dans le domaine de la santé, INSERM UMR_S 872 équipe 20 [2]. Les trois années de ce travail de thèse ont été financées conjointement par le Ministère de l'Enseignement supérieur et de la Recherche au travers d'une convention CIFRE et par l'entreprise MONDECA.

Au cours de ce travail de thèse, j'ai eu l'occasion de travailler sur de nombreux projets tant au sein du laboratoire de recherche que dans mon entreprise (*cf.* figure 1.1). La diversité de mes rôles (chercheur, développeur, chef de projet, consultant, formateur, avant-vente) et des projets (de recherche, privés) dans lesquels je suis intervenu sont la richesse de ce travail. Cette confrontation entre l'aspect de recherche

1. Voir http://www.mondeca.com/
2. Voir http://www.spim.jussieu.fr/

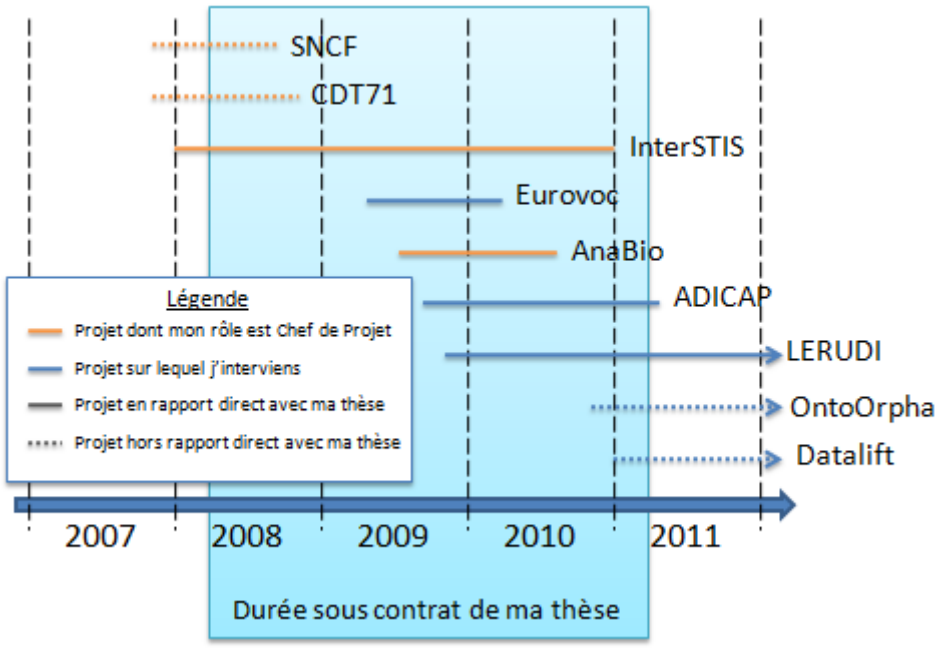

FIGURE 1.1 – Ligne de temps des projets et de mon implication durant ma thèse.

et l'aspect d'entreprise avec un besoin de résultats a été un moteur et a contribué à donner à ce travail de nombreux résultats pragmatiques. Nous détaillerons dans la suite de ce mémoire les projets en rapport direct avec nos travaux de thèse.

1.1.1 Contexte industriel

MONDECA est une entreprise fondée en 2000 et dirigée par Jean Delahousse qui édite des logiciels de gestion de systèmes d'organisation de la connaissance. MONDECA a conçu un logiciel, l'Intelligent Topic Manager^{TM}(ITM), pour répondre aux trois catégories de besoins exprimés par ses clients : (i) la fédération et l'organisation de contenus hétérogènes ; (ii) la gestion des référentiels métiers ; et (iii) la gestion des bases de connaissances dédiées. ITM permet aux entreprises d'organiser leurs contenus et connaissances autour d'une vision métier de leurs activités. Ce logiciel leur offre des solutions efficaces de recherche, de navigation, de raisonnement et de réutilisation de ces contenus et connaissances.

Ce travail de thèse a été motivé par l'intérêt de disposer d'un référentiel qui regroupe différentes ressources terminologiques ou ontologiques. Un tel référentiel permet de rendre des services d'extraction automatique de connaissances, d'annotation sémantique des contenus, de recherche d'informations, de raisonnement. Les ressources terminologiques ou ontologiques, des plus linguistiques aux plus for-

melles, doivent être gérées de façon cohérente, les unes par rapport aux autres et dans le temps. Actuellement, plusieurs domaines tels que la santé, la documentation ou encore la justice, utilisent de nombreuses terminologies, thésaurus, vocabulaires (dans la suite de nos travaux, nous utiliserons préférentiellement le terme *Système d'Organisation de la Connaissance (SOC)* pour désigner ces types de référentiel [Hodge 2000, Binding 2006]) comme support de gestion, d'échange et d'utilisation de connaissances. Une modélisation commune de ces systèmes pourrait permettre à l'outil ITM de devenir un serveur multi-terminologique qui offre des services unifiés d'édition, d'accès et d'échange de connaissances.

1.1.2 Contexte de recherche

L'unité INSERM UMR_S 872 équipe 20 dirigée par Marie-Christine Jaulent, a pour domaine de recherche l'informatique médicale et la santé publique. Le projet de l'équipe est de proposer des méthodes innovantes de gestion informatisée des données et des connaissances complexes en médecine à des fins d'amélioration de qualité des soins et de sécurité des patients. Les projets réalisés au sein de l'équipe s'inscrivent dans des contextes d'usages divers : la détection du signal en pharmacovigilance, la construction d'ontologies à partir de corpus textuels pour l'aide au codage médical du dossier patient, le traitement automatique de la littérature scientifique en biologie, l'informatisation des guides de bonnes pratiques, la mise en œuvre informatique de systèmes d'aide à la décision ou encore l'évaluation de l'impact des aides informatisées sur la pratique médicale.

Que ce soit pour des raisons économiques, sociales, d'étude de populations ou encore d'interactions entre établissements de santé, l'utilisation d'un ou plusieurs SOC est nécessaire. La possibilité de disposer d'un cadre pour gérer ces ressources peut grandement améliorer les travaux de recherche. Sur la base des terminologies locales et de leurs alignements, il serait possible de comparer un phénomène entre pays. A défaut d'avoir des SOC qui respectent un même standard ou qui satisfassent les besoins de chacun, nous devons envisager une solution qui prenne en compte l'hétérogénéité des SOC. Notre travail de thèse est motivé par ce constat.

1.2 Une approche pluridisciplinaire

Notre sujet de thèse prend place au sein du domaine de l'Ingénierie des Connaissances (IC) appliquée notamment à la discipline médicale. Considérer seulement le domaine de l'Ingénierie des Connaissances ne permet pas une compréhension de notre travail : notre sujet de thèse se comprend dans un cadre pluridisciplinaire. Nous

avons dirigé nos recherches pour résoudre une problématique de l'informatique médicale [3] grâce aux méthodes, techniques et outils issus de domaines complémentaires. L'Ingénierie des Connaissances mais aussi l'Ingénierie des Modèles (IM) apportent des moyens pertinents pour résoudre la problématique de notre travail.

1.2.1 L'informatique médicale

Un système de santé a pour objectifs (i) de mieux soigner les malades et (ii) de mieux gérer la distribution des soins [Dusserre 1985]. Avec l'évolution des techniques et le perfectionnement des instruments, les informations dans le domaine de la médecine se complexifient. La pratique d'une médecine moderne et de qualité ne peut être dissociée d'un traitement rationnel de l'information médicale. L'informatique médicale est une science qui aide à recueillir les faits, à les mémoriser, à les échanger et à les interpréter [Degoulet 1998]. Les Systèmes d'Organisation de la Connaissance (SOC) occupent à ce titre une place privilégiée. Parmi les catégories d'utilisation (nous reviendrons plus en détail sur les utilisations des SOC en section 3.5), nous pouvons retenir :

- le codage de dossiers patients ;
- la normalisation d'un vocabulaire commun pour faciliter l'échange d'information ;
- l'inscription d'un volume de connaissances qu'un cerveau ne peut mémoriser pour en extraire des tendances (data mining) ;
- la formalisation des connaissances pour effectuer des tâches automatiques plus « intelligentes » par un ordinateur (extension de recherche par inférence).

1.2.2 L'Ingénierie des Connaissances

L'Ingénierie des Connaissances est une discipline « qui élabore des systèmes d'inscription numérique et les instrumente pour leur conférer intelligibilité et exploitabilité dans des tâches requérant des connaissances pour leur réalisation » [Bachimont 2004]. Les travaux en Ingénierie des Connaissances s'attachent à « construire des modèles adaptés à la nature des connaissances à décrire pour pouvoir ensuite les représenter dans des formalismes adéquats » [Charlet 2002]. Les Systèmes d'Organisation de la Connaissance sont des produits de cette discipline. Notons que l'Ingénierie des Connaissances se situe à l'intersection de plusieurs réflexions comme l'informatique, la logique, la linguistique et la psychologie. La santé

3. Cette problématique peut être généralisée à l'ensemble des domaines utilisateurs de SOC. Le domaine de la santé est toutefois privilégié car précurseur de systèmes d'organisation de la connaissance.

n'est pas une composante de cette discipline mais un domaine d'application.

1.2.3 L'Ingénierie des Modèles

L'Ingénierie des Modèles est un domaine de l'informatique qui met à disposition des outils, des techniques et des langages pour créer et transformer des modèles [Bézivin 2005]. L'Ingénierie des Modèles est une nouvelle approche utilisée dans l'Ingénierie Logicielle et met les modèles (et non les programmes) au centre de la démarche [Favre 2006]. L'Ingénierie des Modèles est une forme d'ingénierie générative par laquelle tout ou partie d'une application informatique est générée à partir de modèles. Le processus de développement des systèmes peut alors être vu comme un ensemble de transformations de modèles ordonnés. Chaque transformation prend des modèles en entrée et produit des modèles en sortie jusqu'à obtenir des artefacts exécutables [Fleurey 2006]. Certaines approches de l'Ingénierie des Modèles que nous utilisons dans notre travail de thèse seront présentées et détaillées en section 2.3. L'Ingénierie des Connaissances vise à représenter des connaissances au travers de modèles. L'Ingénierie des Modèles est donc une discipline complémentaire à l'Ingénierie des Connaissances (une « boîte à outil supplémentaire ») qui permet de manipuler les modèles que l'on construit.

1.3 Synthèse et guide de lecture

1.3.1 Synthèse

1.3.1.1 La genèse de notre problématique

L'informatique occupe actuellement une place grandissante dans les établissements de soins confrontés à une politique socio-économique de rentabilité. Pour réduire les coûts, certains hôpitaux se regroupent, des services fusionnent. Il devient difficile de garder une communication effective transcendant les différences d'interprétation propres à chaque individu ou groupe d'individus. Les SOC permettent de partager un contexte commun de connaissances et ainsi réconcilier [4] la multiplicité des points de vue [Zweigenbaum 1999]. Ce besoin s'accentue avec la mondialisation de l'information qui essaye par exemple à l'échelle Européenne de détecter l'émergence d'une résistance de bactérie à un médicament [Lovis 2009]. Dans ce dernier

4. Comme pour tout système de référence, cette qualité de réconciliation n'est pas garantie. Nous reviendrons sur l'importance de son aspect consensuel lors de sa construction et de sa maintenance en section 2.3.1.

exemple, le partage d'informations peut être facilité par l'utilisation d'alignements entre les différents SOC de chaque pays ou hôpital.

La recherche en santé a besoin de manipuler de l'information qui est volumineuse, détaillée, multi-institutionnelle, multi-spécialisée, liée à d'autres types d'information (sociale, économique, environnementale). Pour répondre à ce besoin, il faut maximiser l'utilisation des informations actuellement disponibles en améliorant leurs mécanismes d'acquisition, d'inscription et de mise à disposition.

Une solution pour obtenir un tel système passe par [Chute 2000] :

- la création d'un contexte global partagé et explicite (des SOC et des modèles d'information utilisant ces SOC) ;
- la migration ou l'alignement des structures d'informations actuelles sur ce contexte (intégrer les modèles et les SOC dans le Système d'Information) ;
- la définition et la mise en place de méthodes pour satisfaire les besoins de partage d'information (développer des outils et des méthodes de gestion de l'information ainsi qu'appliquer des standards).

Notre travail de thèse s'intègre dans une vision où les SOC sont des briques élémentaires qui doivent être : accessibles à la demande, maintenues à jour et alignées [Chute 1999]. Notre travail vise à fournir un cadre formel de représentation unifiée de SOC et de leurs alignements. Ce cadre doit mettre son contenu à disposition et fournir les services nécessaires à son utilisation par des applications et des utilisateurs.

1.3.1.2 La difficulté de l'évaluation

L'évaluation de travaux de recherche en Ingénierie des Connaissances est complexe mais néanmoins importante puisqu'elle conditionne la crédibilité du caractère scientifique des approches proposées. Contrairement à d'autres disciplines, la recherche en Ingénierie des Connaissances propose peu de critères et les métriques y sont souvent absentes [Aussenac-Gilles 2005b]. Comme le souligne B. Bachimont, l'évaluation en Ingénierie des Connaissances ne correspond pas à la validation expérimentale d'une théorie ou d'hypothèses concernant des lois établies puisque n'étant pas une science mais une ingénierie [Bachimont 2004]. Les modèles conceptuels que nous construisons contribuent à doter le système d'information cible d'un comportement particulier. Leur validation, empirique, se mesure à leur capacité à rendre le système final pertinent et efficace dans la tâche prévue.

En ce qui concerne les modèles conceptuels, la communauté donne un sens particulier aux termes « validation » et « évaluation ». La phase de validation d'un modèle se produit au moment où celui-ci est présenté à l'expert. Le rôle de l'expert

est de valider ou d'invalider les choix de modélisation effectués. L'enjeu de cette phase est de s'assurer que la conceptualisation modélisée n'est pas en contradiction avec ses connaissances. La procédure d'évaluation, quant à elle, se propose de vérifier l'adéquation entre le modèle produit et les attentes spécifiées au début du projet. Les difficultés d'une telle évaluation tiennent au fait que (i) le modèle n'est qu'une composante de l'application cible dont les résultats sont difficilement isolables et (ii) les procédures d'évaluations sont dépendantes du type d'application finale.

Nous essayerons dans nos résultats, au-delà du respect du cahier des charges, de montrer et de quantifier (quand cela est possible) la valeur ajoutée de notre solution.

1.3.2 Organisation du mémoire

Après ce premier chapitre d'introduction de notre mémoire de thèse, nous poursuivons avec une première partie dédiée à l'état de l'art des artefacts, projets, standards, langages et systèmes que nous utilisons et auxquels nous nous positionnons.

Dans le 2e chapitre, nous proposons une revue des notions de base de notre travail. Cette revue retrace l'évolution de la représentation et de l'organisation de la connaissance jusqu'aux nouvelles technologies du Web sémantique. Nous insistons également sur l'importance de la formalisation des connaissances et de la modélisation.

Dans le 3e chapitre, nous présentons les Systèmes d'Organisation de la Connaissance dans tous leurs états. En partant de leur diversité et de quelques exemples, nous analysons leurs caractéristiques (composition, propriétés, utilisation). Enfin nous étudions le processus éditorial des SOC.

Dans le 4e chapitre, nous explorons les systèmes, langages, et projets principaux existants pour représenter des systèmes d'organisation de la connaissance. Cette étude nous permet de positionner notre travail au sein des travaux de recherche existants et de caractériser l'originalité de notre travail.

La seconde partie développe les méthodes et modèles que nous élaborons dans le cadre de ce travail de thèse.

Dans le 5e chapitre, nous exposons la problématique scientifique qui anime notre travail de recherche, mais aussi les enjeux liés à cette problématique ainsi que l'originalité de notre démarche.

Dans le 6e chapitre, nous abordons la construction de notre modèle pivot de représentation des systèmes d'organisation des connaissances.

La troisième partie porte sur la mise en application de nos travaux dans des cas concrets.

Dans le 7e chapitre, nous expliquons les outils et techniques que nous mettons en place pour proposer, autour de notre modèle, des services aux utilisateurs.

Dans le 8e chapitre, nous détaillons la mise en place de nos travaux au sein de plusieurs projets de recherche mais également industriels. L'atout majeur d'une thèse avec subvention CIFRE est la possibilité grâce à l'entreprise, de mettre en pratique nos travaux de recherche théorique dans des projets.

Dans le 9e chapitre, nous revenons sur les points importants de nos travaux et proposons de nouveaux axes de recherche que notre travail rend possible.

En annexe A, nous proposons une métaphore de l'apiculture pour appréhender notre travail sous un angle nouveau.

En annexe B, nous détaillons les interfaces et la navigation dans l'outil ITM. Ce logiciel est guidé par des modèles qui créent dynamiquement et contraignent les interfaces utilisateurs.

Première partie

Etat de l'art – vers la représentation des Systèmes d'Organisation de la Connaissance

Chapitre 2

De l'évolution et de la nécessité d'organiser la connaissance

Sommaire

L'objectif de ce chapitre est de présenter les notions de base sur lesquelles reposent nos travaux. Nous commençons par retracer l'héritage de la représentation des connaissances, activité vielle de plusieurs millénaires. Ce rapide historique nous amène à la dernière grande révolution dans ce domaine : le support et l'outil informatique. Dans ce cadre, nous explorons le Web sémantique qui réconcilie les interprétations humaine et computationnelles des connaissances représentées. Enfin nous présentons les méthodes existantes que nous utilisons pour modéliser des connaissances.

2.1 La connaissance et son organisation

2.1.1 La connaissance

Sans entrer dans une définition de la connaissance [1], essayons de la caractériser afin de mieux cerner pourquoi nous construisons et voulons représenter des Systèmes d'Organisation de la Connaissance. Pour cela nous nous appuyons sur les réflexions existantes [Kayser 1997, Ganascia 1998, Charlet 2002, Bachimont 2004] et nous nous focalisons sur les caractéristiques des SOC concernant leur inscription et leur échange :

La connaissance est **dépendante de son environnement technique** qui permet les actions d'inscription sur un support, de manipulation et d'échange. Cette dépendance est fondamentale : la nature des connaissances et des traitements qu'il est possible d'en faire procède directement du support et des techniques à disposition. Nous verrons ainsi dans la section 2.1.3 que la représentation de la connaissance évolue avec son support.

La connaissance peut être employée comme **code de communication**. Un code de communication dépend d'un support matériel, d'un environnement technique et des formes qui peuvent s'y inscrire (langage de représentation). Comme nous l'explique B. Bachimont, « Le code permet de communiquer dans la mesure où les formes matérielles donnent lieu à des réinterprétations par un interprétant visé comme le destinataire. Ces réinterprétations permettent au destinataire d'établir une réinscription qui explicite un sens supposé exprimé par l'auteur ou source de la communication ». La connaissance « nait » de l'interprétation, c'est à dire du sens donné à des signes, suivie d'une action (réinscription, utilisation, etc.).

2.1.2 Les Systèmes d'Organisation de la Connaissance

Après avoir identifié les principales caractéristiques de la connaissance, nous traitons les Systèmes d'Organisation de la Connaissance. Organiser, c'est utiliser **les moyens** (disponibles et/ou réunis) pour parvenir à des **finalités choisies**. Nous pouvons définir un SOC comme étant un ensemble de connaissances en interaction, représentées et regroupées au sein d'une structure dans le but de répondre à des besoins et d'atteindre des objectifs déterminés. Les SOC sont conçus avec une intention afin de répondre à un usage. En ceci, ce sont des modèles privilégiés d'étude de l'Ingénierie des Connaissances auxquels nous nous intéressons dans notre travail.

1. Les lecteurs intéressés peuvent se référer à [Bachimont 2004].

2.1.3 Un rapide historique de l'évolution de l'organisation des connaissances

Depuis le début des peintures rupestres jusqu'à aujourd'hui, la transmission de connaissances n'a eu de cesse d'évoluer et de s'organiser afin d'améliorer l'accès à l'information véhiculée. Dès la préhistoire, les peintures dans les grottes s'organisaient selon l'histoire qu'elles racontaient, puis aux environs de 12 000 ans avant notre ère, se sont réparties en trois zones spatiales (l'entrée, la zone de passage et le fond) de la grotte [Fayet-Scribe 1997].

Les premières apparitions de structuration après l'invention de l'écriture picto-idéographique en 4 000 av. J.-C. étaient des tableaux à but administratif, suivis en 800 av. J.-C. des premiers catalogues avec un classement thématique. C'est au premier siècle avant notre ère qu'apparaît la première encyclopédie « Antiquitates rerum humanarum et divinarum » de Varron présentant un classement systématique qui décrit en 41 livres l'histoire de l'Italie et de ses habitants. La première classification bibliothécaire apparaît au XVI^{e} siècle avec une proposition de classification universelle des œuvres passées et présentes.

Il faut attendre le début du XVII^{e} siècle à Londres pour voir apparaître une terminologie, dans le domaine médical. Celle-ci sert à recenser, comme l'illustre la figure 2.1, de manière hebdomadaire, les cas de mort selon leur cause au moyen de 44 termes comme « Peste », « étouffé », « subitement » ou « arrêt de l'estomac ». Les premières apparitions de collections de mots structurés selon leur sens datent du XIX^{e} siècle et sont nommées « thésaurus » par P.M. Roget dans le « Thesaurus of English Words and Phrases » [Roget 1856].

La **dernière révolution majeure** se passe dans les années 1960 avec **l'apparition de l'informatique**. Les recherches portent alors sur les techniques automatiques de recherche d'information linguistique (centrée sur le terme) mais également sémantique (centrée sur le concept). L'utilisation des thésaurus devient une évidence. Ces réflexions sont même à l'origine de la relation hypertexte qui permet de lier des notions ayant un sens proche ou connexe sur le Web.

L'intention de ces nouvelles organisations qui **capturent de la sémantique**, est de se rapprocher du fonctionnement non linéaire des idées dans la pensée humaine. Cette possibilité nourrit l'ambition de construire un modèle unique décrivant formellement les concepts régissant le monde appelé « ontologie » par analogie à l'Ontologie en philosophie (étude des propriétés générales de tout ce qui est). Cette vision rapidement remise en question sera discutée en section 3.3.6 où nous présentons la définition contemporaine d'une ontologie dans le domaine de l'Ingénierie des Connaissances.

The Diſeaſes and Caſualties this Week.

Abortive	5	Infants	13
Aged	36	Kingſevil	2
Apoplexie	1	Leproſie	1
Childbed	25	Meagrome	1
Chriſomes	20	Mother	1
Conſumption	130	Plague	2817
Convulſion	58	Pluriſie	1
Cough	2	Purples	2
Diſtracted	1	Quinſie	3
Dropſie	32	Rickets	14
Drowned in a Ditch at Saviours Southwark	1	Riſing of the Lights	32
Feaver	374	Rupture	3
Flox and Small-pox	11	Scowring	3
Flux	1	Scurvy	3
Griefe	3	Spotted Feaver	174
Griping in the Guts	70	Stillborn	11
Jaundies	2	Stone	5
Impoſthume	16	Stopping of the ſtomach	10
		Suddenly	2
		Surfeit	85
		Teeth	90
		Thruſh	4
		Tiſſick	3
		Ulcer	3
		Vomiting	1
		Wormes	18

Chriſtned: Males 90, Females 88, In all 178. Buried: Males 2022, Females 2008, In all 4030. Plague 2817.

Increaſed in the Burials this Week 1016.

Pariſhes clear of the Plague 44. Pariſhes Infected 86

The Aſſize of Bread ſet forth by Order of the Lord Mayor and Court of Aldermen:
A penny Wheaten Loaf to contain Nine Ounces and a half, and three half-penny White Loaves the like weight.

FIGURE 2.1 – Les « London Bills of Mortality » étaient publiés chaque semaine pour comptabiliser le nombre de morts et leur cause. Ce suivi permettait de contrôler l'évolution d'épidémies comme la peste.

L'informatique a énormément changé la conception, la maintenance et l'utilisation des systèmes d'organisation de la connaissance. Leur nombre et leur volume ont considérablement augmenté ces dernières décennies. L'outil informatique permet de réaliser des manipulations automatiques complexes qui produisent des résultats jusqu'alors impensables comme la déduction automatique de nouvelles connaissances.

2.2 Le Web sémantique et le Web de données

Les SOC que nous nous proposons de représenter servent de ressources de référence, ce qui facilite la compréhension et permet une interprétation commune et unique d'une information. Cette direction rejoint celle du Web avec l'avènement du

Web sémantique dans les années 2000.

2.2.1 Du Web au Web sémantique

Le World Wide Web s'est imposé comme un **espace unique d'échange d'informations sans frontière**. L'essor du Web a changé la manière de produire, de publier, de partager et d'utiliser l'information. De par l'ubiquité du Web, sa nature distribuée, sa maturité et ses possibilités d'évolution, il est le médium idéal pour le partage de connaissances [Heath 2011]. Ce succès est dû à l'infrastructure et aux technologies qui constituent maintenant les fondements du Web. La structure du réseau (l'Internet) permet facilement l'interconnexion (au moyen de liens hypertextes) de milliards de pages web et de documents. Les technologies d'identification ou de représentation de l'information telles que les URL [2] ou encore le langage HTML [3] sont maintenant universellement acceptées et utilisées. Toutefois, ces informations interprétables par les humains ne le sont par les machines qu'au travers des traitements automatiques des langues sur un contenu majoritairement textuel [4].

Pour permettre aux machines d'interpréter de manière plus « intelligente » les ressources sur le Web, le W3C [5] a développé un ensemble de technologies désignées par le terme « Web sémantique ». Le Web sémantique est décrit comme « une extension du Web actuel dans lequel on donne à une information un sens bien défini pour permettre aux ordinateurs et aux individus de travailler en coopération. » [Berners-Lee 2001]. Le Web sans son extension sémantique permet le partage d'informations non formelles, laissant à l'utilisateur le travail d'interprétation du contenu. Le Web sémantique est un ensemble de technologies qui s'ajoute au Web classique et qui permet **l'utilisation de connaissances formalisées**. Désormais, Le Web devient un espace de partage d'informations à destination des humains mais également de données formalisées **interprétables par les machines** (ce point est approfondi dans la section suivante). Cette infrastructure s'intéresse à l'identification des ressources, aux langages formels de représentation et aux langages d'interrogation, de partage et de preuves pour la publications de ressources.

2. Uniform Resource Locator (URL). Une URL identifie de manière unique une ressource sur le Web.

3. Hypertext Markup Language (HTML) est un langage de balisage qui permet de représenter des pages Web et des liens vers d'autres pages ou ressources grâce aux liens hypertextes.

4. Bien que ces dernières années la place des traitements sur contenu multimédia ait augmenté.

5. World Wide Web Consortium. Organisme de standardisation chargé de promouvoir la compatibilité et l'adoption des technologies du World Wide Web. Voir http://www.w3.org/

2.2.2 Le Web de données

Dans cet environnement est né un projet particulier : le Linked Open Data [6] (LOD) qui contribue à améliorer la qualité des données partagées en favorisant leur mise en relation et leur réutilisation [Bizer 2009a]. La force du LOD est de mettre en relation les ressources exprimées dans des jeux de données différents mais faisant **référence à un même objet**. De ce fait, il est possible d'avoir plusieurs vues potentiellement différentes sur un objet et ainsi construire une information multi-sources plus complète. Grâce à ce type d'approche, les données formalisées sur le Web ne se présentent plus comme des îlots de connaissances isolés mais comme un réseau global de connaissances. La figure 2.2 montre l'état de ce réseau de données disponible sur le Web en Septembre 2010. Les données sémantiques publiées sur le Web reposent sur l'utilisation de vocabulaires ou d'ontologies qui les structurent et nous aident à les décrire. Nous reviendrons en section 3.3.6 sur la nécessité des structures d'organisation de la connaissance pour l'expression de données formalisées et normalisées. Avec cette vision, le Web s'ouvre à une nouvelle perspective centrée sur des données formalisées et interprétables par les machines.

2.2.3 Les principales technologies du Web sémantique

Le Web sémantique repose sur la plateforme de langages techniques qui a participé à l'adoption du Web (*cf.* premier niveau « Web platform » de la figure 2.3). Il s'agit du processus d'identification des objets (URI), du protocole de transport (HTTP) et des services liés à la sécurité (AUTH). Cette première brique technologique constitue **le niveau technique**.

La formulation d'un Web sémantique universel est partie du principe de « modélisation minimaliste fondée sur un modèle commun généraliste » [Berners-Lee 1998]. Ce modèle général est le *Resource Description Framework* (RDF) [Klyne 2004]. Il s'agit d'une seconde brique technologique volontairement simple [7] pour permettre la plus grande adoption au sein d'applications (*cf.* second et troisième niveau « Formats » et « Information exchange » de la figure 2.3). Ce langage peut être sérialisé dans divers formats d'encodage tels que RDF-XML et RDF-TURTLE. Cette brique constitue **le niveau syntaxique**.

Alors que le langage RDF nous donne les éléments pour construire notre graphe de connaissances, il ne nous permet pas de définir une sémantique sur les assertions

6. Données Ouvertes Liées. Voir `http://linkeddata.org/`

7. Simple d'un point de vue de sa correspondance en logique mathématique. A ce niveau de description sémantique, le langage est assez limité et ne permet pas par exemple l'expression de négation ni d'implication logique.

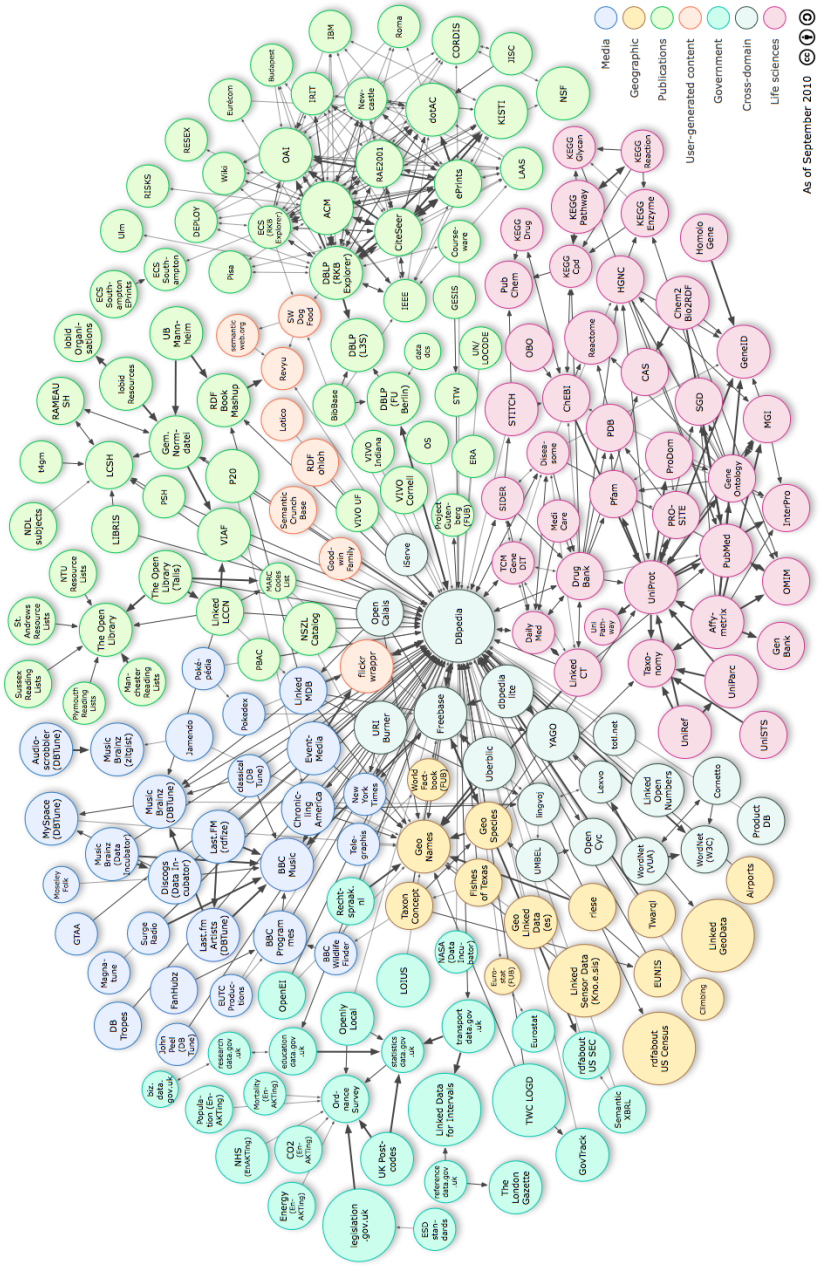

FIGURE 2.2 – Diagramme du réseau des jeux de données du Linking Open Data généré par Richard Cyganiak et Anja Jentzsch (`http://lod-cloud.net/`). Chaque cercle représente un jeu de données relié à d'autres par des relations d'équivalences entre des ressources des jeux de données sources et cibles. Un exemple de jeu de données est DBpedia [Auer 2007], une extraction sémantique formelle de données issue de Wikipedia.

que nous déclarons. C'est pour répondre à ce besoin que les langages RDF-Schema (RDFS) et Web Ontology Language (OWL) ont été définis (*cf.* quatrième niveau

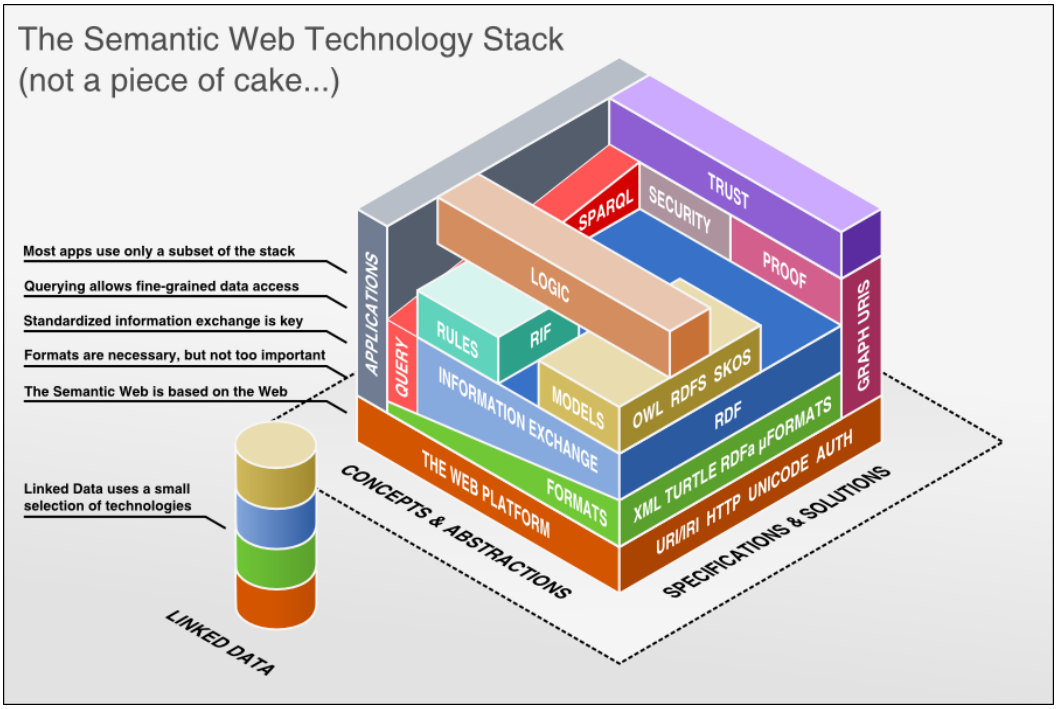

FIGURE 2.3 – La pile des technologies du Web sémantique actuel (http://bit.ly/rz5Re). Cette pile illustre une architecture cible du Web sémantique où certaines briques sont encore en cours de construction.

« Models » de la figure 2.3). En utilisant des expressions logiques, il est possible d'ajouter des contraintes, des propriétés, des classes et des relations sur le graphe d'assertions. Cette brique constitue **le niveau sémantique** et permet d'utiliser le formalisme logique de ces langages pour modéliser et capturer les connaissances d'un domaine. Nous présenterons en détail ces langages d'importance pour nos travaux en section 4.2.

Les autres niveaux de technologies du Web sémantique sont le niveau logique (utilisé par les moteurs d'inférences), les niveaux de preuve et de vérité. Ces derniers niveaux sont en cours de définition. Ils permettent de définir le niveau de confiance d'une information, de donner des mécanismes de vérification et d'apposer une signature électronique à une assertion.

Le Web sémantique définit un protocole et langage de requêtage pour interroger les graphes de connaissances exprimés en RDF (par extension RDFS et OWL) nommé SPARQL Protocol and RDF Query Language (SPARQL)[8]. SPARQL est capable d'exprimer une requête et de retourner des résultats en utilisant la formali-

8. « SPARQL » est une acronymie récursive.

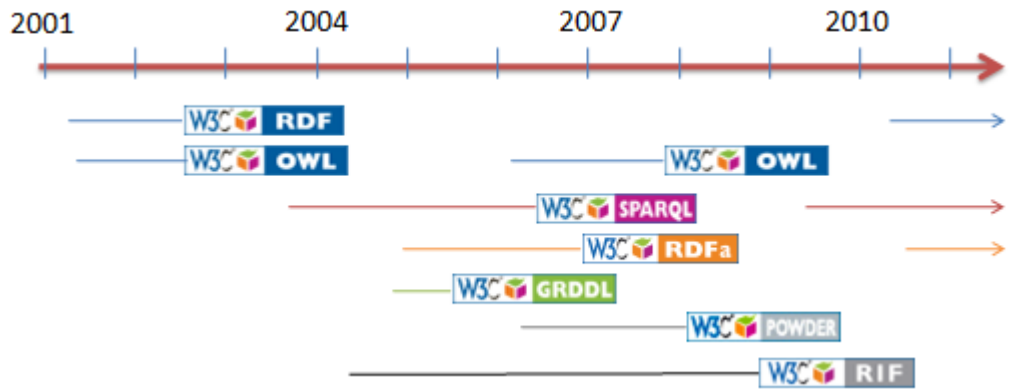

FIGURE 2.4 – Chronologie des principales technologies du Web sémantique.

sation d'une ontologie. SPARQL est décrit par le W3C comme le langage de requête et le protocole de transport des données sur le Web de Données. Nous détaillerons ce langage en section 4.4.1.

Les technologies du Web sémantique sont assez jeunes (*cf.* figure 2.4) comparativement à l'histoire de l'informatique mais bénéficient d'une **large adoption**. Ces langages évoluent pour s'adapter aux utilisations : le langage OWL (*cf.* section OWL) est dans sa version 2 depuis octobre 2009 ; le langage de requête SPARQL dans sa nouvelle version 1.1 est, en août 2011 à l'état de brouillon.

2.3 La modélisation et les modèles

2.3.1 Des produits de modélisation

En tant que produit de modélisation, les SOC possèdent certaines caractéristiques qui résultent de ce processus de construction.

La modélisation permet de représenter un phénomène complexe qu'il n'est pas possible d'observer directement. L'OMG [9] donne la définition suivante : « A model represents some concrete or abstract thing of interest, with a specific purpose in mind »[OMG 2001]. Cette définition met en avant la notion capitale **d'intention** d'une modélisation. Une modélisation est toujours faite avec un objectif précis (s'il n'est pas clairement défini, le modèle sera mal utilisé) qui va guider certains choix quant au modèle produit (granularité de la représentation, langage de description choisi, etc.). Ces choix auront un impact sur le **périmètre de validité** du modèle. Par exemple, si nous représentons l'eau comme un liquide, alors notre modèle n'aura

9. Object Management Group. Voir http://www.omg.org/

de validité que dans les conditions de pression et de température où l'eau est en phase liquide. Le choix d'un langage de description va être contraint par les propriétés du modèle : est-ce un modèle destiné à la communication, au traitement informatique ?

Rothenberg insiste dans sa définition sur **l'efficacité** et **la simplification** d'un modèle vis-à-vis de la réalité qu'il se propose de représenter : « Modeling, in the broadest sense, is the cost-effective use of something in place of something else for some cognitive purpose. It allows us to use something that is simpler, safer or cheaper than reality instead of reality for some purpose. A model represents reality for the given purpose ; the model is an abstraction of reality in the sense that it cannot represent all aspects of reality. This allows us to deal with the world in a simplified manner, avoiding the complexity, danger and irreversibility of reality. » [Rothenberg 1989]. Le degré de simplification choisi va directement impacter la granularité du modèle produit. La difficulté est de simplifier au maximum la représentation d'une partie du réel pour faciliter sa compréhension et son utilisation tout en gardant un niveau de détail suffisant pour atteindre l'objectif fixé.

La modélisation, comme toutes les activités humaines, est fondée sur des choix. Malgré la volonté d'objectivité, un modèle reste néanmoins **subjectif**. Il est important de veiller à son aspect **consensuel** dans la communauté de pratiques partageant les mêmes intentions. Ceci peut être atteint en impliquant un groupe d'experts représentatifs de cette communauté. Un moyen pour essayer de s'affranchir de cette subjectivité personnelle est d'utiliser des logiciels d'analyse des corpus textuels de documents produits par les experts en activité [Charlet 2006]. Mais cette subjectivité reste un des écueils majeurs pour de futures utilisations ou réutilisations. Ainsi, un objet du monde réel peut être modélisé au travers d'une infinité de points de vue. Si nous prenons l'exemple d'une terminologie de la médecine chinoise traditionnelle, ce point de vue sur les connaissances du domaine médical ne sera pas partagé par les collégiales adeptes de la médecine occidentale.

2.3.2 Model Driven Architecture

L'architecture dirigée par les modèles : Model Driven Architecture (MDA) est une initiative de génie logiciel proposée et soutenue par l'OMG [10] [Soley 2000]. Cette proposition d'architecture s'inscrit dans le domaine de l'Ingénierie des Modèles et place les modèles au centre de la réflexion : « Tout est modèle » [Bézivin 2004]. L'approche MDA n'est pas fondée sur une idée unique. Parmi ses propositions, nous détaillons dans les sections suivantes : le découpage des modèles en niveaux d'abstraction ; la définition de règles de transformation de modèles et la séparation

10. Object Management Group. Voir http://www.omg.org/

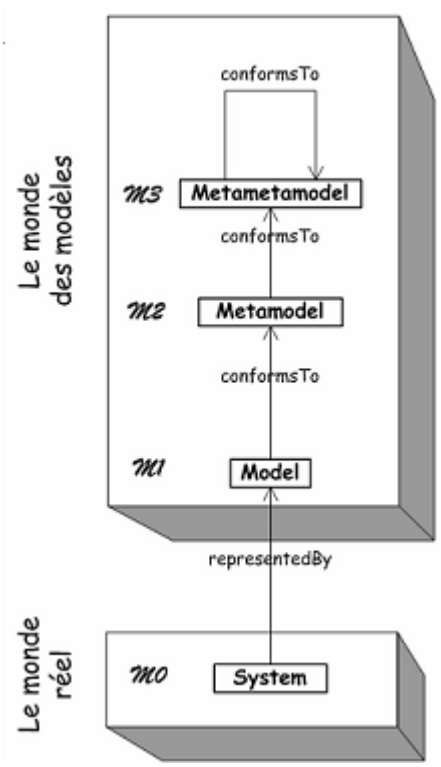

FIGURE 2.5 – Architecture de méta-modélisation en 3+1 couches. Cette figure est empruntée à [Bézivin 2004].

de descriptions métiers indépendantes de la plateforme de mise en œuvre.

2.3.2.1 Métamodèles

Pour décrire un modèle, il nous faut avoir un langage pour l'exprimer : un métamodèle. L'activité de méta-modélisation est la **création de formalismes de modélisation**, chaque modèle se conformant à un métamodèle prédéfini. De même, à un niveau d'abstraction supérieur, le métamodèle a besoin d'être clairement défini par un métamétamodèle. Afin d'éviter une décomposition infinie de niveaux d'abstraction, un patron d'architecture en 4 couches sert maintenant de référence. Comme le souligne J. Bézivin [Bézivin 2004], il est plus juste de nommer cette **architecture 3+1** (*cf.* figure 2.5) puisque la nature des relations entre chaque niveau n'est pas identique. Le niveau *m0* est le monde réel tandis que les niveaux *m1* à *m3* constituent le monde des modèles. *m1* est une représentation de *m0* ; *m1* est conforme à *m2* ; de même *m2* est conforme à *m3* ; et *m3* est conforme à lui-même (définition récursive).

La figure 2.6 illustre cette architecture de modèles. Dans cet exemple, le modèle

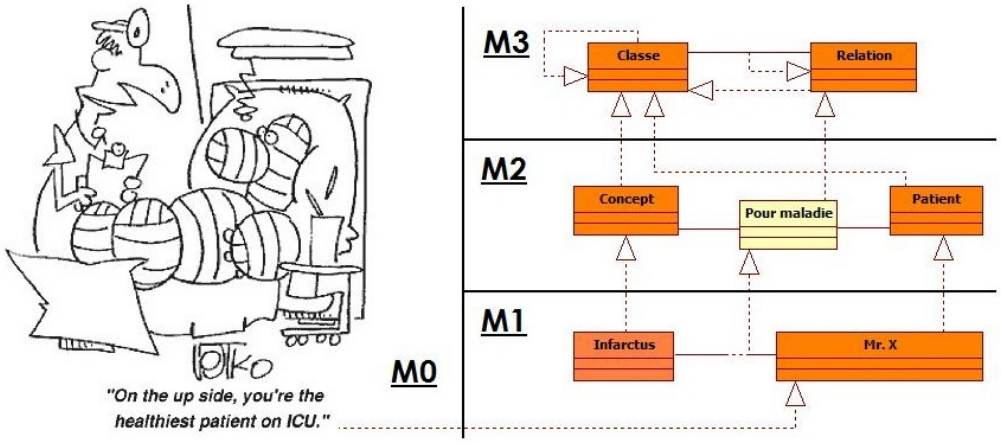

FIGURE 2.6 – Exemple de méta-modélisation en 3+1 couches. Notons que le niveau M0 est le monde réel. Toute illustration aussi fidèle soit-elle, est par nature une représentation et non le monde réel lui-même. Toutefois, par souci de compréhension, considérons notre illustration du niveau M0 comme la réalité. Le sigle « ICU », présent dans la figure, signifie Intensive Care Unit (en français : unité de soins intensifs).

M1 représente un patient qui a pour maladie un infarctus (cette connaissance représente le niveau *M0*). Les modèles *M1*, *M2* et *M3* sont conformes respectivement à leur métamodèle [11] *M2*, *M3* et *M3*. Pour bien comprendre cette architecture nous vous renvoyons à la spécification du MOF [12].

2.3.2.2 Transformation de modèles

La transformation de modèles est une problématique qui se situe au cœur de l'approche MDA. La transformation de modèles est le processus de génération automatique d'un modèle cible *Mb* à partir d'un modèle source *Ma* sur la base d'un ensemble de **règles de transformation** *Mt* [Bézivin 2001, Kleppe 2003] (*cf.* Figure 2.7). Une règle de transformation définit comment passer d'un ou plusieurs artefacts du modèle source à un ou plusieurs artefacts du modèle cible. En suivant les principes de méta-modélisation, le modèle source *Ma*, le modèle cible *Mb* et les règles de transformation *Mt* sont conformes respectivement aux métamodèles *MMa*, *MMb* et *MMt*. De même les métamodèles *MMa* et *MMb* sont conformes au même [13]

11. La notion de modèle et de *méta* est relative au niveau d'abstraction où l'on se place.
12. MOF (Meta Object Facility) est un modèle de niveau M3 réflexif, il définit la grammaire d'UML (Unified Modeling Language) au niveau M2. http://www.omg.org/mof/
13. Nous parlons donc d'une transformation de modèles exogènes

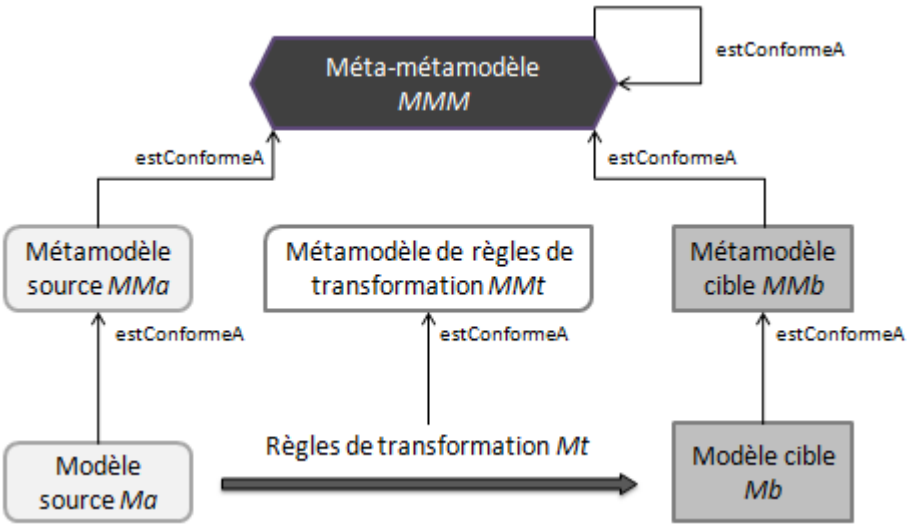

FIGURE 2.7 – Transformation de modèles exogènes du modèle source *Ma* vers le modèle cible *Mb*. Ces deux modèles partagent un métamétamodèle commun *MMM*.

métamétamodèle *MMM* qui est lui récursif.

Parmi les technologies du Web sémantique, le langage SPARQL (*cf.* section 4.4.1 permet grâce à sa clause « CONSTRUCT » d'effectuer des transformations de modèles conformes au métamétamodèle RDF. Nous reviendrons sur cette caractéristique du langage et son utilisation dans nos travaux en section 7.1.

2.3.2.3 Indépendance de description

Un des atouts de la méta-modélisation est la **séparation des préoccupations**. Chaque niveau de modèle identifie un degré d'abstraction précis et chaque modèle représente une vue particulière d'un système. L'approche MDA propose l'élaboration de différents **modèles successifs** : en partant d'un modèle métier indépendant de l'informatisation (Computation Independent Model, CIM) ; la transformation de celui-ci en modèle indépendant de la plateforme (Platform Independent Model, PIM) ; et enfin la transformation de ce dernier en modèle spécifique à la plateforme cible (Platform Specific Model, PSM) pour l'implémentation concrète du système [Bézivin 2005]. Cette démarche peut être adaptée pour la modélisation des connaissances (*cf.* section 5.3). Le but de cette approche n'est pas la génération de code source mais la production d'un modèle général (PIM) et de modèles dédiés à l'échange et à la visualisation (PSM), générés à partir de plusieurs transformations de modèles. Nous reviendrons sur l'utilisation de cette démarche en section 6.2.

2.4 Synthèse et discussion

Représenter, organiser et modéliser la connaissance n'est pas en soi quelque chose de nouveau. Nous employons quotidiennement ces techniques pour communiquer, comprendre et expliquer des informations efficacement : nous les utilisons dans les panneaux de signalisation, lorsque nous faisons un petit schéma explicatif sur une serviette ou encore lorsque nous écrivons notre liste de courses. Ces procédés ont toutefois évolué selon les outils et les supports mis à notre disposition. L'émergence de l'outil informatique au milieu du XX^{e} siècle a précipité les techniques et donc l'organisation de la connaissance dans une nouvelle ère. Ce changement accompagné par de nouveaux moyens de partage d'information à une échelle mondiale – le Web – bouleverse les techniques d'édition, de publication et d'utilisation des SOC (nous reviendrons sur ce point en section 3.6). L'utilisation de formalismes qui disposent d'une sémantique formelle pour représenter la connaissance offre de nouvelles manières de traiter l'information. C'est la voie que suit le Web sémantique. Il permet non seulement aux humains mais également aux machines d'interpréter la connaissance.

Nos travaux dont le cœur de l'activité est la modélisation de connaissances, peuvent bénéficier des méthodes existantes pour accomplir cette tâche. Nous nous sommes intéressés à l'initiative Model Driven Architecture qui propose un ensemble de méthodes pour construire et manipuler des modèles. Dans cette perspective, la méta-modélisation peut nous aider à positionner notre travail vis-à-vis des SOC que nous voulons représenter ; la transformation de modèles peut permettre de garantir l'interopérabilité vers ou depuis des modèles exprimés dans un langage similaire (nous préciserons cette technique en section 7.1) ; enfin, l'indépendance de description nous procure une méthode très pertinente pour nos travaux. En effet, la séparation de notre travail en plusieurs phases nous offre l'avantage de rester indépendant d'un schéma d'intégration et d'implémentation.

Chapitre 3

Des Systèmes d'Organisation de la Connaissance

Sommaire

Ce chapitre se propose de disséquer les Systèmes d'Organisation de la Connaissance que nous voulons représenter. Nous présentons tout d'abord la place importante

qu'occupent les SOC et leurs correspondances dans la résolution de l'ambiguïté d'interprétation des connaissances. De ce constat, nous analysons les caractéristiques de ces référentiels avant d'exposer leur diversité. Nous exemplifions ensuite l'hétérogénéité des SOC au travers de ressources que nous utiliserons dans nos travaux. Enfin nous étudions les différentes utilisations de ces SOC et le processus éditorial dans lequel ils s'inscrivent.

3.1 La présentation des SOC

3.1.1 L'ambiguïté d'interprétation de la connaissance

La représentation des connaissances sert non seulement à l'inscription d'un savoir-faire mais également à faciliter la circulation de l'information. Il existe cependant plusieurs freins à cet échange de connaissances dont **l'ambiguïté d'interprétation**. P. Zweigenbaum décompose cette notion en trois points [Zweigenbaum 1999] :

- **le manque de consensus** sur la définition d'une notion. Ceci peut s'expliquer par la différence de culture des utilisateurs de cette information ou par la variation dans le temps du fait de l'évolution des connaissances.
- **la polysémie** ou l'emploi d'un mot possédant plusieurs sens. Ainsi le mot « table » peut référer au mobilier ou encore à un élément d'une base de données.
- **l'imprécision** correspond à une description dont la spécification ne permet pas d'identifier de manière certaine une notion dans un contexte donné.

Pour répondre à ces problèmes, la représentation des connaissances s'est dotée d'un moyen de capturer le sens des informations qui se réfère à une vision du monde caractérisée par les trois sommets d'un « **triangle sémiotique** » illustré en figure 3.1 [Lerat 1989, Otman 1994, Scherrer 1997]. Dans cette vision, on représente un objet du monde réel en l'idéalisant sous forme de concept captant le sens (signifié). Pour parler de ce concept, nous utilisons par exemple un terme du langage naturel, une image, un signe. Cette vision fait l'objet de débats [Rastier 1995, Slodzian 2000] qui remettent en cause la rigidité des liens de ce triangle. Il est reproché au triangle sémiotique, de ne pas tenir compte de la **contextualité du sens** : le sens des mots varie selon leur contexte d'emploi. Toutefois, la connaissance d'un SOC tend à être consensuelle et sert de référence à un moment donné. Dans cet emploi, la signification des concepts qui sont décontextualisés est figée (avant une nouvelle version de ce SOC) et la vision de ce triangle sémiotique est recevable. Nous reviendrons sur la place du sens dans ces structurations en section 6.3.1. Grâce au sens qu'ils véhiculent,

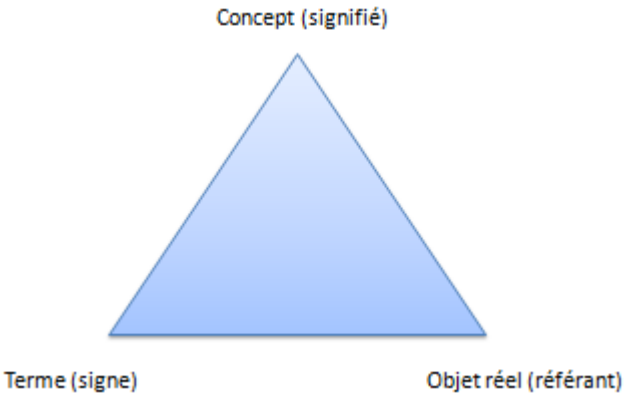

FIGURE 3.1 – Triangle sémiotique dans lequel le concept idéalise et se rapporte à un objet réel ; le signe symbolise le concept et représente l'objet.

les SOC occupent un rôle central pour la résolution des ambiguïtés d'interprétation.

3.1.2 L'anatomie des SOC

Dans cette section, nous proposons une présentation des différents éléments constituant des SOC. La nature et la complexité des SOC sont très variés (*cf.* section 3.3), de ce fait, certains éléments peuvent ne pas être présents dans tous les référentiels. Cette description repose sur les définitions proposées par la communauté d'Ingénierie des Connaissances [Gandon 2002].

Les concepts. L'élément de base d'un SOC est le concept. Cette entité représente par exemple un objet, une notion, une idée. Un concept est généralement exprimé par des termes (dans leur forme la plus simple : un libellé). Un même concept peut être exprimé de manières différentes. Par exemple, l'ablation de l'appendice iléo-cæcal peut se représenter par le concept dit pré-coordonné « Appendicectomie » ou par la post-coordination des concepts « excision » et « Appendice iléo-cæcal ». L'approche compositionnelle permet de décrire une infinité de concepts à partir d'un support fini. La terminologie SNOMED 3.5, présentée en section 3.4.1, est un exemple de SOC post-coordonné, tandis que la classification CIM-10, présentée en section 3.4.2, est pré-coordonnée. Les ontologies abordent la pré/post-coordination d'une manière différente. Les concepts simples (primitifs) d'une ontologie peuvent être reliés (par une définition formelle) entre eux pour définir un concept défini. La définition compositionnelle d'un concept bénéficie de l'expressivité de l'ontologie.

Les attributs. Les concepts des SOC comportent des attributs qui les caracté-

risent. Prenons par exemple, un SOC qui décrit les villes de France : le concept *Verlinghem* qui représente la ville de Verlinghem située dans le département du Nord, a un attribut pour décrire le nom de cette ville « Verlinghem » ou encore un attribut de géo-localisation ayant pour valeur « 50.68330,2.99530 ». Des attributs communément utilisés dans les SOC sont : **l'URI**, identifiant unique d'un concept ; **le code**, identifiant la place d'un concept dans la hiérarchie ; **le libellé**, pour nommer un concept.

Les relations. Les concepts prennent généralement place dans un réseau de concepts inter-reliés. Les relations sont le moyen de caractériser un lien entre deux ou plusieurs concepts. Dans notre exemple précédent, le concept *Verlinghem* a une relation *a pour ville limitrophe* le concept *Saint-André*. Des relations communément utilisées dans les SOC sont : **la relation de subsomption** (aussi appelée relation is-a) représente un lien de spécialisation d'un concept. La relation de subsomption est généralement une symbolique d'héritage d'attributs. Par exemple, le concept de *pomme* peut être considéré comme une spécialisation du concept de *fruit*. Ce type de relation est souvent confondu avec **la relation de partition** (aussi appelée relation has-a) : le concept de *graine* est une partie du concept de *fruit*. La relation de subsomption est aussi à différencier de **la relation d'instanciation** (aussi appelée relation instance-of) entre une classe et son instance. Dans notre exemple de fruit, les assertions : *Pink lady* qui est une instance du concept de *pomme* et *pomme* qui est une instance du concept de *fruit*, sont tous deux recevables et dépendent de l'intention du SOC. Dans certains SOC où la sémantique est moins importante, **la relation plus général/plus spécifique** (aussi appelée Broader-Narrower) est préférée et recouvre les deux relations de subsomption et de partition.

Les contraintes. Un SOC peut être décrit au moyen d'un langage formel ou ayant une correspondance dans une logique mathématique. Dans ce cas, des contraintes peuvent être ajoutées et contraindre les composants de ce SOC. Il est par exemple possible de spécifier une cardinalité de 1 sur la relation *est située dans le département* qui lie une ville à son département. Sur cette même relation, nous pouvons appliquer une restriction de domaine (concepts autorisés en sujet de la relation) au concept *ville* et appliquer une restriction de co-domaine (aussi appelé « range » : concepts autorisés en objet de la relation) au concept *departement*. Les contraintes disponibles dans un SOC sont **dépendantes du choix de langage formel**, comme OWL, pour décrire ce SOC.

Les instances. Un concept est parfois assimilé à une classe d'objets [Uschold 1995]. Une instance est un individu d'une classe (concept). Par exemple l'instance du concept de maladie est l'appendicite. De même que pour la méta-modélisation (*cf.* section 2.3.2.1), la définition de classe/instance est relative à l'intention de la modélisation. Le dernier exemple pourrait être vu d'une autre manière où appendicite serait un concept spécialisant le concept de maladie et instancié par l'appendicite de Monsieur x.

3.1.3 Leurs propriétés

Comme nous avons pu le voir, les SOC présentent des propriétés inhérentes à tout modèle mais également des propriétés spécifiques à la représentation de connaissances d'un domaine. Les propriétés majeures sont :

Le périmètre du domaine décrit (intention, couverture et granularité). La ressource doit avoir une définition claire de ses prétentions. Tout d'abord, la finalité du SOC doit être connue, c'est-à-dire l'application pour laquelle il a été construit. Si une ressource a un usage donné, alors elle décrit un domaine particulier avec une granularité de l'information représentée. Dans [Gandon 2002], F. Gandon détaille des méthodes pour capturer l'intention des modélisateurs. Ces méthodes partent de scénarios [Carroll 1995] ou de questions de compétence [Uschold 1996] (en anglais « compentency questions ») pour guider la construction d'un SOC ;

La volumétrie. Dépendant de son périmètre et de la granularité voulue, le volume d'un SOC peut fortement varier et contraindre les outils et les processus de maintenance et d'utilisation. Nous illustrons en section 3.4, la différence de volumétrie des SOC que nous utilisons dans nos projets ;

La richesse linguistique. La langue est un point d'entrée pour la recherche d'informations par une personne. L'expressivité linguistique va dépendre des primitives définies dans le modèle telles que la gestion de la langue sur un terme, les relations de synonymie et de traduction ;

L'expressivité formelle. Le caractère formel sous forme d'une logique mathématique permet d'opérer des traitements automatiques sur une ressource. Par exemple, grâce à la définition formelle d'une relation de subsomption, un raisonneur peut exécuter des traitements computationnels d'inférence sur la connaissance ou encore détecter les connaissances qui violent des contraintes formelles ;

La conformité aux normes, aux standards et aux recommandations.

L'utilisation de standards pour la construction d'un SOC ou pour l'échange de cette ressource favorise l'interopérabilité.

3.2 Les correspondances entre SOC

Nous séparons intentionnellement la présentation des correspondances de celle des SOC. En effet, les correspondances jouent un rôle très important dans l'amélioration de l'interopérabilité sémantique au même titre que les SOC, mais sont le produit d'une activité différente. Plusieurs termes sont utilisés dans la littérature pour définir le mécanisme de mise en correspondance entre les concepts de différents SOC. nous renvoyons le lecteur aux définitions données par J. Euzenat *et al.* [Euzenat 2007]. Dans le reste de notre manuscrit, nous utiliserons les termes « alignement » pour décrire la méthode de mise en correspondance de SOC et « correspondance » pour décrire le résultat de cette méthode. Un alignement est défini comme une fonction qui produit un ensemble de correspondances à partir de deux SOC.

T. Merabti distingue deux catégories de méthodes d'alignement dans ses travaux [Merabti 2010] :

Les méthodes lexicales. Elles utilisent les propriétés lexicales attachées aux concepts (principalement leur libellé ou terme principal) pour définir une distance syntaxique entre eux. Elles représentent la façon la plus triviale d'identifier des correspondances entre concepts. Aux correspondances produites sont généralement associés des scores de confiance ;

Les méthodes structurelles Elles sont fondées sur les structures hiérarchiques ou relationnelles des SOC pour identifier les correspondances. Ces méthodes sont souvent combinées avec les méthodes lexicales pour une plus large couverture. En utilisant la hiérarchie (ou d'autres types de relations) souvent présente dans un SOC, ces alignements permettent de repérer des correspondances entre différents niveaux de granularité de concepts (correspondance plus spécifique, etc.).

Ces méthodes sont accompagnées d'outils automatiques, semi-automatiques ou manuels. Toutefois, pour obtenir des correspondances de qualité (qu'il sera possible de valoriser dans des applications), l'intervention d'experts du domaines des SOC est nécessaire. De tels outils sont décrits dans la littérature [Cimino 2001, Sun 2004, Euzenat 2004, Choi 2006, Mazuel 2009].

3.3 La diversité des SOC

Il existe tout un panel de systèmes de structuration pour représenter les connaissances : liste contrôlée, classification, thésaurus, terminologie, ontologie, etc. Nous rappelons que nous utilisons préférentiellement le terme *Système d'Organisation de la Connaissance (SOC)* pour désigner tout type de référentiel précédemment énoncé [Hodge 2000, Binding 2006]. Face à ce pluralisme, un constat peut être fait : la complexité au sein de chaque structure de ressources rend leur catégorisation difficile [Ingenerf 1998]. Il existe ainsi des terminologies plus ou moins formelles, des thésaurus avec une structure plus ou moins complexe, des ontologies avec ou sans contraintes sur leurs relations. Commençons par replacer l'éventail des ressources auxquelles nous sommes confrontés et dont nous empruntons les définitions aux écrits existants sur le domaine.

3.3.1 Classification

Une classification peut être définie comme étant « une opération de l'esprit qui, pour la commodité des recherches ou de la nomenclature, pour le secours de la mémoire, pour les besoins de l'enseignement, ou dans tout autre but relatif à l'homme, groupe artificiellement des objets auxquels il trouve quelques caractères communs, et donne au groupe artificiel ainsi formé une étiquette ou un nom générique » [Cournot 1851]. Cette définition a peu évolué depuis un siècle et demi, en témoigne la définition suivante : « Une classification est la répartition systématique en classes, en catégories d'êtres, de choses ou de notions ayant des caractères communs notamment afin d'en faciliter l'étude ; c'est aussi le résultat de cette opération » [Bourigault 2004]. Une classification a donc pour principe la répartition d'objets en classes suivant leur similarité. Ces classes sont hiérarchiquement organisées selon un principe générique-spécifique. Lors de la classification de document, celui-ci est rangé dans une seule classe selon une méthode de rétrécissement dans la hiérarchie classificatoire.

Un exemple de classification est la CIM-10 que nous présenterons en section 3.4.2.

3.3.2 Nomenclature

Le mot nomenclature provient du latin *nomenclatura* qui signifie une classe ou ensemble de mots. Une nomenclature est définie comme une liste méthodique, systématique des objets, des éléments d'un ensemble. Dans notre domaine, une nomenclature désigne une liste contrôlée de termes techniques d'un domaine. Il s'agit d'une

des formes les plus simples d'organisation de la connaissance. Cette structuration ne possède aucun agencement particulier, mais vise à l'exhaustivité d'un domaine particulier. La principale caractéristique d'une nomenclature tient à la précision de l'objectif suivi. Alors que la classification est clairement orientée vers un objectif précis, la nomenclature a pour seul but l'exhaustivité.

3.3.3 Terminologie

Une terminologie « est une liste de termes d'un domaine ou d'un sujet donné représentant les concepts ou notions les plus fréquemment utilisés ou les plus caractéristiques, cette liste étant ou non structurée » [Lefèvre 2000]. Cette caractéristique potentiellement structurelle fait du terme « terminologie », un terme utilisé plus généralement pour nommer indifféremment les SOC. Cette structuration présente l'ensemble des termes spécialisés relevant d'un domaine et d'un usage donné comme le précise R. Dubuc dans sa définition : une terminologie est utilisée « dans une situation concrète de fonctionnement de façon à répondre aux besoins d'expression de l'usager » [Dubuc 1977]. Contrairement à un thésaurus, dans une terminologie, l'accent est mis sur l'exhaustivité des termes et des relations linguistiques telles que la synonymie, l'hyperonymie, etc. Aujourd'hui, la terminologie est au cœur de plusieurs disciplines, notamment les disciplines documentaires. La terminologie se retrouve également dans la rédaction technique, la rédaction de documentation et de manuels d'emploi, la traduction automatique et plus généralement dans la gestion de connaissances. Les termes d'une terminologie sont obtenus par normalisation de mots ou syntagme nominaux (groupes de mots). Un exemple de terminologie est la SNOMED 3.5 [Lussier 1998] (*cf.* section 3.4.1).

3.3.4 Thésaurus

Un thésaurus est un ensemble structuré de termes normalisés organisés au sein d'une hiérarchie de concepts liés par des relations sémantiques. Les termes sélectionnés sont nommés concepts ou descripteurs parce qu'ils sont destinés à l'analyse de contenu et au classement des documents d'information. Un concept de thésaurus peut être défini comme « une représentation mentale stable d'un aspect de la réalité qui peut être pensé en l'absence de cette réalité » [AFN 1981]. Le concept tend à rester indépendant de tout contexte particulier. Nous approfondirons cette notion de concept en section 6.3.1. Les descripteurs d'un thésaurus sont utilisables en coordination pour l'indexation ou la recherche d'un document. Par exemple, l'article intitulé « Building medical ontologies by terminology extraction from texts: an experiment for the intensive care units » écrit par J. Charlet *et al.* est indexé par les

descripteurs MeSH « Humans ; Intensive Care Units ; Medical Informatics ; Natural Language Processing ; Semantics ; Software ; Terminology as Topic ; Vocabulary, Controlled » [1].

Eurovoc (*cf.* section 3.4.5) et le MeSH (*cf.* section 3.4.3) sont des thésaurus.

3.3.5 Taxonomie

A l'origine, la taxonomie décrivait une partie de la biologie visant à établir une classification systématique des êtres vivants. Mais à l'instar des *terminologies* et des *thésaurus*, le sens d'une *taxonomie* a beaucoup changé pour maintenant embrasser plusieurs techniques et applications. parmi lesquelles, A. Gilchrist [Gilchrist 2003] distingue *les répertoires sur le Web.* Que ce soit pour la navigation ou la classification, les « taxonomies » sont couramment utilisées sur Internet et de plus en plus dans les Intranets. Il s'agit, en réalité, plus souvent de classification possiblement multi-axiale. Cette structure permet à un utilisateur d'affiner sa recherche en explorant les branches de la hiérarchie. Un exemple de ces utilisation et appellation de taxonomies est celui de Yahoo ! (*cf.* figure 3.2) qui organise de cette manière certains sites Web d'intérêt. Ce guide pour les utilisateurs n'en est pas moins une œillère et par là même un facteur de non-acceptation de cette technique. Une autre initiative, nommée *Open Directory Project* [Sherman 2000], est fondée sur la collaboration de plus de 35 000 éditeurs volontaires et comporte plus de 350 000 termes. Ce dernier exemple est présenté en Figure 3.3. Une telle taxonomie fondée sur un consensus favorise l'acceptation par les utilisateurs de guider leurs recherches.

3.3.6 Ontologies

Avec l'évolution des outils informatiques et les possibilités de calcul, la représentation formelle des connaissances au sein d'ontologies occupe une place centrale en Ingénierie des Connaissances et plus largement en Intelligence Artificielle. Pour illustrer cette évolution et l'intérêt grandissant envers cette structure, nous présentons en Figure 3.4, l'occurrence des termes *ontology, taxonomy, thesaurus* et *terminology* dans les livres anglophones entre 1900 et 2008. Cette illustration n'a pas de valeur scientifique mais nous renseigne sur l'utilisation et l'intérêt de ces types de structuration dans le temps. Nous voyons principalement la corrélation de l'accroissement continu de l'utilisation du terme *ontology* avec l'apparition des outils informatiques dans les années 1950 et plus récemment à partir des années 1990.

La première apparition du mot « ontologie » date du XVII^{e} siècle. Il désigne alors

1. Information provenant de Pubmed : http://www.ncbi.nlm.nih.gov/pubmed/16198328

FIGURE 3.2 – Taxonomie utilisée dans les répertoires de Yahoo ! Directory (`http://dir.yahoo.com/`) pour catégoriser des sites Web.

FIGURE 3.3 – Taxonomie créée au sein du projet Open Directory Project (`http://www.dmoz.org/`). Il s'agit d'une taxonomie obtenue par la participation ouverte de tout utilisateur.

une discipline en philosophie initiée par Aristote. Le Petit Robert définit l'Ontologie dans ce domaine comme étant « la partie de la métaphysique qui s'intéresse à l'Être en tant qu'Être ». Toutefois, ce terme, appliqué à l'Ingénierie des Connaissances, possède une signification différente. Nous l'écrirons avec un « o » minuscule dans ce contexte comme le recommande N. Guarino et P. Giaretta [Guarino 1995]. T.R. Gruber propose une première définition et introduit la notion de conceptualisation : « une ontologie est une spécification partagée d'une conceptualisation » [Gruber 1993]. Les attendus des ontologies se sont précisés depuis le début de leur

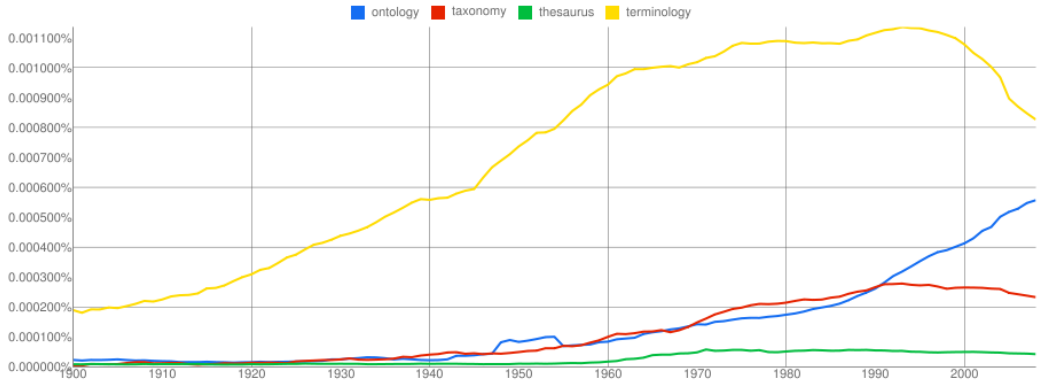

FIGURE 3.4 – Statistique d'occurrence de termes liés aux SOC dans les livres anglophones fournie par l'outil Ngram Viewer développé par Google (http://ngrams.googlelabs.com/).

utilisation en informatique dans les années 1990. En effet, leurs objectifs sont devenus plus modestes et plus pragmatiques. La définition de J. Charlet met l'accent sur la finalité d'une ontologie et son domaine d'application : une ontologie est constituée d'un « ensemble de concepts et de relations pour une utilisation particulière d'un domaine déterminé ; cette structure repose sur une formalisation mathématique afin d'effectuer des traitements logiques tels que l'inférence dans un système informatique » [Charlet 2006]. Une ontologie en Ingénierie des Connaissances est considérée comme un artefact qui doit être **opérationnel** à la différence de la discipline d'Ontologie en philosophie.

Une ontologie présente une vue d'un domaine particulier avec un but donné. Cette conceptualisation est rendue explicite et non ambiguë par la spécification d'une signification aux concepts du domaine (formalisation dans une logique mathématique). En tant qu'objet de modélisation, cela implique qu'une ontologie demeure subjective par nature (généralement établie par consensus d'un groupe d'experts) et non exhaustive. Le but n'est pas de décrire la totalité d'un domaine mais de rendre efficace la connaissance exprimée dans l'ontologie au regard de sa finalité.

La littérature fait état de nombreux outils et méthodes pour construire, maintenir et utiliser des ontologies. Nous renvoyons le lecteur aux travaux de J. Charlet [Charlet 2002] et F. Gandon [Gandon 2002].

L'état actuel des technologies proposées par le W3C concernant le Web sémantique accorde une place privilégiée aux ontologies [Davies 2003, Heath 2011]. Le Web de données (*cf.* section 2.2.2) repose sur la définition de jeux de données exprimés grâce à des ontologies [Amann 2010]. Les nouveaux comportements Web sociaux

(aussi appelés Web 2.0) ouvrent la voie à de nouvelles méthodes de construction d'ontologies. Ainsi les tags créés par les utilisateurs pour annoter des documents Web (aussi appelé folksonomie) peuvent servir de base à la construction d'ontologies [Gandon 2008, Limpens 2008]. Certaines méthodes permettent même d'extraire des relations sémantiques à partir de ces folksonomies [Mika 2005] et confirment cette nouvelle source de connaissance pour l'élaboration d'ontologie.

Les thésaurus et ontologies capturent tous deux de la sémantique, mais leur **engagement formel** est différent : alors que les thésaurus reposent sur quelques notions formelles telle que la subsomption, les ontologies utilisent l'expressivité d'un langage formel mathématique. Cette différence influe directement sur l'utilisation que l'on aura de ces référentiels.

Des exemple d'ontologies sont Menelas [2] [Zweigenbaum 1995], le FMA [3] [Rosse 2003] ou la SNOMED-CT [4] [Stearns 2001].

3.4 Les SOC d'intérêt pour nos travaux

3.4.1 SNOMED 3.5

La Systematized Nomenclature of Medicine (SNOMED) est une terminologie multi-axiale [5] hiérarchique. Cette structure a été développée à l'origine par le collège des pathologistes américains comme une nomenclature fonctionnelle pour les pathologies en médecine clinique. Cette SOC s'est par la suite organisée en hiérarchie pour proposer en 1993 dans sa version 3.5 (aussi appelée « International ») une répartition en 11 axes [Côté 1993] avec plus de 100 000 concepts [6] (*cf.* table 3.1). Cette terminologie couvre tous les champs de la médecine, de la dentisterie humaine, ainsi que de la médecine vétérinaire. Chaque terme est identifié par un code composé de la lettre de l'axe auquel il appartient et une suite alphanumérique (e.g.« DA-75630 » identifie le concept dont le terme principal (ou préféré) est « Conjonctivite chronique »).

2. Voir : http://estime.spim.jussieu.fr/Menelas/. L'ontologie est disponible en OWL à l'adresse : http://bit.ly/owOciG

3. Voir : http://fma.biostr.washington.edu/

4. Voir : http://1.usa.gov/VTGCg

5. Une structure multi-axiale est composée d'entités qui figurent potentiellement à plusieurs endroits de la hiérarchie.

6. Nous reviendrons sur la distinction terme/concept en section 6.3.1. Considérons ici, que la SNOMED 3.5 est une terminologie. On peut estimer que chaque terme fait référence à un concept qui prend place dans une hiérarchie et participe à des relations sémantiques. L'emploi des mots « terme » et « concept » est, dans ce cas, interchangeable.

TABLEAU 3.1 – Statistiques concernant la SNOMED 3.5.

Élements de la terminologie et caractéristiques	Valeur
Profondeur de la hiérarchie	7
Nombre de concepts	106 171
Nombre de relations générique/spécifique	106 160
Nombre de relations « est décrit par »	62 497

L'indexation d'une ressource en utilisant la SNOMED 3.5 repose sur une combinaison de termes appartenant à différents axes (post-coordination). Ainsi un diagnostic est traduit par plus d'un élément signifiant, mais chaque axe ne doit pas être obligatoirement validé. Par exemple, la juxtaposition : T2856 (lobe supérieur du poumon gauche) / M4100 (inflammation) / F0300 (fièvre) / E2012 (pneumocoque) correspond à la phrase « Pneumonie fébrile à pneumocoque du lobe supérieur gauche ». L'ajout de connecteurs concernant notamment les liens de causalité permet de décrire un fait complexe en plusieurs phrases [Baneyx 2007].

La SNOMED 3.5 est l'un des SOC médicaux les plus complets mais comporte certaines limites. Un même concept peut y être décrit de différentes façons et rien n'empêche de créer par combinaison des concepts inconsistants [Le Bozec 2001]. Une autre limite vient de son format de publication non adapté à la représentation de SOC par l'Agence des Systèmes d'Information Partagés de Santé (ASIP-Santé). Nous reviendrons en détail sur cette dernière limite en section 5.2.1.1.

En raison de ce type de défaut, la SNOMED 3.5 a évolué en SNOMED-RT (pour Root Procedure) puis en SNOMED-CT (pour Clinical Terms). Cette évolution traduit la volonté de résoudre les problèmes d'ambiguïté de non consistance et d'aboutir à une ontologie formelle [Dolin 2001]. la SNOMED-CT est conforme au modèle HL7 version 3 et repose sur une sémantique formelle (logique de description). En janvier 2008, elle contenait plus de 311 000 concepts. Cette ontologie a pour vocation d'être utilisée pour tout document clinique, allant du dossier patient électronique, jusqu'à la télé-médecine.

3.4.2 CIM-10

La classification internationale des maladies a pour appellation complète Classification statistique internationale des maladies et des problèmes de santé connexes (en anglais : International Statistical Classification of Diseases and Related Health Problems). La désignation usuelle abrégée de « Classification internationale des maladies » est à l'origine du sigle couramment utilisé pour la désigner : « CIM » (en

Tableau 3.2 – Statistiques concernant la CIM-10.

Élements de la classification et caractéristiques	Valeur
Profondeur de la hiérarchie	6
Nombre de concepts	19 856
Nombre de relations générique/spécifique	20 403
Nombre de relations d'exclusion	6 766

anglais : ICD). La CIM permet le codage des maladies, des traumatismes et de l'ensemble des motifs de recours aux services de santé. Cette classification est éditée et publiée par l'Organisation Mondiale de la Santé (OMS). Elle est utilisée à travers le monde pour enregistrer les causes de morbidité et de mortalité à des fins diverses, parmi lesquelles le financement et l'organisation des services de santé ont pris ces dernières années une part croissante. La dernière version de la CIM est la 10^{e}, publiée en 1993. Au moment de l'écriture de ce mémoire, la CIM est en cours de révision pour la publication d'une 11^{e} version prévue pour 2015 [7]. La classification de l'OMS sert, en France, au codage des causes de décès ainsi qu'au regroupement des séjours hospitaliers en groupes homogènes de malades dans le cadre du PMSI [8].

La CIM-10 est une classification mono-axiale [9] divisée en 21 chapitres en fonction de caractères communs (étiologique, topographique, physiologique ou pathologique). Chaque élément de cette classification est identifié par un code alphanumérique comportant trois à cinq caractères (e.g. « H10.4 » identifie la classe « Conjonctivite chronique »). La hiérarchie de la CIM-10 a une profondeur de 6 niveaux et possède en plus d'une relation générique/spécifique, une relation d'exclusion (*cf.* table 3.2).

3.4.3 MeSH

Le Medical Subject Headings (MeSH) [10] est un thésaurus édité et publié par la National Library of Medicine (NLM) [11] pour indexer les publications scientifiques

7. Voir http://www.who.int/classifications/icd/revision/en/index.html

8. Le Programme de médicalisation des systèmes d'information (PMSI) est un dispositif médico-administratif du système de santé français. Grâce au recueil des informations codées sur le séjour et les interventions des patients, il est le moyen de mesurer l'activité afin de réduire les inégalités de ressources entre les établissements de santé.

9. Une structure mono-axiale est composée d'entités identifiées de manière unique n'apparaissant qu'à un endroit de la hiérarchie. Cette propriété lève toute ambiguïté et s'applique ainsi très bien aux classifications.

10. Voir http://www.ncbi.nlm.nih.gov/mesh

11. Bibliothèque de médecine des États-Unis d'Amérique. Voir http://www.nlm.nih.gov/

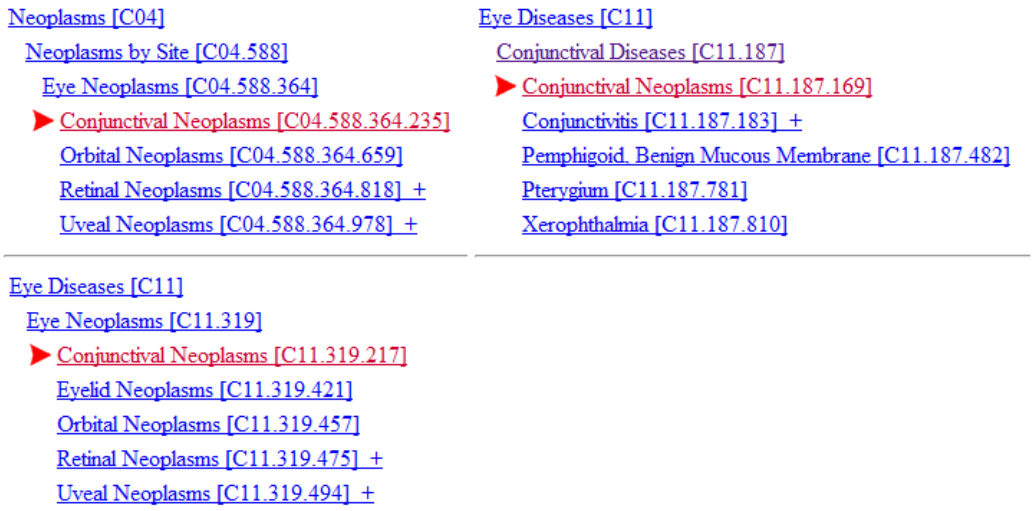

FIGURE 3.5 – Structure de la hiérarchie MeSH pour le concept « Conjunctival Neoplasms ». Capture écran du navigateur MeSH proposé par le NLM accessible à cette adresse : `http://1.usa.gov/pakMhN`

de santé dans le système bibliographique biomédical automatisé de stockage et de recherche MEDLARS [Austin 1968] (devenu depuis MEDLINE [12] regroupant plus de 18 millions d'articles). Ce thésaurus multi-axial maintenu et publié chaque année, est divisé en 15 axes ayant chacun un code spécifique (*e.g.* « C » identifie « maladie »). La hiérarchie du MeSH comprend des *Headings* et débouche sur des *Concepts*. Chaque *Concept* possède un ou plusieurs *Term* qui portent l'entrée terminologique au thésaurus. La figure 3.5 montre la hiérarchie MeSH pour le concept « Conjunctival Neoplasms ». Le détail du concept est illustré en Figure 3.6 [12]. on remarque que chaque élément de ce SOC est identifié par un identifiant *UI* ou *ID* unique et porte des métadonnées supplémentaires comme la date de création *Date* ou encore des remarques historiques *History Note*. En outre, des connecteurs, qui permettent des références explicites entre termes, expriment des relations de synonymie, de voisinage ou d'association tandis que des qualificatifs permettent de considérer les différentes facettes d'un concept. Par exemple, « cancer des os/traitement médicamenteux » permet de restreindre le cancer des os au seul aspect du traitement médicamenteux. Deux autres types de relations existent :

– La relation « voir aussi » permet de naviguer d'un mot clé à l'autre et de relier des termes proches.

12. Accessible grâce au moteur de recherche Pubmed sur le site `http://www.ncbi.nlm.nih.gov/pubmed/`

MeSH Heading	Conjunctival Neoplasms		
Tree Number	C04.588.364.235		
Tree Number	C11.187.169		
Tree Number	C11.319.217		
Annotation	/ blood supply / chem / second / secret / ultrastruct permitted; coord IM with histol type of neopl (IM)		
Concept 1 (Preferred)	**Conjunctival Neoplasms**		
	Concept UI	M0005014	
	Scope Note	Tumors or cancer of the CONJUNCTIVA.	
	Semantic Type	T191 (Neoplastic Process)	
	Term (Preferred)	Conjunctival Neoplasms	
		Term UI	T009419
		Date	01-JAN-1999
		Lexical Tag	NON
		Thesaurus	NLM (1981)
Allowable Qualifiers	BL BS CF CH CI CL CN CO DH DI DT EC EH EM EN EP ET GE HI IM ME MI MO NU PA PC		
Entry Version	CONJUNCTIVAL NEOPL		
Previous Indexing	Conjunctiva (1966-1980)		
Previous Indexing	Eye Neoplasms (1966-1980)		
History Note	81		
Date of Entry	19800404		
Unique ID	D003230		

FIGURE 3.6 – Vue détaillée du concept MeSH « Conjunctival Neoplasms ». Capture écran du navigateur MeSH proposé par le NLM accessible à cette adresse : `http://1.usa.gov/pakMhN`

– La relation « ne pas confondre » permet de préciser le sens et de lever les ambiguïtés.

En janvier 2009, le MeSH contenait plus de 25 000 concepts et de 455 000 termes (*cf.* table 3.3).

3.4.4 LOINC

Logical Observation Identifiers Names and Codes (LOINC) est un dictionnaire de référence pour l'identification d'observations biomédicales par les laboratoires. Il est édité et publié tous les 6 mois par le Regenstrief Institute [13] et répond à la demande d'informatisation de la prise en charge médicale clinique. Son domaine d'application s'est étendu depuis sa création en 1994 pour inclure, en plus des éléments de laboratoire et de clinique, les diagnostics et interventions infirmiers et la

13. Le Regenstrief Institute est une organisation à but non lucratif des États-Unis d'Amérique. Voir `http://www.regenstrief.org/`

TABLEAU 3.3 – Statistiques concernant le MeSH 2009.

Élements du thésaurus et caractéristiques	Valeur
Profondeur de la hiérarchie	11
Nombre de concepts	25 588
Nombre de relations générique/spécifique	27 322
Nombre de relations d'exclusion	6 766

prise en charge des patients. LOINC est mis à disposition gratuitement [14] et disponible au format de base de données dans une version complète ou comportant uniquement les changements depuis la dernière version. Plusieurs standards tels que IHE [15], HL7 [16] ou openEHR [17] recommandent l'utilisation du dictionnaire LOINC lors d'échanges d'information entre un hôpital et un laboratoire d'analyses biomédicales. Nous reviendrons sur l'utilisation de LOINC par les Systèmes d'Information Hospitaliers en section 4.1.

Le dictionnaire LOINC présente un ensemble de plus de 60 000 *Records* (correspondant à des concepts) et composés eux-mêmes de 6 éléments fondamentaux : *Component*, *Property*, *Time Aspect*, *System*, *Scale Type*, *Method* (en français : analyte, propriété, temps de mesure, milieu, unité, méthode). Chaque *Record* est identifié par un code unique (*e.g.* « 12190-5 » identifie l'observation de la proportion de créatinine dans le sang « Creatinine [Mass/volume] in Body fluid »). Le dictionnaire n'est pas organisé hiérarchiquement mais présente une relation de dépréciation (*cf.* table 3.4).

3.4.5 Eurovoc

Eurovoc est un thésaurus multilingue (24 langues) édité et publié par l'Office des Publications de l'Union Européenne [18]. Il permet d'indexer les documents dans les systèmes documentaires des institutions Européennes et de leurs utilisateurs. La

14. Accessible à l'adresse : http://loinc.org/

15. Integrating the Healthcare Enterprise (IHE) est une initiative visant à améliorer l'interopérabilité entre les acteurs de santé. Concrètement, cela passe par la définition de profils *IHE profiles* qui utilisent le dictionnaire LOINC pour l'échange d'information entre un hôpital et un laboratoire d'analyses biomédicales.

16. Health Level Seven (HL7) est une organisation à but non lucratif impliquée dans le développement international de standard d'interopérabilité en santé.

17. open Electronic Health Record (openEHR) est une spécification standardisée ouverte. Ce standard s'intéresse à la même problématique que le standard HL7.

18. Voir http://europa.eu/eurovoc/

Tableau 3.4 – Statistiques concernant LOINC 2.34.

Élements du thésaurus et caractéristiques	**Valeur**
Profondeur de la hiérarchie	1
Nombre de concepts	61 255
Nombre de relations de dépréciation	1 568

Tableau 3.5 – Statistiques concernant Eurovoc 4.3.

Élements du thésaurus et caractéristiques	**Valeur**
Profondeur de la hiérarchie	5
Nombre de concepts	6 797
Nombre de relations générique/spécifique	6 320
Nombre de relations (hors générique/spécifique)	8 563

construction de ce thésaurus est conforme aux normes ISO 2788-1986 et ISO 5964-1985 et se compose de *descripteurs* ou termes préférentiels, de *non-descripteurs* ou termes non-préférentiels organisés au sein d'une classification hiérarchique chapeautée par une décomposition en groupes à deux niveaux (domaines et microthésaurus). Les relations sémantiques utilisées sont au nombre de quatre : relation d'appartenance au microthésaurus (MT) ; relation d'équivalence de synonymie entre un terme préférentiel et un terme non-préférentiel (*Used For* et *Use*) ; relation hiérarchique (*Broader Term* et *Narrower Term*) ; relation associative (*Related Term*). En termes de volumétrie, il est composé de 6 797 concepts reflétés par plus de 270 000 termes (préférés ou non) toutes langues confondues (*cf.* table 3.5). Chaque concept et chaque groupe est identifié de manière unique (e.g. « 5216 » identifie le groupe « détérioration de l'environnement » ou encore « 218649 » identifie le concept « séisme »). Ce thésaurus, maintenu dans 24 langues, nécessite la prise en compte d'un processus de traduction et de validation mais également d'une traçabilité complète qui permet par exemple de connaître l'auteur d'une traduction. La figure 3.7 montre les informations du concept *séisme*.

FIGURE 3.7 – Vue détaillée du concept Eurovoc « séisme ». Capture écran du navigateur Eurovoc accessible à cette adresse : `http://bit.ly/qlaocp`. On remarque que la traduction en Maltais (*MT*) du terme préféré désignant ce concept, n'a pas encore été réalisée (*under translation*).

3.5 Les utilisations

Dans ses travaux, O. Bodenreider [Bodenreider 2008] distingue trois grandes catégories d'utilisation des SOC [19] que sont (i) la gestion de connaissances, (ii) l'interopérabilité sémantique et (iii) l'aide à la décision et le raisonnement. Comme le montrent les projets exposés en chapitre 8, l'utilisation pratique des SOC met souvent en jeu plusieurs de ces thématiques pour atteindre un but souvent plus complexe et plus complet.

3.5.1 La gestion de connaissances

Une des fonctions principales des SOC est de rassembler le vocabulaire d'un domaine, *i.e.* une liste des libellés des entités décrites dans le référentiel. Cette **partie terminologique** des SOC est une composante essentielle pour le traitement automatique des langues [Bodenreider 2006] et pour permettre des tâches liées à la gestion de connaissances telles que l'annotation, l'indexation de ressources ou l'accès et la recherche d'information. Cependant, la prise en compte de l'aspect

19. Nous ne reprenons pas l'utilisation du mot « ontologie » que O. Bodenreider emploie pour désigner n'importe quel type de structure.

terminologique et la finesse de l'expressivité linguistique varient beaucoup entre les différents SOC. L'expressivité linguistique va dépendre des primitives définies dans le modèle des SOC telles que la gestion de la langue sur un terme, les relations de synonymie, de méronymie et de traduction.

L'annotation.

L'annotation consiste en l'apposition d'éléments d'un vocabulaire contrôlé à un document. Ces annotations utilisées par l'indexation, servent à la recherche d'information. Par exemple, l'annotation et l'indexation d'un article biomédical grâce au MeSH. Ce processus appliqué aux documents de santé est souvent nommé *coder* [Giannangelo 2006].

L'accès à l'information.

La principale fonction de l'indexation de documents est le support à la recherche d'information. En effectuant une recherche textuelle, il est possible d'accéder à un document grâce aux entrées linguistiques et leurs variations (synonymes, abréviations, etc.) contenues dans le SOC utilisé pour l'annotation. Les recherches peuvent être étendues aux concepts plus spécifiques que celui originellement trouvé grâce à la hiérarchie présente dans la SOC [Greenberg 2001]. par exemple la recherche *sports nautiques* pourra apporter les résultats étant annotés par le concept ayant pour terme *sports nautiques* mais aussi celui ayant pour terme *planche à voile* ou encore celui ayant pour terme *water-skiing* (en considérant que le premier concept est décrit comme plus général que les deux derniers dans le SOC).

3.5.2 L'interopérabilité et l'intégration de données

Les SOC peuvent également servir à centraliser, fédérer et partager des données potentiellement hétérogènes venant de plusieurs sources. Détaillons deux approches différentes d'utilisation de ces référentiels que sont l'interopérabilité sémantique et l'intégration de données. La première vise le partage de données inter-systèmes d'information sur la base d'un contexte commun d'interprétation : le SOC. La seconde vise à regrouper de l'information dans un ensemble homogène, pour faire de l'analyse de données par exemple.

L'interopérabilité.

Les SOC fournissent à une communauté d'utilisateurs une conceptualisation partagée d'une partie spécifique du monde, dans le but de faciliter la communication

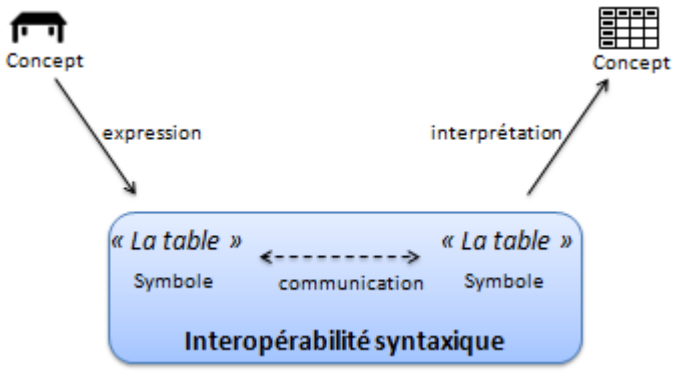

FIGURE 3.8 – Interopérabilité syntaxique. Les deux systèmes peuvent échanger l'information « La table » dans un même langage. Ce niveau ne garantit pas la non ambigüité de ce symbole et conduit dans cet exemple à une interprétation différente entre les deux systèmes.

efficiente d'une connaissance complexe. Ce besoin d'échange d'informations et de partage d'une conceptualisation traduisent la notion d'interopérabilité que P. Miller définit par : « process of ensuring that the systems, procedures and culture of an organisation are managed in such a way as to maximise opportunities for exchange and re-use of information, whether internally or externally » [Miller 2000]. L'interopérabilité en Ingénierie des Connaissances peut être vue comme la capacité de deux personnes ou systèmes d'informations à communiquer des informations (interopérabilité syntaxique) et à partager un contexte commun pour interpréter cette information de manière non ambigüe et similaire (interopérabilité sémantique) [Degoulet 1997].

Considérons ces deux niveaux d'interopérabilité :

- **L'interopérabilité syntaxique** en Ingénierie des Connaissances a pour objectif de permettre à deux systèmes d'échanger de l'information en utilisant un même langage. La figure 3.8 illustre ce niveau d'interopérabilité entre deux systèmes échangeant le symbole (ou terme) *La table*. Ce premier niveau concerne le format de représentation des connaissances. Le fait qu'une ressource soit exprimée ou puisse être traduite sous un format **standardisé ou normé** conduit vers l'interopérabilité syntaxique. Toutefois ce niveau est insuffisant : il ne garantit pas que les deux systèmes interpréteront cette information de manière non ambigüe et similaire : dans notre illustration, il y a deux interprétations différentes du symbole *La table* faisant référence au mobilier et à un élément d'une base de données.
- **L'interopérabilité sémantique** a pour objectif que les parties communicantes aient une compréhension commune de la signification des informations

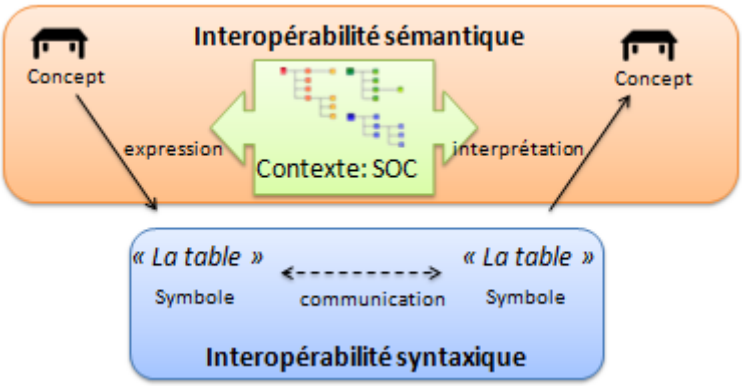

FIGURE 3.9 – Interopérabilité sémantique.

qu'elles échangent [Heiler 1995]. La Figure 3.9 illustre ce niveau d'interopérabilité et garantit que les systèmes communicants interprètent le symbole *La table* de la même manière comme faisant référence au mobilier.

L'utilisation de référentiels communs (SOC) fournit un contexte commun d'expression et d'interprétation, et contribue ainsi à l'interopérabilité sémantique.

L'intégration de données.

L'intégration de données peut se faire de plusieurs manières. M. Hacid et C. Reynaud [Hacid 2004] distinguent deux approches :

l'approche distribuée (approche médiateur). Cette approche utilise un système de médiation. Le système de médiation offre à l'utilisateur une vue uniforme et centralisée des données distribuées. Cette vue « peut aussi correspondre à une vision plus abstraite, condensée et qualitative des données et donc, plus signifiante pour l'utilisateur » précisent P. Laublet *et al.* [Laublet 2002]. Le projet DebugIT [Schober 2010] utilise cette approche distribuée pour interroger des serveurs de données cliniques de différents hôpitaux Européens. Dans ce projet, le système de médiation est lui-même partiellement distribué pour filtrer et sécuriser les données de chaque hôpital ;

l'approche centralisée (entrepôt de données). Cette approche consiste à définir un schéma qui permet l'intégration des données. Au préalable, les données sources sont transformées pour se conformer au schéma défini. L'utilisateur pose ses requêtes dans les termes du vocabulaire structuré fourni par le schéma. Le projet LexGrid (*cf.* section 4.5.3) utilise ce type d'approche.

Nous reviendrons en section 5.2.1.1 sur l'approche utilisée dans nos travaux.

3.5.3 L'aide à la décision et le raisonnement

Depuis les terminologies les plus simples permettant de raisonner (par exemple, dont la structure est limitée à une hiérarchie de subsomption) jusqu'aux ontologies riches en descriptions formelles, les SOC offrent une représentation de la connaissance d'un domaine avec une **expression formelle** pouvant être **interprétée par les machines**.

L'aide à la décision.

En plus des outils classiques de fouille des données, la formalisation de l'information améliore l'aide à la décision. Grâce au typage explicite et au raisonnement sur les heuristiques du SOC, le système peut mettre en évidence des corrélations ou des alertes qui aident à la prise de décision. En médecine, nous pouvons citer l'exemple de détection d'interaction médicamenteuse [Amardeilh 2009] pour la pharmacovigilance.

Le raisonnement.

A partir des axiomes formels d'une ontologie de domaine, certaines informations peuvent être déduites automatiquement et permettre, en prenant l'exemple d'une ontologie des urgences médicales, de déduire qu'un patient atteint d'une appendicite est susceptible de contracter une péritonite. Des moteurs de raisonnement permettent aussi de valider la conformité d'une information au regard de règles formellement décrites dans le référentiel. Par exemple, un système peut nous renseigner sur l'applicabilité d'un article de loi à un individu en fonction de ses propriétés et des contraintes définies.

3.6 Le processus éditorial des SOC

La problématique générale de notre travail de thèse a pour contexte la représentation de SOC. Ce contexte est lié au processus éditorial des SOC. Comprendre les actions qui permettent d'**éditer, publier et utiliser** ces référentiels nous permettra de mieux répondre aux attentes formulées quant à nos travaux (*cf.* chapitre 5). Grâce à l'expérience de l'entreprise MONDECA dans la gestion de SOC, nous avons identifié avec l'aide de J. Delahousse les différentes activités et actions de l'élaboration jusqu'à l'utilisation de ces référentiels. Détaillons maintenant les activités de ce processus présenté en figure 3.10 :

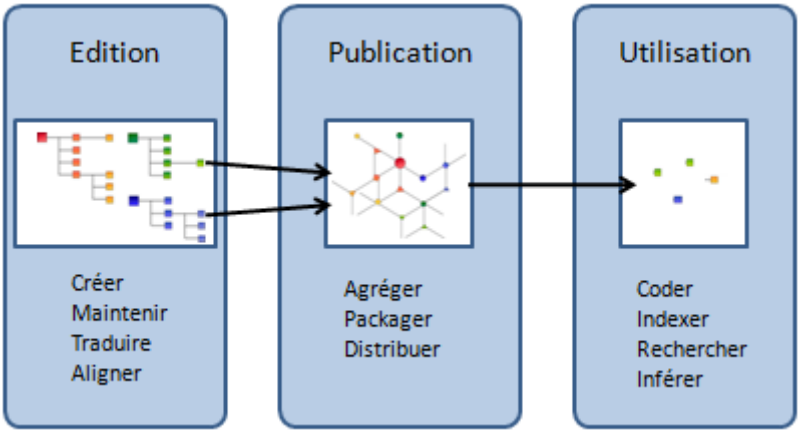

FIGURE 3.10 – Processus éditorial de l'élaboration jusqu'à l'utilisation des Systèmes d'Organisation de la Connaissance.

La première activité d'édition met en jeu des experts du domaine modélisé, des spécialistes de la représentation de la connaissance et des outils d'édition de SOC. Cette activité comprend notamment l'action de création qui aboutit à l'élaboration de la SOC mais également l'action de maintenance qui permet de faire évoluer les connaissances qu'elle représente. D'autres actions plus spécifiques peuvent intervenir dans cette activité. C'est le cas de la traduction qui fait intervenir des traducteurs grâce auxquels la ressource est multilingue et favorise son utilisation mais aussi la recherche d'information dans différentes langues. Une autre action spécifique est l'alignement de plusieurs SOC et met en jeu des experts ainsi que des outils et méthodes de mise en correspondance de concepts. Cette activité d'édition doit supporter un environnement multi-terminologique et multi-lingue.

La seconde activité de publication met en jeu les fournisseurs de SOC et a pour objectif la mise à disposition de tout ou partie de ces ressources au travers de fichiers ou de services d'accès. Cette activité comprend l'action d'agrégation et de packaging, c'est à dire la mise à disposition d'un sous-ensemble ou de la totalité du contenu du serveur de terminologie. C'est également lors de cette activité que des filtres sur des versions, des transformations vers des formats d'échange peuvent être réalisées. Enfin l'action de mise à disposition propose des méthodes d'accès à l'information.

La troisième activité d'utilisation comprend les actions et applications telles que le codage, l'indexation, la recherche d'information ou encore l'inférence fondées sur les SOC. Cette activité est intimement liée à celle de publication pour l'accès aux données des structurations.

3.7 Synthèse et discussion

La place des SOC et de leurs correspondances est très importante pour pallier l'ambiguïté d'interprétation, favoriser l'échange d'une information univoque et ainsi contribuer de manière plus générale à l'**interopérabilité sémantique**. Alors que cette fonction n'est plus remise en question, la distinction entre les différents types de Systèmes d'Organisation de la Connaissance est de plus en plus floue, ce qui est **source d'ambiguïté** [Ingenerf 1998]. Il existe ainsi des terminologies plus ou moins formelles, des thésaurus avec une structure plus ou moins complexe, des taxonomies utilisées en dehors du domaine des êtres vivants, des ontologies avec ou sans contraintes sur leurs relations. Le classement d'une ressource dans un type particulier de structuration est une tâche ardue, en témoignent les critiques de certaines ressources qui prétendent être ce qu'elles ne sont pas [Oltramari 2002, Slodzian 2000].

Cette propension à confondre les types a certainement été accentuée par son utilisation dans des disciplines et des domaines divers, particulièrement depuis l'apparition des outils informatiques. Comme le souligne L. Wittgenstein, **si l'on veut connaître le sens d'un mot, il faut regarder quel en est son usage** [Wittgenstein 1953]. De même pour un SOC, il est plus important de comprendre quelles utilisations il peut rendre. Est-ce un SOC construit pour aider la recherche d'information textuelle ? Sert-il grâce à sa nature formelle à effectuer des raisonnements logiques ? Ces questions rejoignent le besoin de définir clairement les prétentions du SOC que l'on construit et se rapprochent de la méthode des questions de compétences [Uschold 1996].

Un caractère commun aux exemples de SOC que nous avons détaillés est leur capacité à **gérer le langage naturel**. Dans l'absolu, une conceptualisation d'un domaine pourrait ne pas contenir d'empreinte linguistique et ne représenter que des relations et contraintes sur des concepts. Toutefois cette perspective rendrait impossible la maintenance et l'utilisation par un humain. Les symboles du langage naturel sont nécessaires à tous types de SOC. La plupart des SOC proposent des relations qui permettent d'organiser leurs concepts dans une **hiérarchie** ou d'ajouter des liens sémantiques entre eux. La profondeur de ces hiérarchies varie et dépend principalement du niveau de granularité voulu. En excluant LOINC qui n'est pas hiérarchique, la profondeur de la hiérarchie varie entre 5 pour Eurovoc et 11 pour le MeSH. Certains SOC proposent d'autres relations sémantiques comme la SNOMED 3.5 : « est décrit par » ou encore la CIM-10 : « est exclu ». Toutefois, ces relations sont **spécifiques** au domaine et au SOC dans lequel elles sont définies.

Chapitre 4

Des moyens pour représenter, échanger et accéder aux SOC

Sommaire

Ce chapitre propose un état de l'art sur les systèmes, langages, standards et projets existants pour représenter et manipuler des SOC. Les travaux présentés dans ce mémoire s'inspirent et utilisent la modélisation de ces initiatives avec lesquelles ils essayent d'être conformes pour prétendre favoriser l'interopérabilité. Nos travaux (i) doivent être capables de communiquer avec les standards d'interopérabilité ; (ii) s'appuient sur les langages de représentation généralistes pour définir notre modèle ; (iii) s'inspirent, utilisent et étendent des langages spécialisés dans la représentation de SOC ; (iv) définissent des services d'accès aux connaissances des SOC en se fondant sur la littérature ; (v) se positionnent vis-à-vis des projets existants.

4.1 Les principaux standards d'interopérabilité en santé

Des travaux sont en cours pour construire et pour diffuser des normes de représentation, d'échange des informations entre systèmes informatisés qui reposent sur des SOC. En santé, les principaux organismes influents au niveau international sont : CEN 13606 (Européen), OpenEHR (Australien), HL7 et IHE (Internationaux). Nous détaillerons uniquement HL7 et IHE avec lesquels nous interagirons dans nos travaux (*cf.* section 8.2).

4.1.1 HL7

Health Level Seven[1] est une organisation accréditée ANSI et ISO. Issue d'une initiative américaine (1987), son objectif est de créer des spécifications flexibles et peu coûteuses, des guides de bonnes pratiques et des méthodologies qui permettent le partage des données patients entre systèmes d'informations hospitaliers. HL7 version 3 (initiée en 1997) définit la structure et le rôle des messages entre applications de ces systèmes grâce au modèle formel de référence (RIM) et à une architecture de documents cliniques (CDA).

– **Le RIM** (Reference Information Model) est l'élément de structuration conceptuelle des modèles standards proposés par les groupes de travail d'HL7. Il four-

1. Le chiffre « 7 » (HL7) fait référence non pas à une septième version mais à la septième couche applicative du modèle OSI (Voir http://fr.wikipedia.org/wiki/Mod%C3%A8le_OSI). Voir http://www.hl7.org/

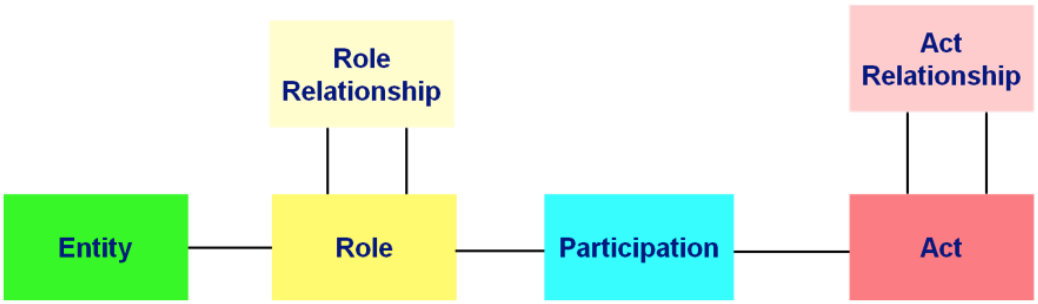

FIGURE 4.1 – Les six classes de base du Modèle Conceptuel de Référence (RIM) d'HL7. Une *Entity* représente une personne, un lieu, etc. Le *Role* inclut le fait d'être patient, d'être un lieu de traitement, d'être un spécimen, etc. Un *Role relationship* connecte plusieurs rôles. l'entité *Participation* relie des rôles aux actes. Un *Act* représente la prescription, l'observation, l'opération, etc. Ces actes sont reliés par un connecteur : *Act Relationship*.

nit une vue statique des besoins de modélisation d'information et il permet de s'assurer de la cohérence des modélisations entre les domaines médicaux. Le RIM peut être utilisé pour décrire la structure logique des données dans les documents échangés. Son modèle s'articule autour de 6 classes principales illustrées en figure 4.1. La multitude et la difficulté d'interprétation formelle des modèles de la proposition HL7 en font un standard difficile à mettre en place au sein d'un système d'information hospitalier et reste peu utilisé à ce jour.

– **La CDA** (Clinical Document Architecture) [Dolin 2006] est un standard ANSI depuis Mai 2005. Il s'agit d'un marquage qui spécifie, dans une perspective d'échange, la structure et la sémantique des documents cliniques. Ce standard est compatible avec le RIM. La CDA est retenu pour structurer les documents du Dossier Médical Partagé en France [2].

La version 3 de HL7 s'intéresse largement à la place prépondérante des SOC pour l'échange d'information. C'est dans ce contexte que HL7 a établi et fait évoluer un ensemble de services cohérents pour les SOC nommé Common Terminology Services 2 (CTS 2) que nous détaillerons en Section 4.4.2.

4.1.2 IHE

Initiée en 1997 par des professionnels de la santé, Integrating the Healthcare Enterprise (IHE) est destinée à améliorer la façon selon laquelle les logiciels du domaine

2. Voir : http://bit.ly/reH6LX

```
MSH|^~\&|OF|LabSystem|OP||20050205094510||
MFN^M08^MFN_M08|2106|2.5|||NLD|8859/1|NL|
MFI|OMA|OF_OMA_NL_1.2|REP|||ER|
MFE|MAD|1846||1846^CREABL/Creatinine^L|CE|
3320  OM1|1|1846^CREABL/Creatinine^L|NM|Y|K231^Klinisch Chemisch
Laboratorium^L|||Creatinine|||||||||A|
OM2|1|umol/l|6.0||
OM4|1||stolbuis rode dop4||ml|BLDV^volbloed^HL70487|
MFE|MAD|1848||1848^CREAUV/Creatinine^L|CE|
OM1|2|1848^CREAUV/Creatinine^L|NM|Y|K231^Klinisch Chemisch
Laboratorium^L|||Creatinine|||||||||A|
OM2|2|mmol/l|6.0||
OM4|2||24-uurs bokaal|||UR^urine^HL70487|
```

FIGURE 4.2 – Exemple de transaction LAB-51 du profil LCSD d'IHE. Cette transaction illustre, en langue néerlandaise, la description des analyses biomédicales possibles à tester.

échangent leurs informations. Pour cela, IHE développe des profils d'échange de documents et d'informations qui reposent sur des standards établis comme HL7. Par exemple, le profil Laboratory Code Sets Distribution (LCSD) propose une spécification pour le partage de SOC entre systèmes de laboratoires qui utilise le modèle RIM de HL7. Plus précisément ce profil définit un cas d'utilisation et une transaction.

- Cas d'utilisation : le serveur de terminologies (Code Set Master Actor) qui contient les SOC utilisés par les laboratoires (par exemple LOINC) envoie les SOC en entier aux systèmes d'information des laboratoires demandeurs (Code Set Consumer Actor) dès qu'une nouvelle version est disponible.
- Transaction : le profil d'intégration prévoit une transaction pour ce cas d'utilisation appelé « LAB-51 ». Cette transaction repose sur le standard de messages (MFN) de la version 2.5.1 de HL7. Un exemple de transaction est donné en figure 4.2. Nous reviendrons sur l'utilisation de ce profil dans nos travaux en section 8.2.

4.1.3 Synthèse et discussion

Les principaux standards d'interopérabilité que nous avons étudiés ne concernent pas directement la représentation de SOC mais les utilisent dans la circulation de l'information. Le domaine de santé s'illustre ici par la maturité des systèmes pour lesquels des standards **spécifient des utilisations possibles de SOC**. Nos travaux doivent pouvoir s'intégrer à ces systèmes, c'est-à-dire fournir les services nécessaires à l'utilisation des connaissances contenues dans les SOC pour supporter la transmission d'information dans un format standard. Cette intégration sera illustrée dans le projet AnaBio en section 8.2.

4.2 Les langages généralistes de description de connaissances

Dans cette section, nous présentons les langages généralistes de description de connaissances qui sont d'intérêt pour nos travaux. Ces langages nous sont utiles pour l'expression d'un modèle général de représentation de SOC. La correspondance avec une sémantique formelle et la compatibilité avec le Web sémantique sont pour nous des critères importants.

4.2.1 Présentation

La syntaxe des langages dont les expressions sont destinées à être manipulées par une machine est en général définie à partir de règles précises établies par une **grammaire formelle** : le langage est alors qualifié de **formel** (langage de programmation, langage XML, etc.). Un langage formel dont les expressions sont destinées à être interprétées en plus par l'homme est qualifié dans ce mémoire de **langage de description de connaissances**. Un tel langage permet de décrire l'état d'un domaine abstrait ou concret (par exemple, une configuration du monde à un certain moment). Pour établir une telle description, on combine entre eux les composants élémentaires du langage (appelés symboles) de telle sorte que cette combinaison (i) respecte les règles de la grammaire formelle et (ii) « ait du sens » : c'est-à-dire qu'elle décrive une configuration possible du domaine. Déterminer le sens associé à une expression d'un tel langage est périlleux tant que le sens accordé à ses symboles demeure ambigu ou subjectif car il peut être interprété de manières différentes voire contradictoires selon les individus (*cf.* section 3.5.2). Pour résoudre ce problème, le sens des symboles et de leur combinaison est établi dans un autre langage formel : la **sémantique formelle**. Formaliser le sens d'un langage consiste (i) à associer à chacune de ses expressions l'ensemble des configurations du domaine qu'elle représente et (ii) à mettre en évidence les **rapports sémantiques** qui existent entre deux expressions données (*e.g.* rapports de subsomption, de déduction, de conséquence sémantique, de spécialisation/généralisation, etc.). En cas de doute sur le sens d'une telle expression ou le rapport qu'elle entretient avec une autre, la sémantique formelle fait autorité pour lever l'ambiguïté. Une sémantique formelle est en général exprimée dans un langage déclaratif mathématique : sémantique ensembliste (*e.g.* la théorie des ensembles) ou logique : sémantique logique (*e.g.* la logique du premier ordre). Ces langages sont sans doute les meilleurs candidats pour sa définition car ils détiennent un fort pouvoir d'expression, permettent d'établir des démonstrations rigoureuses (généralement pour prouver que deux expressions entretiennent un certain

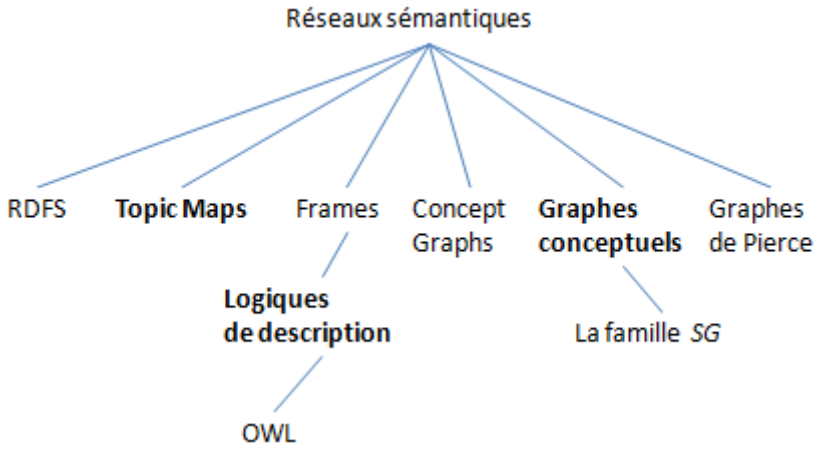

FIGURE 4.3 – Généalogie des principaux formalismes de représentation par réseaux sémantiques de connaissances.

rapport sémantique) et ont été utilisés et étudiés depuis plusieurs siècles par la communauté scientifique comme nous le verrons dans les sections suivantes. Lorsqu'un langage est « **sémantisé** » ils est plus qualifié pour représenter les connaissances mais pas suffisamment pour permettre un raisonnement automatique. En effet, raisonner dans un langage sémantisé consiste à déterminer les rapports sémantiques qui existent entre deux expressions données de ce langage ; le raisonnement étant qualifié d'automatique lorsqu'il est mené sans intervention humaine. Certains langages sont reconnus pour fournir des méthodes de raisonnement adaptées à son expressivité et à la structure des composants de sa syntaxe : on dit alors que le langage est « **opérationnalisé** ». Dans ce mémoire, les langages de description des connaissances qui sont à la fois sémantisés et opérationnalisés, sont appelés langages de **Représentation des Connaissances et Raisonnements** (RCR).

En général, la syntaxe adoptée par les langages de description de connaissances est relativement proche du modèle entité-relation et peut facilement être retranscrite sous forme de graphe. Associer une représentation visuelle à ces langages a pour intérêt de rendre la phase de modélisation plus intuitive et moins laborieuse pour l'utilisateur. Cela explique sans doute qu'au cours du siècle précédent, un certain nombre de formalismes représentant les informations sous une forme réseau aient vu le jour (*cf.* figure 4.3).

Les réseaux sémantiques sont apparus en 1909 avec les premiers graphes pourvus d'une interprétation et d'heuristiques logiques développés par C.S. Pierce. En 1975, apparurent les frames et les scripts [Minsky 1974] qui sont une notation plus formelle des réseaux sémantiques et qui conduiront plus tard à la définition des logiques de

TABLEAU 4.1 – Langages généralistes de représentation de connaissances. Par exemple, le langage OWL possède, pour son fragment DL, une sémantique formelle en théorie des modèles (sémantique ensembliste) mais ne prévoit pas d'algorithme de raisonnement. Pour effectuer des raisonnements sur le langage OWL DL, il est possible de l'opérationnaliser par les logiques de description.

Langage formel	**Sémantique formelle**	**Algorithme de raisonnement**	**Opérationnalise le langage**
Topic Maps	-	Non	-
Graphes Conceptuels	Logique des prédicats	Projection et ses dérivés	Topics Maps ; RDF/S
Logiques de description	Théorie des modèles	Méthode des tableaux ; Automates finis	OWL DL
RDF/S	Théorie des modèles	Non	-
OWL	Théorie des modèles pour OWL DL	Non	-

description [Baader 2003]. En 1984, J. F. Sowa s'inspire en particulier des graphes de Peirce pour définir le formalisme des graphes conceptuels [Sowa 1984] et le munir d'une sémantique logique (similaire à celle des graphes existentiels). Parallèlement aux réflexions autour d'un Web sémantique, sont nés deux langages de description de ressources sur le Web : RDF/S [3] et les Topic Maps. En 2003, un nouveau langage plus expressif que RDF-S apparaît : le langage OWL. Les langages RDF/S et OWL sont munis d'une sémantique formelle définie en théorie des modèles [Hayes 2004, Patel-Schneider 2004], tandis que le formalisme des Topic Maps n'en dispose pas.

Les langages proposés pour le Web (RDF/S, OWL et Topic Maps) n'étant pas opérationnalisés, ils ne fournissent pas les algorithmes de raisonnement nécessaires au développement des services automatisés « intelligents » tels qu'attendus dans la vision du Web Sémantique de T. Berners-Lee [Berners-Lee 2001] (*cf.* section 2.2). En effet, de tels langages nécessitent d'être opérationnalisés par des langages de RCR. L'opérationnalisation d'un langage par un autre peut être réalisée de différentes manières, dont une consiste à traduire les expressions du langage en celles de l'autre tout en préservant leur sémantique formelle. Le tableau 4.1 présente les principaux langages de description des connaissances en relation avec nos travaux, avec leur sémantique formelle et les langages qu'ils permettent d'opérationnaliser.

3. Nous notons RDF/S les langages RDF et RDF-Schema.

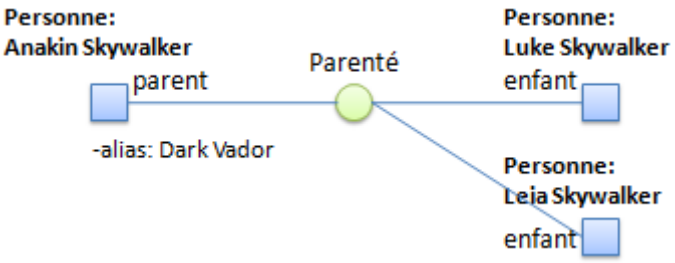

FIGURE 4.4 – Exemple de topic map pour l'assertion « Anakin Skywalker (alias Dark Vador) est le père de Luke Skywalker et de Leia Skywalker ».

4.2.2 Langages du web sémantique

4.2.2.1 Topic Maps

Le concept de « topic maps » (en français : carte topique) est apparu au début des années 90 dans les secteurs documentaire et terminologique. Il a été développé au cours d'un projet de création d'index et de glossaires à partir de documents dispersés sur le Web. Un consortium international animé par M. Biezunski et S. R. Newcomb se met alors en place pour formaliser ce concept. Au début des années 2000, au même moment que la création du standard RDF (*cf.* section 4.2.2.2), il propose une syntaxe concrète en XML nommée XML Topic Maps (XTM) [Pepper 2001, Park 2003] qui sera adoptée sous la forme d'une norme ISO [Biezunski 1999].

Une topic map n'est pas limitée à la représentation de connaissances des domaines documentaire et terminologique mais a un usage plus général : elle est définie pour décrire toute idée pouvant être l'objet d'un discours sous une forme suffisamment intuitive et structurée pour être exploitable à la fois par l'homme et la machine. Une topic map est composée de *Topics* caractérisés par des *Occurrences* et des *Associations* qui relient ces topics entre eux. Les topics et les associations sont typés. Une association identifie le rôle joué par chacun des topics qui y participe. Deux associations spécifiques sont normalisées pour toute TM : l'association superclasse-sousclasse et l'association classe-instance. Une occurrence est un lien depuis un topic vers une ressource sur son sujet (le sujet est ce que le topic essaie de représenter formellement) et peut être de types document texte, image, etc.

Un exemple de topic map est illustré [4] en figure 4.4.

4. Attention cet exemple révèle une information capitale au sujet de la saga cinématographique Star Wars. Il est déconseillé de le lire avant d'avoir visionné la saga en entier.

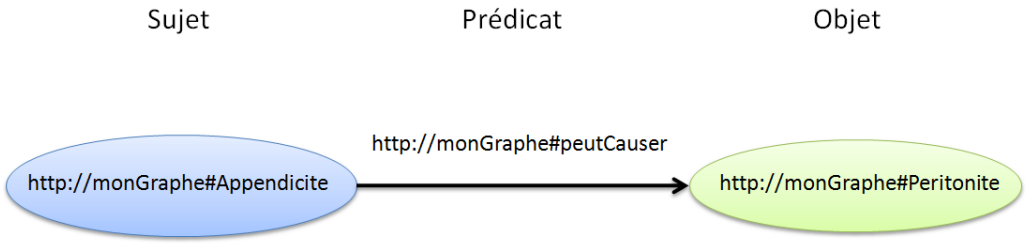

FIGURE 4.5 – Exemple d'un triplet RDF qui décrit les ressources Appendicite et Péritonite, et la relation « peut causer » qui les relie.

4.2.2.2 RDF/S

Le langage RDF [Lassila 1999] est l'aboutissement d'une initiative de formalisation des métadonnées proposée en 1994 par T. Berners-Lee. Il s'appuie sur une syntaxe XML qui peut aussi être considérée comme un format de sérialisation de graphe. Un document RDF est un ensemble de triplets (sujet, prédicat, objet) qui représentent une ressource (le sujet) en attribuant une valeur (l'objet) à une de ses propriétés (le prédicat). Les triplets RDF s'inspirent de la structure sujet-verbe-objet des expressions les plus simples du langage naturel. La figure 4.5 présente le graphe RDF correspondant à l'éventualité qu'une « appendicite puisse causer une péritonite ». Dans cet exemple, le sujet représente la maladie *Appendicite*, le prédicat la relation *peutCauser* et l'objet la maladie *Péritonite*. Une relation dite de subsomption peut être définie entre deux documents RDF. Lorsqu'elle est établie telle que a subsume b, cela signifie que la situation représentée par b est un cas particulier de celle représentée par a.

Il peut être nécessaire d'organiser le vocabulaire employé dans un document RDF sous la forme d'une ontologie rudimentaire. Pour ce faire, on dispose du langage RDFS [Brickley 2004] qui permet de définir des classes et des propriétés avec leurs domaine et co-domaine ; et de les structurer (en sous-classes et sous-propriétés) en utilisant une terminologie normalisée de primitives (les classes Class, Property, etc, et les propriétés type, subClassOf, subPropertyOf, etc). Les langages RDF/S disposent d'une sémantique formelle définie en théorie des modèles [Hayes 2004] qui donne l'interprétation mathématique à accorder à un document RDF et à la subsomption. Les langages RDF/S présentent l'avantage d'être intuitifs, pragmatiques et synthétiques mais peuvent parfois être limités. Il est par exemple impossible d'exprimer une disjonction de classes, des restrictions sur le domaine ou le co-domaine,

une restriction de cardinalité ou encore d'attribuer des définitions d'ordre axiomatique aux prédicats (transitivité, etc.) [McBride 2004].

4.2.2.3 OWL

En 2003, un nouveau langage plus expressif que RDFS apparaît : le langage OWL. Il se situe dans la continuité du langage DAML+OIL [McGuinness 2002] et est issu des dernières recherches en matière de logique de description. OWL se présente sous la forme d'une extension au langage RDFS et introduit de nouvelles primitives (allValuesFrom, minCardinality, etc) pour structurer avec plus de finesse le vocabulaire utilisé lors de la modélisation. En OWL on déclare des axiomes pour définir des individus, des classes et des propriétés dans ces classes. Les individus représentent les entités du monde, une classe, un ensemble d'entités et une propriété, un ensemble de couples d'entités entretenant un rapport précis.

En OWL, on s'intéresse essentiellement à deux problèmes de raisonnement :

- La classification d'un individu dans une classe : un individu est classifié dans une classe si et seulement si l'entité qu'il représente dans le domaine appartient à l'ensemble des entités que représente la classe.
- La subsomption d'une classe par une autre : Comme pour RDF, OWL introduit une relation de subsomption. On dit qu'une classe subsume une autre si et seulement si l'ensemble des entités représentées par la première contient l'ensemble des entités représentées par la seconde.

OWL est structuré en trois fragments :

- **OWL Full** est le langage complet. Son inconvénient principal vient de son pouvoir d'expression élevé, qui le rend indécidable. Il permet notamment de considérer un élément de l'ontologie à la fois comme une classe et comme une instance.
- **OWL-DL** (OWL Description Logic) restreint l'expressivité du langage au bénéfice de sa décidabilité : ce qui rend possible des raisonnements automatisés. Dans ce fragment, chaque élément de l'ontologie ne peut être défini à la fois comme une classe et comme une instance. OWL-DL tient son nom des logiques de description (LD) [Nardi 2003] avec lesquels il entretient un rapport étroit puisque d'une part OWL-DL dispose d'une sémantique formelle définie en théorie des modèles [Patel-Schneider 2004] qui est très proche de celle des LD et d'autre part il peut être traduit en une LD tout en préservant sa sémantique formelle. De ce fait, les méthodes de raisonnement définies en LD peuvent être utilisées pour raisonner en OWL-DL (*i.e.* les LD opérationnalisent OWL-DL).
- **OWL-Lite** est un langage inclus dans OWL-DL dont l'expressivité a été en-

core plus réduite au bénéfice de davantage de performances durant le raisonnement. En OWL-Lite, il est impossible de déclarer des restrictions de cardinalité en utilisant des valeurs au dessus de 1, d'établir une disjonction de classes ou bien de définir des classes à partir d'une union de classes, etc.

Il y a une stricte inclusion entre ces trois langages : toute ontologie OWL Lite (respectivement OWL-DL) valide est une ontologie OWL DL (respectivement OWL-Full) valide. De manière analogue, les rapports sémantiques (subsomption/classification) établis dans une ontologie OWL Lite (respectivement OWL-DL) sont aussi valables dans une ontologie OWL DL (respectivement OWL-Full).

La version 2 de OWL parue récemment [Motik 2009] définit un certain nombre de profils qui sont des sous-langages inclus dans OWL-DL. Parmi ces profils, on distingue **OWL2 EL** [5] qui a été défini dans le but d'offrir des raisonnements très performants sur des ontologies volumineuses. En particulier, le passage à l'échelle a été validé en réalisant des tests sur la SNOMED CT.

4.2.2.4 OBO

Open Biological Ontologies (OBO) est un langage de représentation d'ontologies pour les domaines de la biologie et le biomédical. Ce langage est accompagné d'un format d'échange *OBO Flat File Format* qui est dérivé du format de représentation de l'ontologie Gene Ontology (GO) afin de produire une syntaxe qui soit facile à lire (par un humain), commode à parser et qui présente un minimum de redondance. Autour de ce langage, s'articule une communauté d'utilisateurs avec des outils comme le OBO-Edit, ou encore un répertoire d'ontologies OBO nommé OBO Foundry. Parallèlement au développement de l'initiative OBO, la communauté du web sémantique a développé le langage OWL. Alors que OWL n'est pas dédié à la représentation d'un domaine en particulier, OBO se consacre au domaine de la biologie et du biomédical. Les différences ne s'arrêtent pas là. Ainsi une distinction majeure tient en leur formalisation. OWL au travers de sa sémantique formelle et de ses correspondances, permet d'automatiser des raisonnements sur une ontologie. La spécification de la syntaxe du langage OBO quant à elle n'est **pas formellement décrite** et se fait au travers du langage naturel. Ceci engendre des ambigüités quant à l'interprétation de la sémantique d'une ontologie en OBO et donc des choix d'implémentation des outils associés.

Ces dernières années, un **effort de réconciliation** a été initié avec la construction de transformations entre les deux systèmes fondés sur une interprétation commune et formelle de la sémantique d'OBO [Golbreich 2007, Tirmizi 2011].

5. Voir : http://www.w3.org/TR/owl2-profiles/#OWL_2_EL

4.2.3 Langages de Représentation des Connaissances et Raisonnements

4.2.3.1 Graphes conceptuels

Le formalisme des graphes conceptuels (introduit en 1984 par J.F. Sowa [Sowa 1984]), et plus précisément la famille SG [Baget 2002], permet de représenter les connaissances sous la forme de « graphes » (au sens théorie des graphes).

Un graphe conceptuel est constitué de sommets concepts et de sommets relations typés et reliés entre eux par des arêtes étiquetées [6] ; un sommet concept pouvant représenter une entité du monde identifiée ou non. Les types des sommets concept et relation sont définis sous la forme d'un ordre partiel dans une ontologie rudimentaire nommée support. Le formalisme des graphes conceptuels est muni d'une opération élémentaire de raisonnement appelée projection. Elle consiste à déterminer les portions du graphe cible qui représentent des connaissances plus spécifiques que celles du graphe source.

Le « graphe » conceptuel et la projection permettent de définir des composants plus expressifs tels que les règles, les contraintes positives, les contraintes négatives et des opérations plus complexes d'inférence (pour les règles) et de validation (pour les contraintes). Les règles représentent des connaissances du type « si hypothèse alors conclusion » et sont définies à partir de deux graphes, l'un représentant la partie hypothèse et l'autre la partie conclusion de la règle. Les contraintes positives (respectivement négatives) ont la même forme que celle des règles à ceci près que l'hypothèse et la conclusion sont respectivement qualifiées de condition et d'obligation (respectivement d'interdiction). Leur signification est du type « si la condition est présente alors il doit (respectivement ne doit pas) en être de même pour la partie obligation (respectivement interdiction) ». Le formalisme des graphes conceptuels possède une sémantique formelle en logique des prédicats et les raisonnements définis à partir de la projection sont en accord complet avec cette sémantique logique ; ce qui signifie qu'il existe une projection entre un graphe conceptuel et un autre si et seulement si la traduction logique du premier est la conséquence logique de celle du second.

4.2.3.2 Logiques de description

Le formalisme des logiques de description (LD) s'inscrit dans la lignée des « frames » initialement proposés par Minsky en 1974 [Minsky 1974]. En logique

6. l'étiquette d'une arête permet d'attribuer au concept une position dans la liste des arguments de la relation

de description, on déclare des axiomes pour définir des individus, des concepts et des propriétés sur ces concepts. Les concepts correspondent à des ensembles d'éléments dans un univers donné. Les rôles correspondent aux rapports qu'entretiennent ces éléments deux à deux (et représentent des relations binaires définies sur l'univers donné). Les individus correspondent aux éléments (instances des concepts) de cet univers. Le langage des logiques de description se structure en deux niveaux d'abstraction : **la T-box et la A-box** [Baader 2007] :

- la T-box (T pour terminologique) décrit les connaissances générales d'un domaine et contient les déclarations des primitives conceptuelles organisées en concepts et relations. Ces déclarations décrivent les propriétés des concepts et des rôles ;
- la A-box (A pour assertionnel) décrit les connaissances factuelles d'un domaine et en représente un état précis. Elle contient les déclarations d'individus, instances des concepts qui ont été définies dans la T-box. Plusieurs A-box peuvent être associées à une même T-box.

En logique de description, on s'intéresse essentiellement à deux problèmes :

- La classification : déterminer les concepts de la T-Box qui contiennent un individu donné de la A-Box.
- La subsomption : pour deux classes données déterminer si l'une subsume l'autre (i.e. si l'ensemble représenté par l'une contient l'ensemble représenté par l'autre)

Les logiques de description possèdent une sémantique formelle définie en théorie des modèles (qu'il est possible de transcrire en logique des prédicats) et des méthodes de raisonnements intégrées au langage (par exemple, la méthode dite des « tableaux »), ce qui en fait un langage sémantisé et opérationnalisé.

4.2.4 Synthèse et discussion

Les langages que nous venons d'étudier possèdent des points communs et des distinctions que nous allons détailler.

Les points communs. Les langages de description des connaissances que nous avons présentés permettent la représentation de connaissances dans un format proche du langage naturel. Cette représentation a pour but d'être interprétable par les humains et par les machines. Ils disposent tous, à l'exception des Topic Maps, d'une sémantique formelle.

Les distinctions. Le formalisme des GC dispose d'une sémantique formelle (en logique des prédicats) différente de celle du formalisme des LD, RDF/S et OWL (en théorie des modèles). Toutefois il existe des correspondances entre

certains fragments de la théorie des modèles et de la logique des prédicats pouvant être exploitées afin d'établir des ponts entre ces deux formalismes. Les deux formalismes s'influencent mutuellement : citons par exemple J.F. Baget qui propose une LD inspirée des travaux sur les GC [Baget 2008]. Il existe aussi des travaux ayant permis d'établir des correspondances entre le formalisme des GC et d'autres langages de description des connaissances. [Carloni 2009] et [Baget 2005] proposent respectivement une correspondance entre TM/GC et RDF/GC permettant d'opérationnaliser le premier langage en s'appuyant sur les capacités de raisonnement du second. Le modèle des Topic Maps ne propose pas explicitement de séparation en deux niveaux d'abstraction à la différence des autres langages présentés. Toutefois, l'utilisation pratique des Topic Maps révèle une telle séparation marquée par l'association classe-instance qui divise d'une part les types de topics et d'associations et d'autre part les topics et associations.

Avec la définition d'un modèle pour la représentation de SOC, se pose la question du langage dans lequel il sera exprimé. Pour répondre à cette question nous avons étudié les langages généralistes de représentation de connaissances proches du Web Sémantique avec lequel nous voulons être compatible. Le langage OWL DL présente de nombreux avantages : il possède une sémantique formelle opérationnalisé par les logiques de description ; il est bien outillé et profite d'une forte adoption. Il est possible d'utiliser les avantages décrits ci-dessus pour élaborer un modèle de représentation de SOC. Nous reviendrons sur ce choix en section 6.5.

4.3 Les langages de représentation spécialisés

Dans cette section, nous présentons les langages spécialisés pour la représentation de SOC qui sont d'intérêt pour nos travaux. Un modèle général de représentation de SOC se situe au même niveau d'abstraction que ces langages. Pour chaque langage, nous détaillons les propriétés communément identifiées dans les SOC. Nous nous intéressons particulièrement à l'expression de concepts, de dérivés terminologiques, de regroupements de concepts et de correspondances entre SOC.

4.3.1 SKOS

Simple Knowledge Organisation System (SKOS) désigne un langage qui permet la représentation de systèmes d'organisation de connaissances tels que thésaurus, taxonomies, ou tout autre type de vocabulaire contrôlé ou structuré [Miles 2006].

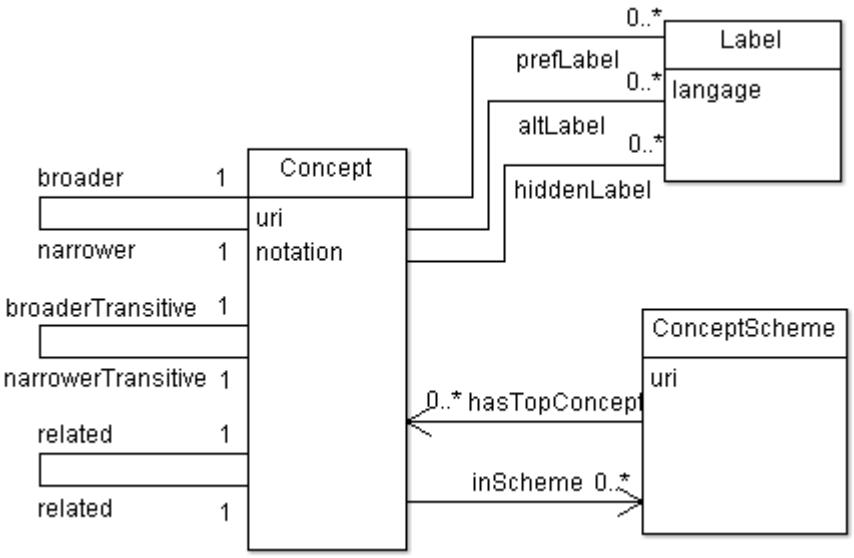

FIGURE 4.6 – Partie termio/conceptuelle simplifiée du modèle SKOS avec son extension eXtended Labels. le modèle est centré autour de la notion de *Concept*. L'extension XL permet de séparer les artefacts terminologiques du concept. Cette séparation permet d'affiner la description linguistique du modèle SKOS.

Ce standard, exprimé en OWL a pour principal objectif de permettre la publication facilitée de vocabulaires structurés pour leur utilisation dans le cadre du Web sémantique. Il met à disposition certaines primitives dédiées à la linguistique : on a d'une part le *Concept* qui représente une notion et d'autre part, pour chaque langue, un terme préféré appelé *prefLabel*, des synonymes nommés *altLabel* et un attribut nommé *hiddenLabel* qui aide à la recherche d'information textuelle en stockant par exemple le libellé préféré avec une faute d'orthographe courante. SKOS est une famille de langages extensibles. L'extension XL (pour eXtended Labels) considère les libellés comme des ressources (on est ici en présence de termes définis indépendamment des concepts), ce qui permet de redéfinir des relations (par exemple une relation de traduction) entre ces *Labels* pour augmenter la richesse linguistique du SOC. Les éléments principaux du langage SKOS et de son extension XL sont illustrés en figure 4.6. Une relation importante dans le langage SKOS est la relation *broader/narrower* qui permet l'expression d'une hiérarchie de concepts et représente indifféremment la notion de partition ou de subsomption. SKOS définit certaines relations comme transitives (*e.g. broaderTransitive*) sur lesquelles des raisonnements peuvent être exécutées.

Le langage SKOS met à disposition trois entités pour représenter des regroupement de concepts. L'entité *ConceptScheme* illustrée en figure 4.6, permet de modéli-

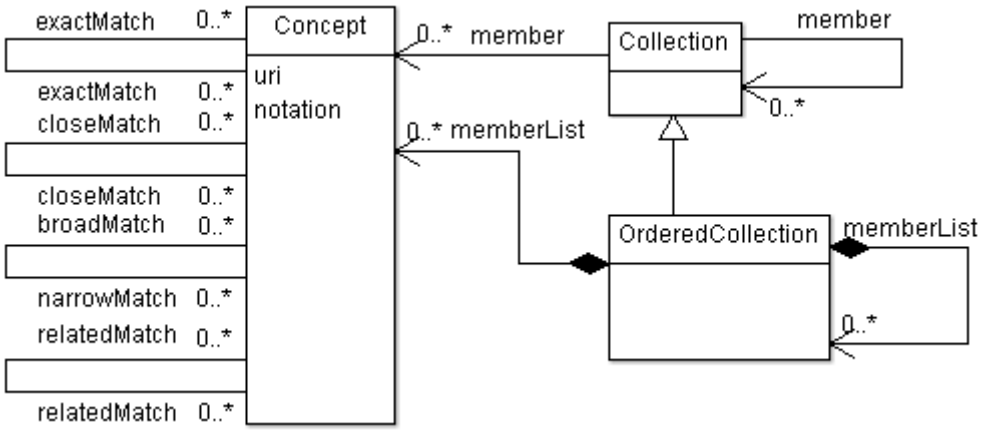

FIGURE 4.7 – Modèle simplifié des groupes et mise en correspondance de SKOS. Les entités *Collection* et *OrderedCollection* représentent un groupe de concepts. Les relations *exactMatch*, *closeMatch*, *broad/narrowMatch* et *relatedMatch* permettent l'expression de correspondances entre concepts.

ser un SOC. Tous les *Concepts* qui font partie d'un *ConceptScheme* déclarent la propriété *inScheme* ayant pour valeur leur SOC d'appartenance. Les entités *Collection* et *OrderedCollection* permettent de représenter des regroupements de concepts ordonnés ou non d'un sous-ensemble de SOC. Les concepts qui les composent peuvent provenir de plusieurs SOC. Ces entités sont illustrées en figure 4.7.

Le langage SKOS permet également la représentation de mise en correspondance de concepts provenant de différents SOC grâce à ses relations de *mapping*. Comme nous avons pu le voir en section 3.2, les correspondances entre SOC sont très importantes pour les applications finales. SKOS propose différents types de correspondances entre concepts illustrés en figure 4.7 : *exact match* pour l'expression de correspondance entre deux concepts interchangeables ; *close match* pour l'expression de deux concepts non interchangeables dont le sens est proche ; *related match* pour exprimer un concept en rapport avec un autre ; *broad match* et *narrow match* pour l'expression de correspondances dont le sens entre deux concepts est plus général ou plus spécifique. Même si cette proposition ne permet pas d'ajouter les informations provenant des alignements (comme le score de confiance), elle constitue une base solide sur laquelle nous nous reposerons (*cf.* section 6.3.2).

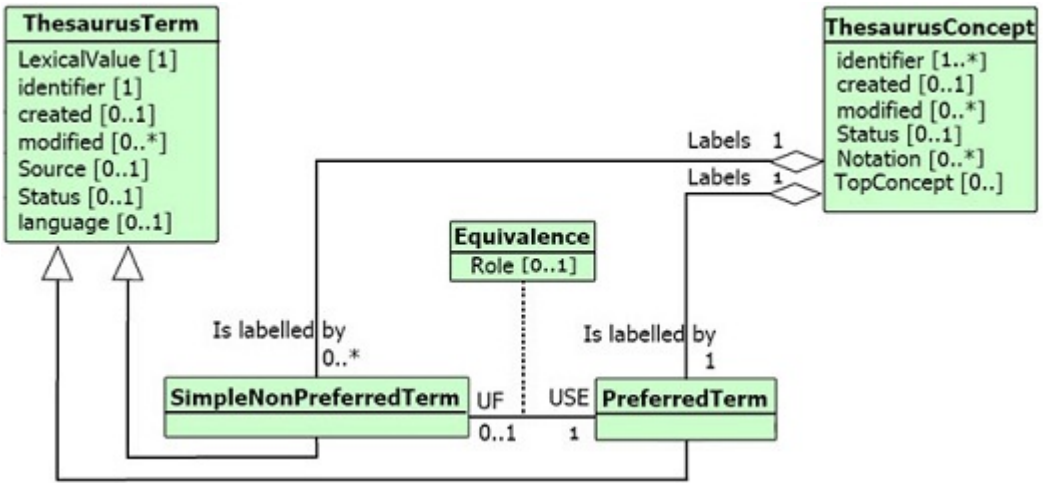

FIGURE 4.8 – Éléments terminologiques du modèle BS 8723. Ce modèle sépare clairement l'aspect conceptuel ne portant aucun élément linguistique et l'aspect terminologique.

4.3.2 BS 8723

Les normes ISO concernant les terminologies sont en train d'évoluer [7] grâce aux travaux de la British Standard et de son projet BS 8723 [BS8723 2008]. La gestion de la linguistique est ici plus fine que dans le standard SKOS comme l'illustre la figure 4.8. Un terme préféré (terme qui est utilisé dans une langue pour désigner un concept) est ici exprimé sous la forme d'une primitive *PreferredTerm*, de même que les termes non préférés (synonymes) *SimpleNonPreferredTerm*. A la différence de libellés (chaînes caractères) apposés sur un concept, cette représentation permet d'exprimer une relation de synonymie directe entre ces éléments terminologiques du modèle.

Ce projet aborde également la problématique de groupement de concepts comme l'illustre la figure 4.9. À cet égard, le modèle BS 8723 met à disposition deux entités *Thesaurus* et *ThesaurusArray*.

L'entité *Thesaurus* représente un SOC composé d'un ensemble de *ThesaurusConcepts* dont l'organisation n'est pas contrôlée par le *Thesaurus* [8] ;

L'entité *ThesaurusArray* représente un groupe de concepts potentiellement or-

7. le projet ISO 25964 va remplacer les normes ISO 2788 relative à l'élaboration et au développement de thésaurus monolingues et ISO 5964 relative à l'élaboration et au développement de thésaurus multilingues. Cette évolution se fonde sur les travaux de la BS 8723.

8. L'organisation hiérarchique du *Thesaurus* n'est pas définie sur cet élément mais est déterminée par l'organisation des *ThesaurusConcepts* entre eux. Ceci implique une faible plasticité de ce modèle et de la redéfinition locale (pour un sous-ensemble du *Thesaurus*) de la hiérarchie principale.

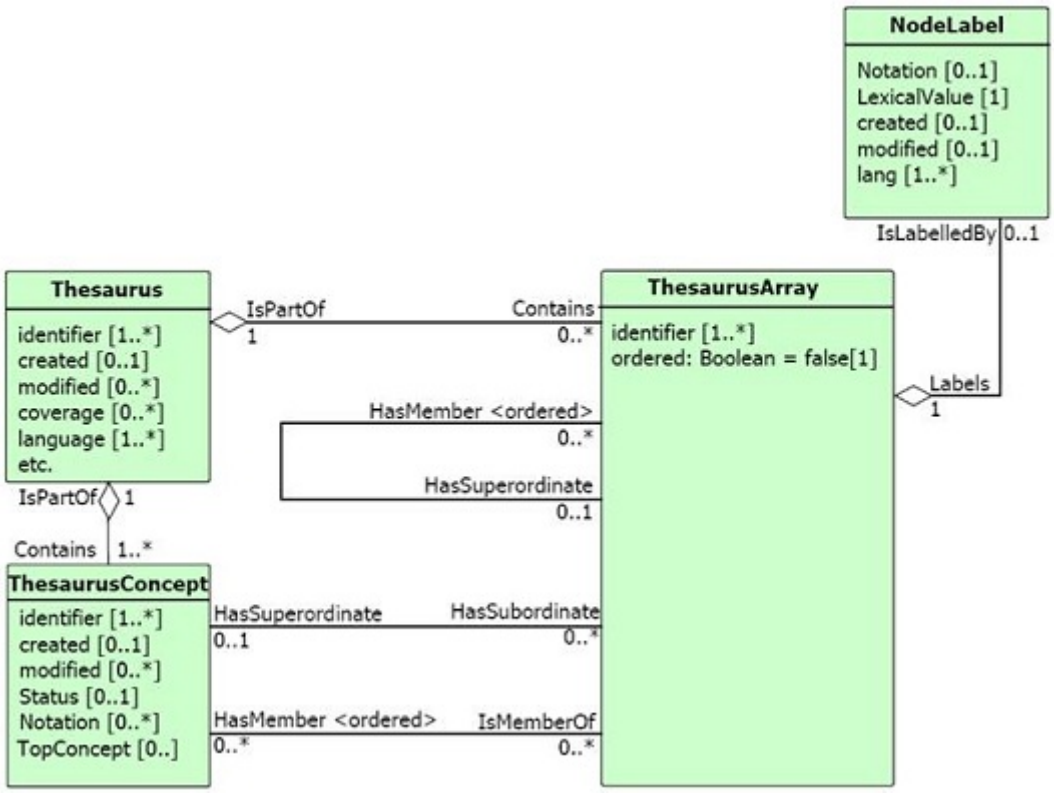

FIGURE 4.9 – Éléments pour le groupement de concepts du modèle BS 8723. Les entités *Thesaurus* et *ThesaurusArray* permettent de représenter respectivement un SOC et un sous-ensemble de concepts ordonnés d'un SOC.

donnés. Ce groupe ne peut appartenir qu'à un seul *Thesaurus*, ce qui ne permet pas l'expression de groupes de concepts provenant de plusieurs SOC. Cette modélisation permet la définition d'un ordonnancement des *ThesaurusConcepts* spécifique à un *ThesaurusArray*.

4.3.3 ISO 25964

Le passage des SOC au format électronique a rendu obsolètes les normes ISO actuelles (*cf.* section précédente). Cette future norme en cours d'élaboration, est centrée sur la gestion de thésaurus électroniques et inclut les problématiques d'interopérabilité avec d'autres SOC [Clarke 2008]. ISO 25964 propose notamment un modèle conceptuel métier (sans toutefois décrire comment l'implémenter) issu du projet BS 8723. Le seul ajout au modèle initial du projet BS 8723 concerne la représentation de groupes de concepts. Cet apport a été proposé par nos travaux de

recherche et sera présenté en section 6.3.3.

4.3.4 LMF

Lexical Markup Framework (LMF) est le standard ISO pour les lexiques du traitement automatique des langues (TAL). LMF met à disposition un modèle (fondé sur les bonnes pratiques dans ce domaine) et des méthodes pour représenter et utiliser des ressources linguistiques [LMF 2008]. L'instanciation du modèle de LMF permet la représentation de ressources lexicales multilingues et couvre aussi bien l'aspect morphologique, syntaxique que sémantique. LMF est par exemple utilisé comme format de représentation de la ressource WordNet [Soria 2009].

La figure 4.10 montre le modèle noyau de LMF. Ce modèle est centré sur l'entité *Lexical Entry* qui représente une occurence d'un groupe nominal au sein d'une ressource lexicale *Lexicon Resource* comme WordNet. Le modèle permet ensuite de décomposer ce groupe nominal pour en expliciter sa forme, son lemme, ses variantes grammaticales, etc. L'entité *Sense* peut être apparentée à la notion de concept. Ce modèle centré sur l'aspect linguistique a toutefois un aspect conceptuel assez pauvre.

4.3.5 Synthèse et discussion

La construction d'un modèle débute par l'étude de solutions existantes : des patrons de modélisation (en anglais « design pattern »), des modèles ou parties de modèle intéressants, etc. L'étude approfondie des principaux langages spécialisés dans la représentation de SOC a beaucoup apporté à l'élaboration de notre modèle (*cf.* chapitre 6) et nous pouvons dès à présent distinguer plusieurs sous-parties d'importance pour la représentation générique de SOC :

La partie conceptuelle. Cette partie est centrée autour du concept. Elle prend en compte l'organisation des concepts au sein d'un réseau sémantique composé de relations sémantiques. Parmi ces relations sémantiques, la relation hiérarchique est très importante et permet de structurer un SOC. La notion de concept est similaire entre les langages que nous avons explorés.

La partie terminologique. Cette partie apporte une composante linguistique aux concepts. La complexité de cette composante varie beaucoup selon l'utilisation d'un SOC. Elle apparait au travers de libellés posés sur un concept dans le langage SKOS (sans son extension XL) pour être décomposée en lemmes, variations syntaxiques et atomes dans LMF.

La partie de groupe de concepts. Le regroupement de concepts permet de représenter des SOC, des sous-ensembles de SOC. Le modèle de LMF est centré

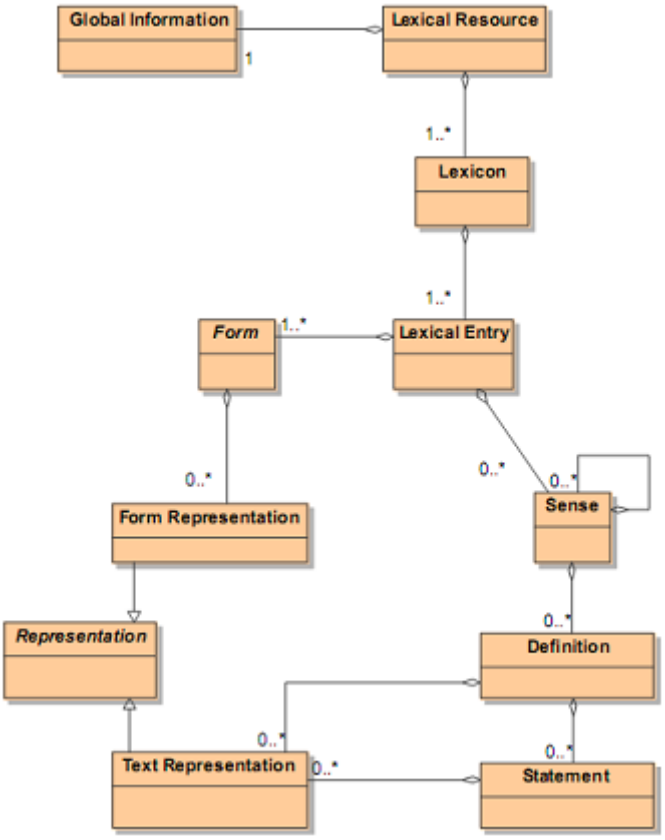

FIGURE 4.10 – Modèle du noyau de LMF. Ce modèle est centré sur l'aspect linguistique autour de l'entité *Lexical Entry*. L'élément *Sense* peut être apparenté à la notion de concept.

sur la linguistique et ne propose pas de regroupement de concepts. Cette notion présente dans les autres langages étudiés sert à la représentation d'un SOC ou d'un sous-ensemble d'un SOC. Ces définitions ne prennent toutefois pas en compte les environnements multi-terminologique où la définition de regroupement inter-SOC est important.

La partie alignement. Cette partie permet la représentation des correspondances entre concepts et données issues des alignements. la partie alignement repose dans les langages étudiés sur la proposition de SKOS.

4.4 Les langages et standards d'accès à la connaissance des SOC

Cette section propose un état de l'art sur les langages et standards d'accès à la connaissance contenue dans les SOC qui sont d'intérêt pour nos travaux. Nous

détaillons les différentes techniques d'accès et discuterons en fin de ce chapitre de leur pertinence pour notre approche.

4.4.1 SPARQL

Le SPARQL Protocol and RDF Query Language [9] (SPARQL) dans sa version 1 est une recommandation de la W3C depuis 2008 [10] [Prud'hommeaux 2008]. Ce langage permet d'exprimer des requêtes sur une base de données de type RDF mais aussi de spécifier la forme du résultat. C'est donc un langage générique de requêtage sur tous types de données pourvu qu'elles soient exprimées en RDF.

Le langage SPARQL permet non seulement l'interrogation d'un graphe RDF grâce à sa clause « SELECT » mais également la construction d'un nouveau graphe à partir d'éléments sélectionnés grâce à sa clause « CONSTRUCT ». Cet opérateur permet de considérer SPARQL comme un langage de règles de transformation pour des métamodèles exprimés en RDF. Plusieurs travaux ont démontré cette capacité [Polleres 2007, Morbidoni 2007]

La syntaxe de SPARQL est proche de celle de SQL [11] dont voici un exemple d'utilisation de la clause SELECT :

```
PREFIX ex: <http://ics.upmc.fr/schema#>
SELECT ?name
WHERE {
   ?person ex:name ?name .
   {
     { ?person rdf:type ex:Adult . }
     UNION
     { ?person ex:age ?age .
     FILTER (?age > 17) }
   }
}
```

La première ligne définit une abréviation *ex* qui fait référence à l'espace de nom `http://ics.upmc.fr/schema#`. Cette requête interroge les valeurs possibles associées à la variable *name* qui répondent au graphe dans la clause WHERE. Le graphe d'interrogation essaye de trouver des parties du graphe interrogé qui répondent au fait qu'une certaine ressource ait un prédicat `ex:name` et soit de type `ex:Adult`

9. Le nom du langage « SPARQL » est récursivement construit avec son acronyme.

10. une version 1.1 est en cours de rédaction. Cette nouvelle version complète la version précédente de nouvelles fonctions comme la possibilité de faire des requêtes imbriquées ou encore d'effectuer des requêtes par comparaison de valeurs (max, min, count, etc.)

11. Structured Query Language (SQL) est un langage de requête sur des bases de données principalement relationnelles.

ou qu'elle ait un prédicat `ex:name` et un autre prédicat `ex:age` dont la valeur est strictement supérieure à « 17 ». L'exécution de cette requête par un moteur d'interrogation de données RDF fonctionne sur un mécanisme d'appariement de graphes. La requête suivante illustre la capacité de SPARQL à servir de langage de règles de transformation de modèles exprimés en RDF :

```
PREFIX ex: <http://ics.upmc.fr/schema#>
CONSTRUCT
{
   ?person rdf:type ex:Adult
}
WHERE
{
   ?person ex:age ?age .
   FILTER (?age > 17)
}
```

Cette requête est assimilable à une règle de transformation. Une règle SPARQL est de la forme :

```
CONSTRUCT { cons } WHERE { ant }
```

où les antécédents et les conséquences sont des conjonctions de prédicats atomiques. Dans notre exemple, la règle construit un triplet indiquant que la ressource retournée par les antécédents (toute ressource ayant un prédicat `ex:age` dont la valeur est strictement supérieure à « 17 ») est de type `ex:Adult`.

4.4.2 CTS 2

Common Terminology Services dans sa deuxième version (CTS 2) est une spécification en cours de soumission auprès de l'OMG (Object Management Group). Cette spécification répond à un constat : des services d'interface aux Systèmes d'Organisation de la Connaissance devraient être suffisamment souples et génériques pour prendre en compte avec précision une grande variété de terminologies et d'autres ressources lexicales. CTS 2 est destinée à servir de médiateur entre des sources terminologiques disparates en fournissant un ensemble de services standards qui reposent sur un modèle d'information indépendant de la plateforme de mise en application. Le modèle d'information comprend la définition des entités, des attributs et des associations communs à des terminologies structurées. La spécification de CTS 2 précise la description des services et des interfaces nécessaires pour accéder et maintenir les SOC. Notons qu'elle ne mentionne aucunement comment les mettre en place. En effet, la spécification de CTS 2 est un PIM (Platform Independant Model) tel que défini par le MDA (Model Driven Architecture).

Les services proposés par CTS 2 sont regroupés dans quatre paquetages :

Administration. Permet de gérer le contenu comme partie d'un SOC : charger, exporter, activer et retirer un SOC. Ces fonctions sont généralement protégées et accessibles uniquement aux administrateurs ayant une autorisation appropriée.

Search / Access. Permet de trouver des concepts basés sur des critères de requêtes : restrictions à des associations spécifiques ou à d'autres attributs du SOC, incluant de la navigation d'associations pour les ensembles résultats.

Authoring / Curation. Permet de créer et de maintenir le contenu. Ajouter, modifier, supprimer des concepts et des associations incluant aussi la gestion des changements de SOC.

Concept Relationships. Permet de faire correspondre un concept d'un SOC source et un concept d'un SOC cible, ou de créer des associations entre concepts à l'intérieur d'un même SOC.

4.4.3 Synthèse et discussion

Un serveur multi-terminologique doit être capable d'intégrer, de stocker et de mettre à disposition des services d'accès aux connaissances des SOC. Les langages que nous venons de présenter apportent à nos travaux des solutions efficaces et de référence pour accéder aux SOC. Le langage de requêtage SPARQL est généraliste (non spécifique aux SOC) et peut servir à transformer des modèles exprimés en RDF. La spécification CTS 2 décrit quant à elle des services dédiés aux SOC.

4.5 Les principaux projets d'intégration et d'accès aux SOC

Cette section propose un état de l'art sur les principaux projets d'intégration multi-terminologique et d'accès à leur contenu. Nous détaillons ces projets d'intérêt pour nos travaux, puis en fin de chapitre, nous positionnons notre approche par rapport à ces projets.

4.5.1 UMLS

En 1986, la NLM (National Library of Medecine) a lancé un programme de développement sur plusieurs années, nommé « Unified Medical Language System » (UMLS) [Lindberg 1993]. L'objectif de ce projet est d'améliorer l'accès à l'information biomédicale à partir de sources différentes. Ce projet met donc en place une

plateforme permettant de regrouper tous les thésaurus, nomenclatures, et classifications existantes dans le domaine biomédical [Bodenreider 2004]. La force de ce projet tient dans sa définition de liens sémantiques d'équivalence entre les différents concepts des SOC intégrés à la plateforme. Cette structure permet aisément la traduction d'une terminologie dans une autre [Fung 2005] et est une ressource intéressante pour le Traitement Automatique des Langues (TAL).

UMLS est composé de trois bases de connaissances : le *métathésaurus* qui regroupe les concepts de terminologies, le *réseau sémantique* qui regroupe les types de relations entre concepts du métathésaurus et le *SPECIALIST Lexicon* qui contient les informations syntaxiques, morphologiques et orthographiques. Seules les deux premières parties sont intéressantes pour nos travaux et sont détaillées ci-après.

Le métathésaurus

Le métathésaurus est considéré comme la plus grande base de données terminologiques avec plus de 140 Systèmes d'Organisation de la Connaissance biomédicales (dont la SNOMED 3.5, la CIM-10, le MeSH et LOINC) et plus de deux millions de concepts [12]. La méthode utilisée pour l'intégration des référentiels mérite d'être étudiée plus en détail. Pour cela nous reprenons l'exemple expliqué par T. Merabti illustré en figure 4.11 [Merabti 2010]. L'UMLS regroupe sous un même concept (identifié par un *CUI* crée à cette occasion les différents termes et concepts des SOC intégrés. Nous pouvons constater que les trois entités « cold temperature » (provenant de la classification CISP2) [13], « Cold » (provenant du thésaurus MeSH) et « Cold » (provenant d'une troisième structure que nous n'aborderons pas ici), désignent un même concept dans le métathésaurus de UMLS nommé « cold temperature ». Chaque élément intégré ou crée dans UMLS se voit attribuer un identifiant unique dans la base et préfixé par la lettre correspondant au niveau termino-conceptuel auquel il appartient (*Atome*, *String*, *Terme* ou *Concept*). Les relations d'origine ainsi que celles crées par les développeurs de la NLM durant la construction du métathésaurus sont ajoutées et relient les concepts.

12. Données issues de la version 2009AA du métathésaurus.

13. Nous avons présenté en section 3.4 les SOC principaux et n'avons pas inclu la description de la CISP2. La Classification Internationale des Soins Primaires est une classification développée par l'Organisation internationale des médecins généralistes et a le même objectif que la CIM-10. Pour plus de détails, se référer aux travaux de M. Jamoulle [Jamoulle 2000].

Concepts (CUI)	Termes (LUI)	Strings (SUI)	Atomes (AUI)
C0009264 **cold temperature**	**L0215040** cold temperature	**S0288775** cold temperature	**A0318651** cold temperature (from CSP)
	L0009264 Cold Cold	**S0007170** Cold	**A0016032** Cold (from MTH)
		S0026353 Cold	**A0040712** Cold (from MeSH)

FIGURE 4.11 – Articulation entre les aspects conceptuel et terminologique de l'UMLS. Les concepts des SOC et leurs formes terminologiques sont agrégés par équivalence sémantique autour d'un identifiant unique dans UMLS nommé CUI.

Le réseau sémantique

Alors que le *métathésaurus* fournit une liste de tous les concepts et termes des SOC du domaine biomédical, le *réseau sémantique* apporte une structuration aux concepts. À chaque concept du métathésaurus sont associés un ou plusieurs types sémantiques parmi les 135 définis dans cette partie de l'UMLS (e.g. « disease or syndrome »). En plus de ces types, le *réseau sémantique* définit 54 relations sémantiques (hiérarchiques ou associatives) qui permettent de relier les concepts. Même si la partie *réseau sémantique* de l'UMLS n'est pas intentionnellement définie comme une ontologie, les efforts de spécification de types et de relations sémantiques nous permettent de la considérer comme telle [Charlet 1996, Bachimont 2000].

4.5.2 GALEN

Le projet européen General Architecture for Language and Nomenclatures (GALEN) [14] vise à mettre en place un serveur de terminologies en médecine. Ce projet repose sur la définition d'une top-ontologie de la médecine, le *GALEN CORE Model* respectant les principes énoncés en section 3.3.6. Cette ontologie respecte une structure arborescente au niveau de ses types primitifs et est le cœur du système et des services qu'il propose [Rector 1998]. Le formalisme utilisé pour la modélisation de l'ontologie s'appelle GRAIL (pour GALEN Representation And Integration Lan-

14. Voir http://www.opengalen.org/

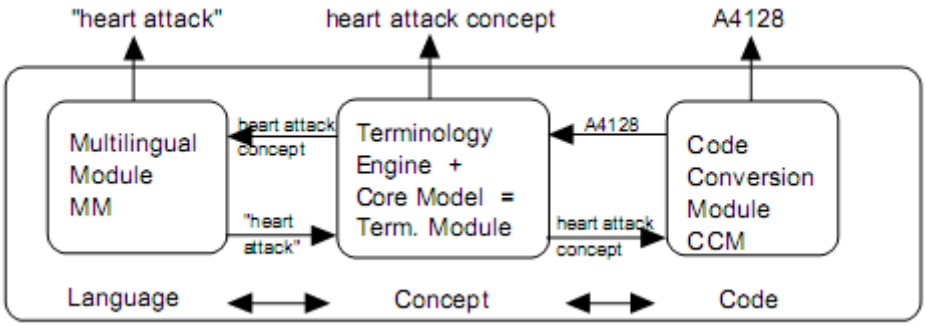

FIGURE 4.12 – Architecture du serveur de terminologies GALEN TeS. Cette figure est extraite de l'article [Goble 1994]. Cette architecture sépare les aspects conceptuel, terminologique et de codage.

guage) [Rector 1993]. GRAIL est un langage génératif qui permet la définition de nouveaux concepts complexes (ou définis) par combinaison de concepts élémentaires (ou primitifs).

L'approche par ontologie du projet GALEN fournit un modèle extensible. Couplé à son langage de représentation formel, il rend possible l'exécution de requêtes, d'inférences et de services pour le traitement automatique des langues. Comparé aux efforts de formalisation du projet UMLS qui définit 135 types sémantiques, le projet GALEN place la formalisation au centre de son approche et fournit une hiérarchie de plusieurs milliers de concepts pour le domaine médical sur lesquels il est possible d'intégrer les SOC [Goble 1994]. Le projet PEN&PAD détaillé dans [Rector 1992] illustre l'utilisation du serveur de terminologies TeS développé par le projet GALEN. Ce serveur propose une séparation entre les aspects conceptuel, terminologique et de codage comme l'illustre la figure 4.12. Cette architecture a permis dès 1995 de gérer le mutlilinguisme et de fournir un serveur multi-terminologique. Toutefois, l'expression des connaissances se fait dans un langage non standard [15] et ne satisfait pas nos besoins d'interopérabilité.

4.5.3 LexGrid

Lexical Grid (LexGrid [16]) est un projet communautaire débuté en 2005 et coordonné par la Mayo Clinic [17] qui a pour but la représentation et le stockage de SOC. Ce projet repose sur un modèle intentionnellement flexible pour permettre la représentation fine de structures de la connaissance exprimées dans divers langages et

15. Les standards tels que OWL ont été développés après le projet GALEN
16. Voir https://cabig-kc.nci.nih.gov/Vocab/KC/index.php/LexGrid
17. Voir http://www.mayoclinic.com/

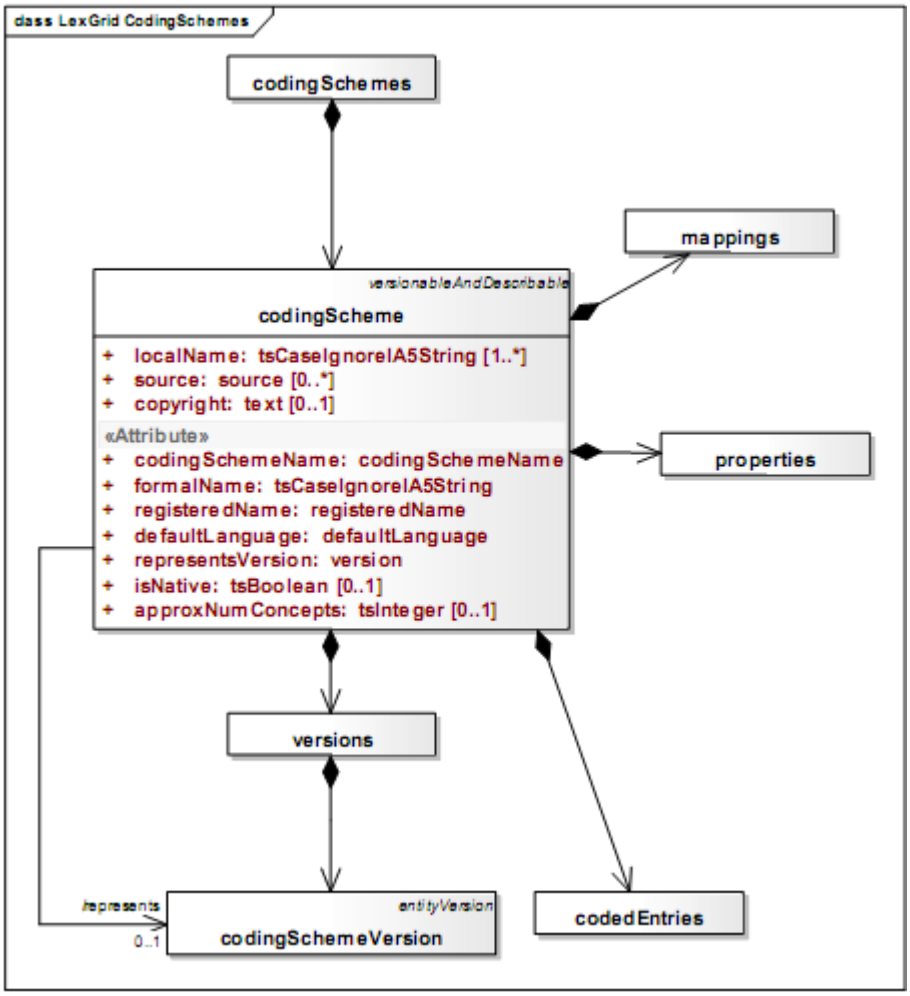

FIGURE 4.13 – Éléments principaux du modèle de LexGrid.

formats. Il est compatible avec les représentations OBO (*cf.* section 4.2.2.4), OWL (*cf.* section 4.2.2.3), le format d'échange UMLS ou encore le modèle RIM de HL7 [Tao 2009]. Autour de ce modèle s'articule un ensemble de services mis à disposition par le projet LexBIG [18] au travers d'API [19].

Détaillons les éléments clés du modèle de LexGrid présentés en figure 4.13. un *codingScheme* représente un SOC composé d'un ensemble de *codedEntries*. Cet élément *codedEntries* représente la notion de concept.

Le modèle de LexGrid a été pensé pour conserver au maximum l'état des SOC

18. LexBIG est un projet qui implémente le modèle de LexGrid pour offrir un ensemble de services d'accès au contenu du serveur de terminologies en respectant les spécifications du Cancer Biomedical Informatics Grid (caBIG®) [Cimino 2009].

19. Application Programming Interface. Interface d'accès aux méthodes d'une application.

Resource	Value	CodingScheme fullName	Property propertyName	Property propertyValue	Property lexGridName
MRSAB SON	International Health Terminology Standards Development Organisation, SNOMED Clinical Terms.	International Health Terminology Standards Development Organisation, SNOMED Clinical Terms.	SON	International Health Terminology Standards Development Organisation, SNOMED Clinical Terms.	fullName
dc:title	SKOS Core Vocabulary 2006-04-18 '2nd W3C Public Working Draft (Amended) Edition	SKOS Core Vocabulary 2006-04-18 '2nd W3C Public Working Draft (Amended) Edition	Title	SKOS Core Vocabulary 2006-04-18 '2nd W3C Public Working Draft (Amended) Edition	fullName
dct:issued	2006-04-18		Issued	2006-04-18	

FIGURE 4.14 – Exemple de représentation de propriétés dans LexGrid. Le modèle LexGrid a pour caractéristique de ne pas modifier les connaissances et leurs structures d'origine. Cette illustration est extraite de [Pathak 2009].

d'origine. Ce choix a pour conséquence de redéfinir pour chaque système, les propriétés et relations qui le composent [Pathak 2009]. En ceci le modèle de LexGrid se situe au même niveau d'abstraction que les langages généralistes (*cf.* section 4.2). La figure 4.14 illustre l'intégration de trois propriétés de types différents au modèle de LexGrid. Dans cet exemple, toutes les données de la propriété source sont stockées : son type, sa valeur, etc. Il n'y a pas de transformation de types d'origine vers des types du modèle LexGrid. L'avantage de cette méthode est de ne pas dénaturer l'information d'origine ; l'inconvénient est de ne pas avoir une représentation unifiée, ce qui rend plus difficile l'interrogation du contenu d'un serveur utilisant ce modèle.

4.5.4 BioPortal

BioPortal est une application web [20] développée par le National Center for Biomedical Ontology (NCBO) qui permet d'accéder et de partager des ontologies biomédicales aux formats OBO (*cf.* section 4.2.2.4), OWL (*cf.* section 4.2.2.3) et Protégé frames [21] [Noy 2009].

L'architecture de BioPortal repose sur le modèle de LexGrid pour représenter les SOC exprimés en OBO et sur Protégé pour représenter les référentiels exprimés en

20. Voir `http://bioportal.bioontology.org/`
21. Pour plus d'informations voir [Wang 2006].

FIGURE 4.15 – Recherche des concepts ayant pour terme « conjunctivitis » sur le portail Web BioPortal. 634 concepts ont été trouvé figurant dans 35 SOC différents.

OWL. Ce projet permet de publier, de commenter un SOC mais aussi d'ajouter des alignements inter-terminologiques. Une autre utilisation possible est la recherche de connaissances comme l'illustre la figure 4.15. le projet stocke plus de 180 SOC pour un nombre total de plus de 650 000 concepts. Les mots « term » et « ontology » sont utilisés dans le projet BioPortal pour désigner les notions que nous appelons « concept » et « SOC ».

Les services de recherche, d'intégration et de visualisation sont particulièrement intéressants dans ce projet. La rapidité d'accès à un tel volume de connaissances en est un autre point fort. Toutefois la recherche d'information se limite à une recherche textuelle là où la structure et les relations sémantiques des SOC permettent de formuler des recherches plus complexes.

4.5.5 Synthèse et discussion

La proposition de solutions pour la mise à disposition d'un serveur multi-terminologique n'est pas nouvelle. Ces solutions peuvent se distinguer par leur approche d'intégration des SOC. Sur la base des deux types d'approche d'intégration identifiés en section 3.5.2 : l'approche distribuée et l'approche centralisée, nous pouvons discuter des projets étudiés ci-dessus.

Tous les projets que nous avons présentés ci-dessus utilisent l'approche centra-

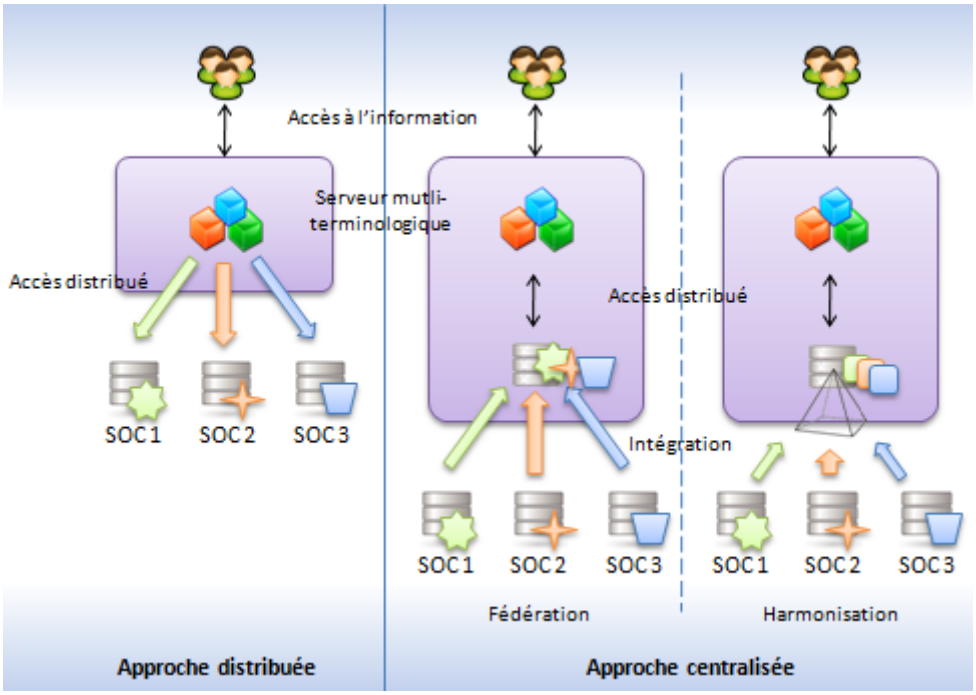

FIGURE 4.16 – Approches et méthodes d'intégration de SOC dans un serveur multi-terminologique.

lisée. Ceci peut s'expliquer par la nature des connaissances intégrées : les SOC. Les SOC sont des ressources volumineuses qui proposent des évolutions de manière régulière dont la fréquence est généralement supérieure à la semaine. Ces caractéristiques (de volumétrie et de stabilité) rendent l'approche centralisée plus efficace que l'approche distribuée pour l'intégration de SOC.

Toutefois, nous pouvons discerner deux méthodes qu'utilisent ces projets au sein de l'approche centralisée. Nous présentons en figure 4.16 les approches distribuée et centralisée, la dernière comprenant deux méthodes par fédération et par harmonisation.

La méthode par fédération. Cette méthode consiste à intégrer des SOC dans un cadre en préservant la structure d'origine. L'avantage de cette méthode est de rester fidèle au modèle d'origine du SOC. Les projets LexGrid et BioPortal adoptent cette méthode.

La méthode par harmonisation. Cette méthode consiste à intégrer des SOC dans un cadre en transformant la structure d'origine vers un format unificateur. L'avantage de cette méthode est d'offrir un accès identique au contenu des SOC quel que soit le SOC. Les projets UMLS et GALEN utilisent cette méthode.

Deuxième partie

De l'élaboration du modèle

Chapitre 5

Problématique scientifique et enjeux

Sommaire

Dans ce chapitre, nous situons la problématique de notre travail de thèse et ses enjeux en expliquant quels étaient nos objectifs pour MONDECA et l'INSERM. L'énoncé de ces objectifs soulève des problèmes que nous identifions. Nous distinguons ensuite plusieurs hypothèses de recherche sur lesquelles nous avons fondé notre travail pour répondre à ces problèmes. Enfin nous faisons la synthèse et soulignons l'originalité de nos travaux et nos contributions aux domaines de l'Ingénierie des Connaissances, de l'Ingénierie des Modèles et de l'Informatique Médicale.

5.1 Objectifs

L'utilisation de SOC comme contexte commun de connaissances est maintenant établie. Les Systèmes d'Informations qui utilisent des SOC sont confrontés à un choix [Nowlan 1994] : soit chaque application au sein de ces systèmes gère son propre accès

aux référentiels dont il a besoin, soit des services communs sont mis à disposition grâce à un serveur dédié. Nos travaux ont pour but de montrer la faisabilité de ce dernier choix et poursuivent les objectifs suivants :

1. **Modéliser les SOC, leurs correspondances et les connaissances métiers.** Il faut être capable de modéliser les SOC dans toutes leurs diversités (de format de représentation, structurelle, formelle, de richesse linguistique, etc.). Ces SOC sont souvent complémentaires et présentent des recouvrements. Le modèle doit permettre la représentation de correspondances entre concepts en recouvrement. Dans certains cas (cf. Section 8.2), les SOC ont une définition locale liée à des connaissances métiers annexes. Nous devons pouvoir étendre le modèle à la représentation des connaissances métiers liés aux SOC. Un modèle de SOC doit supporter le multilinguisme et l'expressivité linguistique et sémantique.
2. **Composer ces connaissances au sein d'un serveur.** Notre solution doit permettre la mise à disposition, au sein d'un serveur, des SOC utiles à un Système d'Information. Ce serveur doit intégrer les référentiels sans pour autant fusionner ceux-ci. Le respect de l'identité de chaque SOC permet de mettre à jour individuellement ces référentiels et leurs correspondances. Un serveur de SOC doit permettre l'intégration de grands volumes de données.
3. **Proposer des interfaces et services optimisés pour le processus éditorial.** Le serveur de SOC doit proposer des services communs, quel que soit le référentiel, qui couvre l'ensemble du processus éditorial [Chute 1999]. En tant que médiateur entre systèmes, le serveur doit permettre l'import, la mise à jour et l'export de tout ou partie de son contenu dans les formats souhaités.

Dans la section suivante, nous allons préciser les problèmes que soulève la réalisation de ces trois objectifs.

5.2 Problèmes

5.2.1 Limites de l'interopérabilité des SOC

Comme nous avons vu en section 3.5.2, l'un des principaux objectifs des SOC est de fournir à une communauté d'utilisateurs une conceptualisation partagée d'une partie spécifique du monde, dans le but de faciliter la communication efficiente d'une connaissance complexe. Cet objectif favorise l'utilisation de SOC dont la syntaxe et la sémantique sont normalisées pour favoriser l'interopérabilité de Systèmes d'Information. Toutefois, la diversité et l'hétérogénéité des SOC (*cf.* chapitre 3) déportent

ce problème d'interopérabilité sur les outils, les systèmes et les solutions pour la gestion conjointe de SOC. Les questions suivantes peuvent légitimement être posées quant à la réalisation des objectifs 1 et 2 :

- *Comment placer ces SOC à un même niveau d'interopérabilité pour pouvoir lier leurs connaissances ?*
- *Comment prendre en compte l'hétérogénéité des SOC sans altérer l'information spécifique à chacun d'eux ?*
- *Quel langage de représentation utiliser pour rester compatible avec les standards existants ?*
- *Comment gérer ces différents référentiels en sachant les recouvrements potentiels ?*

Nous approfondissons ces problèmes dans les sections suivantes avant de détailler notre proposition.

5.2.1.1 Hétérogénéité de représentation des SOC

L'hétérogénéité des formats et langages de représentation des SOC et des correspondances est un frein à leur utilisation et donc à l'interopérabilité entre les systèmes qui les utilisent. La plupart des référentiels sont disponibles dans des formats spécifiques, parfois même propriétaires. C'est le cas de LOINC qui est mis à disposition soit au format texte soit dans un format d'échange et de représentation propriétaire de type tableur [1]. L'organisation au sein de ce fichier (par des séparateurs ou dans des colonnes) est entièrement spécifique à LOINC. Prenons maintenant un autre exemple : la SNOMED 3.5 [2]. Cette terminologie est disponible sous forme de fichiers dans un format tableur propriétaire avec de nouveau une structure spécifique en colonnes.

En plus de l'hétérogénéité, l'utilisation de **formats non adaptés** à la représentation de connaissances induit des erreurs. Par exemple, la structure hiérarchique de la SNOMED 3.5 est présente de manière implicite dans le fichier tableur mis à disposition. Comme le montre la figure 5.1, ceci génère une ambiguïté d'interprétation où le même code « TERMCODE » est donné à plusieurs concepts différents de la hiérarchie. La documentation ne précise pas comment interpréter cette situation.

L'hétérogénéité des SOC tient également à la diversité des langages pour les décrire (*cf.* chapitre 4). Avec les langages de description de SOC, varient la

1. Accessible à l'URL : http://loinc.org/downloads/loinc
2. Accessible à l'URL : http://esante.gouv.fr/snomed/snomed/

	A	B	C	D	E
1	LIGN	TERMCODE	FMOD	FCLASS	FNOMEN
2	2	D0-00000			Chapitre 0 Maladies de la peau et des tissus sous-cutanés
3	3	D0-00000		-	Section 0-0 Maladies de la peau et des tissus sous-cutanés: termes généraux, types histologiques et infections
4	4	D0-00000		0	0-00 Maladies de la peau et des tissus sous-cutanés: termes généraux et types histologiques
5	5	D0-00000		00	0-000 Maladies de la peau et des tissus sous-cutanés: termes généraux
6	6	D0-00000		01	maladie de la peau et du tissu sous-cutané
7	7	D0-00004		01	maladie de la peau
8	8	D0-00004		02	dermatose
9	9	D0-00004		02	affection de la peau

FIGURE 5.1 – Ambiguïté autour de l'interprétation de l'extrait d'un fichier de la SNOMED 3.5 au format tableur. Cet exemple illustre la non adéquation du format tableur pour représenter des SOC. La hiérarchie de la SNOMED 3.5 est ici implicite et doit se déduire de l'ordre d'apparition des « TERMCODE ».

richesse linguistique et l'expressivité formelle. Un SOC exprimé dans un langage non formel ne permet pas la mise en place de contrôles automatiques de cohérence des données par rapport à leur modèle. Certains SOC ne reposent sur aucun langage ou recommandation standard. Parmi les exemples détaillés en section 3.4, seuls LOINC et Eurovoc sont explicitement conformes à des modèles standards de représentation de la connaissances (HL7 pour LOINC ; ISO 2788, ISO 5964-1985 et SKOS pour Eurovoc).

La diversité des formats et des langages des SOC s'explique, par le fait que ces référentiels ont été développés de manière indépendante en privilégiant l'infrastructure informatique déjà en place, ainsi que par le choix de la solution la plus facile à mettre en œuvre. En effet, durant les premières étapes de création d'un SOC, la mise en place d'une plateforme dédiée à l'édition de SOC représente un investissement important (coût, installation et formation des utilisateurs). Il s'ajoute à ceci que certains SOC ont un historique important (pré-électronique) et peuvent ne pas utiliser les standards et recommandations.

La diversité de structure. Dépendant de la nature d'un SOC et de son utilisation, sa structuration peut varier. Pour ne prendre que l'exemple de la relation hiérarchique, elle peut ne pas exister (*e.g.* le dictionnaire LOINC) ; elle peut être présente et utilisée pour modéliser une relation de subsomption aussi bien qu'une relation de partition (*e.g.* la terminologie ATC) ; enfin, elle peut ne représenter que les relations de subsomption et donc être transitive (*e.g.* la

FMA).

5.2.1.2 La représentation des mises en correspondance

L'identification et la représentation des recouvrements entre SOC sont une réponse à leur hétérogénéité. Les alignements sont alors des ponts sémantiques entre ces différents référentiels (*cf.* section 3.2). Pour prendre en compte la représentation de correspondances et de qualificatifs issus de l'alignement, le modèle de SOC doit proposer des éléments pour la représentation de correspondances entre concepts. Ces types d'artefacts sont utilisés pour modéliser le résultat d'alignements entre des SOC partageant un domaine commun mais sous des points de vues différents ou pour modéliser le résultat d'alignements d'un SOC utilisé localement avec un SOC de référence [Daniel 2009].

5.2.1.3 Propositions

Nous proposons de mettre en place une solution **distribuée par harmonisation** (*cf.* sections 3.5.2 et 4.5.5) grâce à un **modèle commun et extensible** de représentation de SOC. Cette solution nous semble la plus adaptée à la représentation, au stockage et à la mise à disposition de SOC et de ses correspondances (*cf.* section 4.5.5). En effet, elle permet d'accéder de manière identique aux connaissances de chaque SOC. Cette solution comprend (i) la construction d'un modèle noyau commun ; (ii) la modélisation d'extension par SOC pour la représentation de particularités et (iii) la migration des données vers ce nouveau modèle. Notre solution ne prétend pas représenter tous les SOC dans leur complexité mais proposer un modèle noyau qui factorise les éléments couramment décrits dans des SOC. Ce modèle doit être extensible pour prendre en compte les spécificités de chaque référentiel.

Nous proposons de modéliser **chaque SOC indépendamment** avec pour seuls ponts les correspondances (sans fusionner les concepts en recouvrement). Une correspondance entre concepts est un élément bien distinct (des SOC) qui peut être généré par un outil d'alignement [Mazuel 2009]. Les informations du type d'algorithme d'alignement utilisé ou de son score sont importantes et ne doivent pas être perdues [Euzenat 2007]. La représentation de ces métadonnées sur les correspondances doit être possible dans notre modèle.

5.2.2 Limites des outils et services de gestion de SOC

Un modèle ne permet pas en soi de proposer d'enregistrer, de manipuler des informations, ou d'offrir des services. C'est le couplage du modèle à un outil qui permet de rendre ces actions effectives. Cette intégration soulève les questions suivantes qui concernent la réalisation des objectifs 2 et 3 :

- *Quel outil choisir pour l'intégration et l'interprétation de notre modèle formel ?*
- *Quels services d'édition et d'accès à l'information peut-on proposer en intégrant notre modèle à un outil dédié ?*
- *Comment importer ou exporter tout ou partie des SOC vers et depuis notre modèle ?*

Nous approfondissons ces problèmes dans les sections suivantes avant de détailler notre proposition.

5.2.2.1 Outils de représentation de SOC

Pour satisfaire notre besoin de couplage avec un outil, il faut que celui-ci puisse interpréter la définition formelle de notre modèle et y associer des comportements. Par exemple, l'édition d'un SOC au travers de cet outil doit respecter les contraintes formelles du modèle telles que la cardinalité des valeurs d'un attribut ou encore la restriction des domaine et co-domaine d'une relation. L'outil de représentation de SOC doit fournir des interfaces Homme-Machine et supporter les services décrits ci-dessous.

5.2.2.2 Services autour des SOC

Nous pouvons distinguer trois catégories de services autour des SOC :

- **les services d'édition.** Ces services ont pour but de créer, modifier, supprimer tout ou partie d'un SOC ou de ses alignements.
- **les services d'accès.** le but de ces services est d'offrir un moyen pour l'interrogation efficace des SOC afin de satisfaire un cas d'utilisation. Par exemple, un service qui permet de retrouver l'ensemble des éléments des SOC ayant un libellé contenant « infarctus » est utile dans les cas d'utilisation mettant en place un champ de recherche sur le contenu de SOC de santé.
- **les services d'import et d'export depuis et vers un format particulier.** Les services d'import et d'export sont utiles pour ajouter, mettre à jour, publier ou échanger des SOC.

5.2.2.3 Propositions

Dans le cadre de notre thèse en collaboration avec l'entreprise MONDECA, nous utilisons l'outil ITM qui permet l'interprétation d'un modèle formel. Toutefois, nous étudierons en section 7.3 la faisabilité d'intégration de notre modèle au sein d'outils ouverts.

Le logiciel ITM possède un ensemble de services pour l'accès à des connaissances. Les services d'édition sont gérés nativement au sein de l'outil ITM qui bénéficie d'une expérience de plus de 10 années dans l'édition de SOC. Nous proposons de développer de nouveaux services d'accès dédiés aux SOC. Dans ce domaine, le standard CTS 2 (*cf.* section 4.4.2) est à notre sens le plus approprié pour satisfaire les besoins de nos clients et partenaires sur les projets que nous avons menés.

Nous proposons également des services d'import et d'export depuis et vers notre modèle. La solution que nous voulons mettre en place pour remplir ces services repose sur la méthode de transformation de modèles (*cf.* section 2.3.2.2) depuis notre modèle formel vers les modèles cibles. Toutefois, cette solution ne permet pas la prise en charge de certains formats dépourvus de modélisation projetables sur la nôtre. Dans ces cas là, nous devons développer un programme spécifique pour chaque format nous amenant à un modèle que l'on peut ensuite transformer.

5.3 Hypothèses de travail

Plusieurs chercheurs ont posé l'hypothèse de la pertinence d'un métamodèle pivot pour la représentation et la médiation sémantique de SOC [Lindberg 1993, Rector 1998, Miles 2006]. La lecture de leurs recherches dans la littérature nous a amenés à dresser la liste suivante d'hypothèses qui concernent plus spécifiquement notre problématique :

Considérer les SOC étudiés comme représentatifs. Prétendre que notre modèle est capable de représenter n'importe quel SOC revient à effectuer le test d'intégration pour chacun d'eux. Cette exhaustivité n'est pas réalisable dans le cadre de notre travail de thèse. Nous considérons que les SOC pour lesquels nous avons testé notre solution sont suffisamment variés (type, format, langage, formalisme, expressivité linguistique) pour être représentatifs de la généricité de notre approche.

Utiliser un métamodèle comme langage pivot pour harmoniser les SOC. Nous avons exploré en section 4.5.5 différentes approches pour intégrer des SOC. La solution centralisée par harmonisation nous semble la plus appropriée (*cf.* section 4.5.5) pour l'intégration de SOC. Nous avons fait le choix

d'utiliser un métamodèle pivot pour harmoniser ces SOC et situons notre approche comme semblable aux langages de représentation spécialisés dans la représentation de SOC (*cf.* section 4.3).

Appliquer les méthodes MDA pour la représentation de SOC. la méthodologie que nous proposons est centrée sur les modèles et utilise les trois notions étudiées dans l'approche MDA (*cf.* section 2.3.2). L'élaboration de notre modèle commun de représentation de SOC présentée au chapitre 6, repose sur l'indépendance de description des modèles (*cf.* section 2.3.2.3). Notre modèle s'intègre dans l'approche par méta-modélisation. Nous utilisons la méthode de transformation de modèles pour l'import et l'export de tout ou partie de SOC.

5.4 Synthèse et originalité des travaux

Les travaux présentés dans la dernière partie de ce mémoire apportent une solution aux objectifs qui nous étaient fixés. Cette solution met en place une méthode originale vis-à-vis des approches existantes pour pallier les problèmes auxquels nous faisons face. Nos travaux s'appuient sur des hypothèses pour répondre à la problématique suivante :

Comment harmoniser la représentation des SOC et de leurs correspondances afin de proposer des services unifiés qui supportent l'édition, la publication et l'utilisation efficaces des connaissances de ces référentiels ?

Nous pouvons présenter l'originalité de nos travaux sous trois axes de contributions :

Les contributions méthodologiques. L'originalité de notre recherche réside en l'utilisation conjointe de méthodologies issues de l'Ingénierie des Connaissances, de l'Ingénierie des Modèles et de la sémantique. Nous avons commencé par élaborer un modèle qui factorise les éléments couramment utilisés dans la représentation de SOC. Cette phase s'est inspirée des réflexions et mises en pratique du projet InterSTIS (*cf.* section 6.1). Ce modèle, extensible pour prendre en compte les spécificités de chaque SOC, est dans un premier temps exprimé dans un langage indépendant de l'informatisation : UML. Il est ensuite transformé dans le langage formel OWL qui permet l'expression de la sémantique associée à notre modèle. Nous poursuivons ensuite l'approche MDA par la transformation de ce modèle exprimé en OWL en des modèles ayant pour métamétamodèle RDF. Cette technique permet l'export de tout ou partie du contenu des SOC et emploie le langage SPARQL pour l'expression des règles de transformation (*cf.* section 7.1).

Les contributions techniques. Notre travail de recherche a également un caractère original de par son intégration au sein de l'outil développé par l'entreprise MONDECA. L'utilisation de notre métamodèle intégré à l'outil ITM (*cf.* section 7.2) a permis le développement dans cet outil, de nouvelles techniques de transformation de modèles qui permettent des conversions de connaissances dans divers formats standards. À cette occasion, nous avons pu cerner quels étaient leurs atouts et leurs limites. Cela nous a fait réfléchir aux possibilités qu'offraient ces techniques de transformation et aux meilleurs moments pour les utiliser. Les contributions de nos travaux ne sont toutefois pas limitées à l'outil ITM. Nous avons démontré la faisabilité de son intégration à des outils ouverts du Web sémantique (*cf.* section 7.3). Cette solution a conduit au développement d'un outil de visualisation qui repose sur l'interrogation de SOC intégrés à notre modèle (*cf.* section 7.3.2).

Les contributions pratiques. Notre travail de recherche a abouti à de nombreuses mises en application tant dans le domaine de la recherche que dans des projets commerciaux. Nous détaillerons dans le chapitre 8 la mise en œuvre de notre solution dans trois projets du domaine de la santé (InterSTIS, AnaBio et LERUDI) et un projet du domaine de la documentation (Eurovoc). Ces projets confortent la solution originale que nous proposons dans ce mémoire. Ces mises en applications mettent en avant la valeur ajoutée de la représentation formelle et normalisée de SOC.

Avant de poursuivre la description de nos travaux, nous proposons au lecteur une petite digression présentée en annexe A, page 181. Cette digression porte sur une métaphore de nos travaux avec l'apiculture.

Chapitre 6

Construction du modèle UniMoKR

Sommaire

Comme nous venons de le voir dans les chapitres précédents, il existe une diversité dans la représentation et l'organisation des connaissances qui s'explique par des historiques, des objectifs, des utilisations différents. Néanmoins ces structurations ont toutes pour vocation d'appréhender de l'information, de la partager et de permettre un traitement humain et computationnel. Ce chapitre se propose d'extraire de cette hétérogénéité apparente, un noyau commun, une modélisation commune de ces structurations de connaissances. Nous présentons le projet InterSTIS au sein duquel nous avons élaboré une grande partie de notre modèle. Nous expliquons ensuite l'utilisation de l'approche MDA dans notre méthodologie avant d'aborder l'élaboration de notre modèle UniMoKR. nous terminons ce chapitre par justifier le caractère extensible de ce modèle et le langage choisi pour l'exprimer.

6.1 InterSTIS, le projet de construction d'un serveur multi-terminologique en santé

InterSTIS (Interopérabilité Sémantique des Terminologies dans les systèmes d'Information de Santé français) est un projet de recherche financé par l'Agence Nationale de la Recherche (ANR-07-TecSan-10) [1]. Ce projet est coordonné par l'entreprise Vidal [2] et la Faculté de Médecine Université de la Méditerranée à Marseille : le LERTIM [3]. Les autres acteurs de ce projet sont MONDECA, Memodata [4], le CISMeF [5], le DSPIM [6], le labSTIC [7], le LIMSI [8] et HON [9].

6.1.1 Contexte et enjeux

Le projet InterSTIS a pour objectif de centraliser des terminologies francophones en santé et de les rendre interopérables au sein d'un serveur terminologique multi-sources. La réalisation d'un tel serveur passe par : la constitution d'une base terminologique unifiée, l'établissement de correspondances entre ces ressources (médiation sémantique) et la normalisation par l'adoption d'un format pivot ou métamodèle. Ces services offerts interactivement aux utilisateurs et aux applications clientes seront à même d'aider les professionnels de la santé à indexer des documents de natures différentes avec la terminologie appropriée au type de document, et d'aider tous les utilisateurs (professionnels, patients, étudiants, simples citoyens, etc.) à rechercher efficacement des informations et des données. Le projet vise exclusivement des terminologies francophones. Et si certaines d'entre elles sont déjà intégrées sous leur forme anglo-saxone à des systèmes comme l'UMLS (c'est le cas de la SNOMED 3.5, de la CIM-10 et de la CISP ; *cf.* section 4.5.1), ou à des serveurs de terminologies étrangers comme le Galen Terminology Server(c'est le cas de la CCAM ; *cf.* section 4.5.2), aucun serveur de terminologies francophones ne les rendait interopérables jusqu'à présent.

Notre travail de recherche a pris place au sein du projet InterSTIS et nous a permis d'élaborer une grande partie de notre méta-modèle. Toutefois, ce projet ne couvre pas les besoins d'édition et de maintenance des SOC.

1. Voir http://www.agence-nationale-recherche.fr/
2. Voir http://www.vidal.fr/
3. Voir http://cybertim.timone.univ-mrs.fr
4. Voir http://www.memodata.com
5. Voir http://www.chu-rouen.fr/cismef
6. Voir http://dossier.univ-st-etienne.fr/dspim/www/
7. Voir http://portail.unice.fr/jahia/page4693.html
8. Voir http://www.limsi.fr
9. Voir http://www.hon.ch/

6.1.2 Intérêts scientifiques

L'intérêt principal de ce projet repose sur le fait qu'il se propose d'intégrer de manière homogène, les SOC usuels francophones dans le domaine médical. Ce projet propose une approche pragmatique d'intégration de SOC. Il débute par l'étude de référentiels standards existants qui représentent chacun un pan différent des connaissances médicales (SNOMED 3.5 : la clinique, CIM-10 : la mortalité et la morbidité, CCAM : les actes pratiqués, CISP : le point de vue du médecin généraliste). Il continue par la conception d'un métamodèle capable de rendre interopérable les SOC et par la modélisation (extension du métamodèle) de chaqu'un d'eux au moyen d'une langage formel. Cette conception de l'interopérabilité n'est pas nouvelle, mais appliquée aux SOC et en particulier dans le domaine médical, elle ouvre une perspective nouvelle que l'on veut se voir généraliser.

6.1.2.1 Médiation sémantique par un méta-modèle pivot

La méthodologie adoptée dans le projet InterSTIS est de : (i) concevoir un métamodèle dans lequel chaque modèle de SOC puisse s'intégrer [Vandenbussche 2009], (ii) concevoir et développer un processus d'intégration des référentiels dans un serveur multi-terminologique conforme au méta-modèle, et (iii) construire et intégrer les résultats d'alignements entre terminologies au sein dudit serveur. Cette approche présente l'avantage de combiner le respect des structures originelles de chacune des terminologies avec un regroupement des métadonnées inhérentes à chacune d'elles.

Nous avons conçu un modèle de chacun des SOC. La confrontation de ces modèles a permis de dégager un méta-modèle dont chacun des modèles initiaux est une spécialisation. L'objectif de ce métamodèle est de factoriser les artefacts (classes, relations, attributs) communs à l'ensemble de ces référentiels. Cette factorisation facilite l'intégration de multiples terminologies au sein d'une même plateforme. L'identification des éléments communs comporte une part de subjectivité : elle se base sur des standards et sur notre expérience concernant la gestion de terminologies. Des artefacts, spécifiques à certains SOC, doivent néanmoins être représentés pour ne pas perdre d'information en dehors du périmètre du métamodèle unifié. Nous avions donc un équilibre à choisir : représenter fidèlement les SOC sans perte d'information tout en extrayant les artefacts qu'ils ont en commun dans le but d'offrir par la suite des services unifiés indépendants d'un SOC en particulier.

6.1.2.2 De l'étude des mises en correspondance

La médiation sémantique que nous voulons mettre en place nécessite l'élaboration d'un méta-modèle pivot de représentation de SOC mais également de mises en correspondance entre les concepts de ces ressources. Nous devons donc prendre en compte ces nouveaux artefacts dans notre modélisation.

La tâche d'alignement a été réalisée par le LERTIM et le CISMeF et utilise l'UMLS (*cf.* section 4.5.1) [Joubert 2011]. L'un des composants de l'UMLS de la National Library of Medicine des Etats-Unis est son Metathesaurus. De manière plus particulière, ont été utilisées : (i) une table nommée MRCONSO qui recense de manière unique chaque concept répertorié dans UMLS et auquel un identificateur est attribué (CUI), (ii) une table nommée MRREL qui décrit les relations, lorsqu'elles existent, entre les concepts dans leur terminologie d'origine et (iii) une table nommée MRMAP qui décrit des alignements explicites entre les terminologies intégrées dans UMLS.

Ce travail a été réalisé avec la version 2009 AA de UMLS. La méthode d'alignement qui a été adoptée est en accord avec la définition des propriétés de correspondances de SKOS (*cf.* section 4.3.1). Elle est la suivante : supposons deux concepts c1 et c2 de deux SOC S1 et S2 respectivement ; supposons CUI1 et CUI2 les projections respectives de c1 et c2 dans le Metathesaurus, alors c1 et c2 sont en correspondance si :

- CUI1=CUI2 (dans MRCONSO), ceci correspond à la propriété SKOS *ExactMatch* ;
- un parent dans la hiérarchie de c1 est aligné avec c2 ou inversement (grâce à MRREL), ceci correspond aux propriétés SKOS *BroadMatch* et *NarrowMatch* ;
- il existe une correspondance explicite entre CUI1 and CUI2 (dans MRMAP), ceci correspond à la propriété SKOS *CloseMatch*.

L'algorithme est déroulé séquentiellement et s'arrête dès qu'une tentative d'alignement est fructueuse.

Pour bien comprendre la dernière étape de la démarche, prenons un exemple. Quand une correspondance explicite existe entre deux concepts c1 et c2 identifiés respectivement par CUI1 et CUI2 provenant de deux SOC (*e.g.* ICD-9-CM vers SNOMED CT [Imel 2002]), on peut en déduire que tout autre couple de concepts c3 et c4 identifié respectivement par CUI1 et CUI2 (peu importe le langage dans lequel ils sont désignés) a cette même correspondance. En d'autres termes, des correspondances explicites entre référentiels peuvent être « réutilisées » pour d'autres SOC du fait de l'organisation conceptuelle de UMLS [Fung 2005] et reposent sur l'équivalence sémantique entre les concepts ayant le même CUI.

Les alignements entre terminologies sont opérés deux à deux. Les résultats sont stockés dans une base de données relationnelle afin d'en extraire trois fichiers au format RDF pour chaque paire de terminologies traitées : un fichier d'*ExactMatch*, un fichier décrivant les *BroadNarrowMatch*, et un dernier fichier décrivant les *CloseMatch*. Nous intégrons ensuite ces jeux de correspondances dans notre solution. Les algorithmes d'alignement ne sont pas dans le périmètre de ce mémoire et ne sont donc pas détaillés. Pour des précisions sur les algorithmes utilisés, nous renvoyons le lecteur aux travaux de T. Merabti [Merabti 2010].

6.2 Utilisation du Model Driven Architecture

Notre démarche se fonde sur la méthode d'Ingénierie Dirigée par les Modèles (*cf.* Section 2.3.2). Cette initiative propose dans un premier temps de concevoir un modèle indépendant du langage cible et dans un second temps de représenter ce modèle au travers d'un langage utilisé pour stocker, échanger ou mettre à disposition l'information. Cette démarche est illustrée en figure 6.1.

Cette méthode a l'avantage dans notre cas (i) de définir un modèle indépendamment des langages d'échanges qui peuvent être obtenus par transformation de modèles et (ii) de s'abstraire des exigences liées à un logiciel en particulier. Dans cette partie nous présentons notre modèle logique isolément de tout langage dédié à la connaissance et utilisons pour cela UML (Unified Modeling Language). Puis nous proposons une représentation de ce modèle utilisant un standard dédié à la représentation de la connaissance. Ce modèle est intégré à des outils et permet d'offrir des services que nous détaillerons au chapitre 7.

Le modèle UniMoKR que nous définissons va contraindre la représentation des modèles des SOC que nous voulons intégrer. Le périmètre du modèle est un compromis entre formalisation et flexibilité. Dans ce modèle nous essayons de factoriser et formaliser le plus grand nombre d'éléments récurrents dans les SOC. Toutefois notre modèle doit rester flexible pour autoriser les spécificités de ces référentiels au travers d'extensions. Cette flexibilité implique que les éléments ne soient pas trop contraints pour convenir à la plupart des SOC.

6.3 Construction d'un modèle unique de représentation

Notre première tâche a été de déterminer les éléments qui devaient ou non faire partie de notre modèle commun de représentation de SOC. Pour cela nous avons modélisé plusieurs de ces référentiels (*cf.* section 6.1). En comparant les modèles

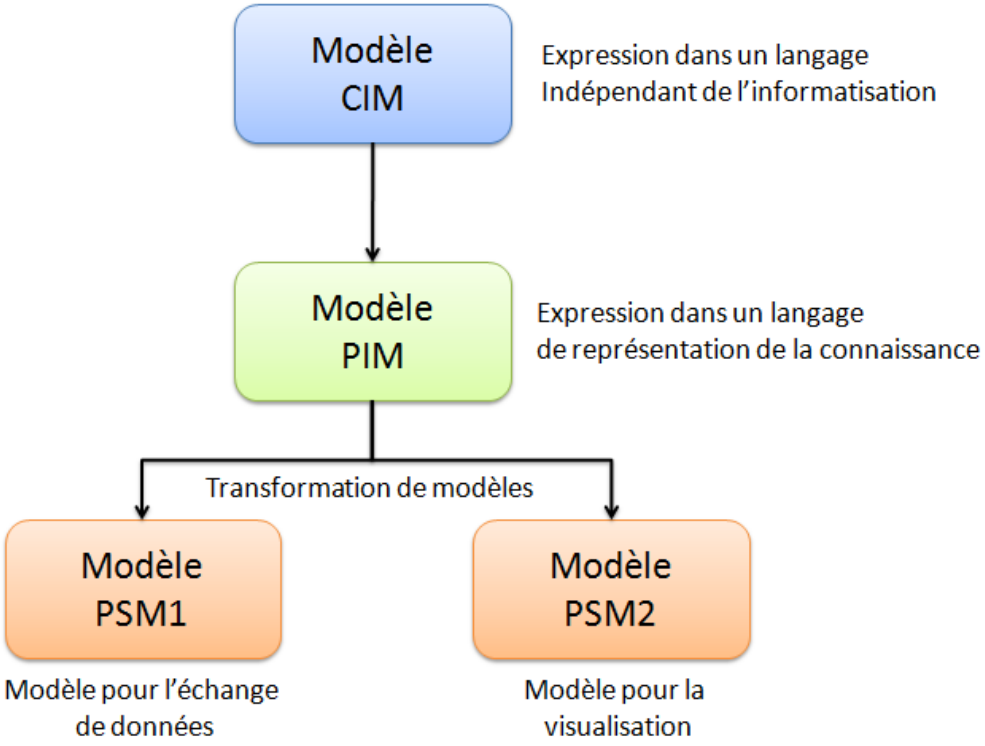

FIGURE 6.1 – Démarche Model Driven Architecture appliquée à nos travaux.

ainsi obtenus, nous étions en mesure de définir un métamodèle qui factorise les artefacts communs à tous les SOC. L'identification des éléments partagés par les différents référentiels s'appuie sur les standards existants (*cf.* section 4.3) et notre expérience dans la gestion de terminologies. Nous allons maintenant détailler les quatre différentes parties (Termino-conceptuelle, Groupement, Correspondances et métaclasses) qui constituent notre modèle présenté en figure 6.2.

6.3.1 Une modélisation organisée autour du concept

La difficulté de prétendre représenter de manière unifiée l'ensemble de ces SOC réside dans l'identification de l'élément fondamental de ces ressources. S'agit-il d'un mot, d'une occurrence, d'un terme ou d'un concept ? Chacune des structures et des utilisations, met plus ou moins en jeu les dimensions lexicale, linguistique et conceptuelle (*cf.* section 3.1). Or ces dimensions ne sont pas souvent clairement énoncées dans la description de ces SOC, ce qui engendre une grande confusion.

La discipline d'Ingénierie des Connaissances est un domaine de recherche récent issu de réflexions communes entre plusieurs disciplines dont la linguistique,

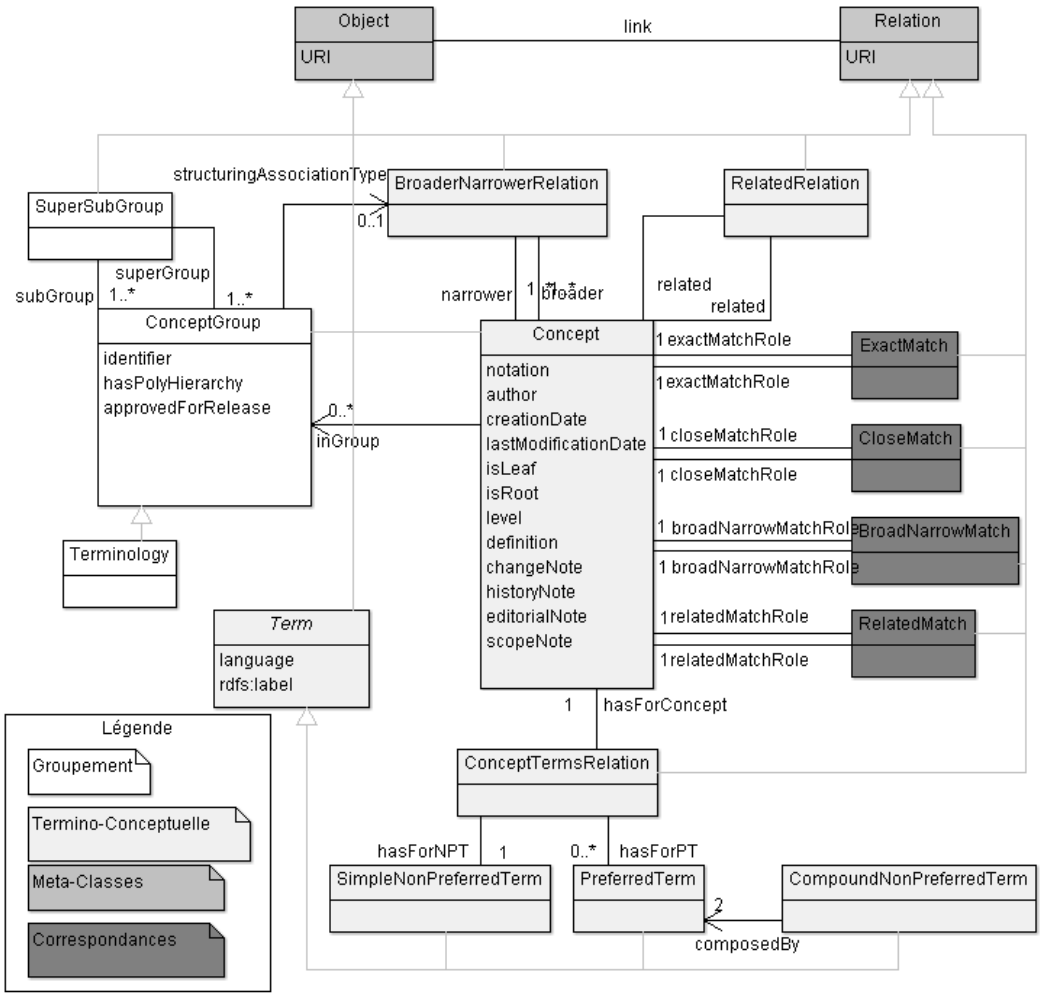

FIGURE 6.2 – Diagramme de classes UML de notre méta-modèle UniMoKR. Notre modèle comporte quatre parties : termino-conceptuelle, groupement, correspondances et métaclasses.

la terminologie, la psychologie, la logique mathématique, l'informatique. Ces réflexions, sous des points de vues différents, permettent de faire évoluer et d'enrichir le domaine. Mais paradoxalement, l'utilisation d'un vocabulaire commun conduit à des divergences conceptuelles, liées aux objectifs et à l'histoire de chacune de ces disciplines, qui rendent complexes les échanges et multiplient les malentendus [Aussenac-Gilles 2005a]. Savoir ce que représentent un concept, un terme et où se situe le sens a été l'objet de débats récurrents parmi les chercheurs travaillant en Ingénierie des Connaissances opposant réalistes et constructivistes [Merrill 2010, Smith 2004]. Ces différents choix de paradigmes influencent la modé-

lisation, la compréhension et l'utilisation des modèles. Nous allons donc commencer par définir sous quel paradigme nous construisons notre modèle.

Un texte est composé d'un ensemble de mots agencés de telle sorte qu'il est possible d'en extraire un sens. Ces mots sont contextualisés par le texte ou le discours dans lequel ils se situent. En effet, la forme de ces mots est fonction de leur position dans le texte, leur ton, leur style, leur rapport aux autres mots. L'analyse d'un texte nous révèle qu'une notion peut être portée non seulement par un mot mais également par un syntagme nominal, par des références internes et externes ou par l'implicite. Nous devons nous prémunir contre le danger que constitue le principe de « un mot-un concept » [Slodzian 2000]. En effet une expression comme « faire chou blanc » n'a rien à voir avec les légumes ni avec la couleur mais prise dans son ensemble, véhicule un sens.

Pour ces raisons, nous considérons qu'un mot ou groupe de mots ne peuvent pas entrer automatiquement dans la production d'un SOC et servir de référence ; un travail préalable est nécessaire.

Nous pensons que les termes présents dans les SOC, ne préexistent pas aux mots ou occurrences dans un texte mais sont construits à partir d'eux au moyen d'un processus de normativité terminologique. Un terme est donc un artefact des terminologues. Ce processus contient notamment les phases de nominalisation, de lemmatisation, de décontextualisation : « un mot devient un terme quand il n'a plus de passé, et qu'on lui attribue une signification indépendante des variations induites par les acceptions et les emplois en contexte » [Rastier 1995].

En poursuivant notre démarche sémasiologique qui nous amène à caractériser le sens en partant d'un syntagme nominal, nous étudions maintenant le rôle du concept. Alors que le terme est un produit pour une utilisation terminologique, il n'est pas adapté à la représentation d'une unité ontologique appelée concept. Toutefois un terme sert souvent de première étape vers la formalisation de celui-ci.

Le dictionnaire Oxford définit un concept comme « an idea or mental image which corresponds to some distinct entity or class of entities, or to its essential features, or determines the application of a term (especially a predicate), and thus plays a part in the use of reason or language ». Nous adoptons dans nos travaux la position constructiviste selon laquelle les concepts sont des constructions de l'esprit. Il advient de constater que les concepts ne se trouvent pas (au même titre que les termes) directement dans le texte mais sont construits par le lecteur en interaction avec le texte [Paquin 2010]. La représentation des concepts en Ingénierie des Connaissances se caractérise par la définition formelle de ses propriétés et de ses relations. Même si un concept existe hors du discours, il ne se révèle que par

son intermédiaire. Autrement dit, un concept a besoin d'un terme et de son lien au langage pour être intelligible et échangé. Cette vision rejoint celle du triangle sémiotique étudiée en section 3.1. La frontière entre le terme et le concept dans un SOC est parfois floue : certaines « terminologies » vont plus loin que la description de termes et définissent des relations formelles ; certaines « ontologies » apportent un engagement formel faible, se rapprochant d'une terminologie.

Georges Adamczewski fait une analyse de la définition de « concept » donnée par le Dictionnaire de l'Académie Française : « le concept, afin de servir de fondement, de principe ou d'idée explicative, se doit d'être minutieusement établi et ne pas varier avec le temps ou l'humeur du moment ». Ce point est nécessaire pour en faire un traitement computationnel. Nous considérons un concept comme étant [Adamczewski 2002] :

- par nature en constante mutation. Il prend place dans le temps et évolue selon l'environnement et les convictions d'une époque et d'un courant ;
- stable à un moment donné. Nous pouvons voir ce concept dans le temps comme une succession de clichés.

Grâce à la stabilité du concept à un instant déterminé, il nous est permis d'exprimer des connaissances sans subir les variations de la portée de ce concept dans le temps. C'est à cette condition qu'un SOC pourra servir de référence tout en continuant à évoluer vers de nouvelles versions figées. Nous considérons donc que l'élément central d'un SOC est le concept tel que décrit dans le standard SKOS (*cf.* section 4.3.1). Le concept est vu comme stable au sein d'une version de SOC pour prétendre être une référence. Autour de cet élément, nous avons des termes issus d'un processus de normalisation d'occurrences de mots ou syntagmes nominaux du langage. L'usage nous incite à identifier un terme préféré et des termes non préférés (synonymes) pour un concept comme nous le montre la figure 6.3. Le terme préféré *PreferredTerm* est considéré comme point d'entrée principal d'un concept *Concept* par un langage et comme le meilleur candidat pour désigner un concept dans une langue naturelle. Comme nous l'avons expliqué ci-dessus, un concept ne s'exprime que par le langage, c'est donc grâce à ce terme préféré qu'il le fait ; les termes non préférés *SimpleNonPreferredTerm* fournissent des alternatives d'expression dans le langage. L'usage de la post-coordination est courant dans les structurations de la connaissance. C'est à dire l'utilisation de plusieurs termes pour en décrire un nouveau comme par exemple « Bruit de moteur » composé des concepts « Bruit » et « Moteur ». C'est le rôle de l'entité *CompoundNonPreferredTerm* de représenter ces post-coordinations.

Les liens entre les entités concept, terme préféré et termes non préférés sont

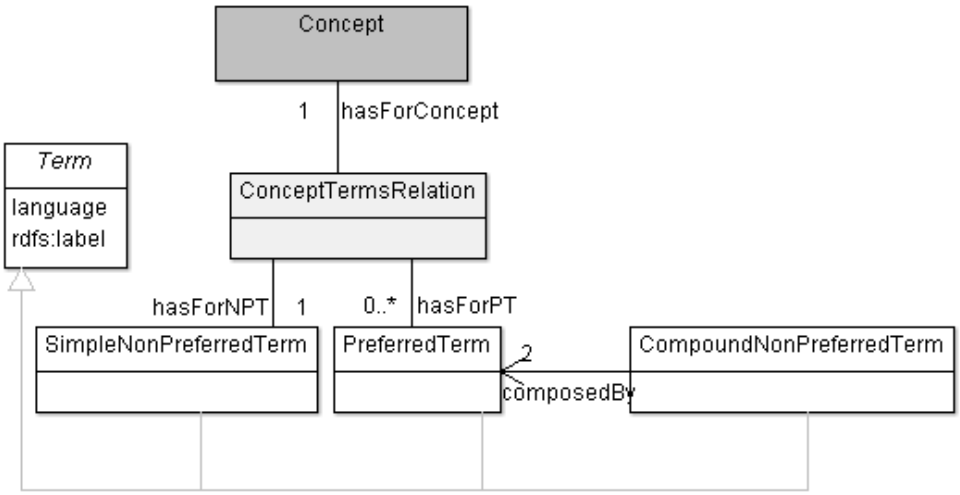

FIGURE 6.3 – Diagramme de classes UML des relations entre concepts et termes

représentés en figure 6.3 et sont adaptés du projet BS 8723 (*cf.* section 4.3.2). De même que l'ensemble des relations dans notre modèle, la relation *ConceptTermsRelation* est représentée sous forme de classes et non au moyen de propriétés (comme c'est le cas dans le projet BS 8723). Cette réification des relations a un double intérêt : (i) elle permet de mettre en relation plus de deux entités pour ainsi devenir des relations N-aires [Noy 2004] ; (ii) elle rend possible l'ajout d'attributs comme le créateur de cette relation. Ce dernier point confère à notre modèle la particularité d'être adapté pour l'échange mais également pour l'édition de SOC. C'est une différence avec le standard SKOS dont la finalité n'est pas l'édition mais plutôt la distribution de systèmes d'organisation de la connaissance. Ce patron de modélisation pour l'articulation des niveaux conceptuel et terminologique a été soumis [10] au portail Ontology Design Pattern (ODP) qui recense les patrons des meilleures pratiques de modélisation ontologique.

6.3.2 Représentation des alignements

Dans le cadre de la gestion conjointe de plusieurs SOC, le standard SKOS a défini des relations de correspondances que nous avons reprises dans notre modèle. Ces relations sont réifiées dans notre modèle pour des besoins éditoriaux des SOC. Il est possible, par exemple, de mentionner sur une correspondance entre concepts,

10. Voir http://ontologydesignpatterns.org/wiki/Submissions:ConceptTerms

quel algorithme a été utilisé pour sa génération, ou encore le degré de confiance. Les alignements entre SOC sont une composante de l'interopérabilité sémantique. Les alignements peuvent provenir de sources vérifiées, de logiciels d'alignements ou d'une mise en correspondance manuelle. Notre modélisation permet de représenter les correspondances issues de ces différentes méthodes ainsi que les possibles informations de métadonnées qui les accompagnent.

6.3.3 Représentation des groupes

Dans les domaines où la taille des terminologies est très grande (*e.g.* la médecine, la biologie, le droit ou la documentation), la possibilité de faire des groupes ou des collections de concepts est cruciale. La plupart des vocabulaires contrôlés sont historiquement et pour des raisons pragmatiques organisés en hiérarchie. En effet, retrouver une information dans une grande liste « à plat » reste très fastidieux. Mais la hiérarchie n'est pas la seule possibilité d'offrir un point de vue sur un SOC et les besoins de limitation des concepts mis à disposition sont nombreux ; parmi lesquels :

- contraindre les concepts disponibles pour un champ de saisi (appelé également « Value Set ») [Rector 2006] ;
- définir une taxonomie de navigation permettant d'accompagner les utilisateurs dans leurs expériences de recherche d'information ;
- masquer la complexité d'un SOC en ne présentant qu'une liste restreinte de concepts pertinents pour un cas d'utilisation précis.

Des standards et projets comme SKOS et BS 8723 permettent la représentation de SOC, mais aucun n'aborde complètement la notion de groupe telle que nous le proposons. Notre approche ne prétend pas pour autant définir un modèle *ex nihilo* mais se veut un paradigme de bonne pratique pour la représentation de groupes de concepts d'un ou de plusieurs référentiels. Notre méthode utilise et étend des parties de modélisation dans des standards et des projets existants qui ont déjà fait leurs preuves dans le domaine de la recherche ou dans l'industrie comme nous l'avons étudié au chapitre 4.

Le modèle proposé en figure 6.4 provient et s'inspire de nos expériences et des réflexions présentées dans notre état de l'art. Ce modèle présente les entités nécessaires à la représentation de groupements de concepts et peut être étendu (des sous-types de classes, de relations ou d'attributs peuvent être ajoutés).

Nous avons tout d'abord défini une entité nommée *ConceptGroup* qui peut être exportée ou échangée séparément de tout SOC. Cet artefact permet de représenter aussi bien un référentiel en entier, qu'un sous-ensemble. Nous avons défini deux moyens pour exprimer l'appartenance de concepts à un groupe :

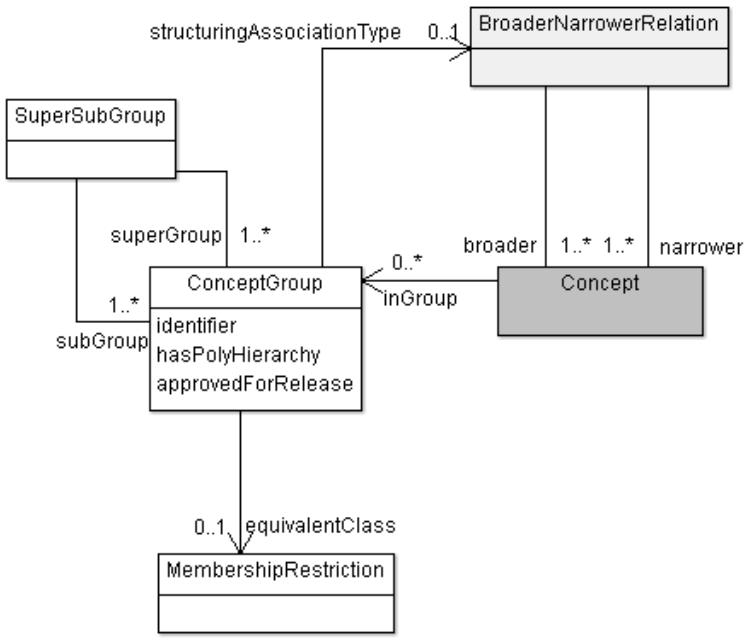

FIGURE 6.4 – Diagramme de classes UML pour la représentation de groupes de concepts.

par intension : font partie d'un groupe tous les concepts satisfaisant une restriction *MembershipRestriction*. La restriction peut être vue comme une requête sur le système où tous les concepts répondant à cette requête sont implicitement membres de ce groupe. Notons qu'aucun lien direct d'appartenance de type *inGroup* n'est utilisé dans ce cas-ci. Cette méthode permet de définir dynamiquement un groupe et facilite ainsi les évolutions des terminologies (tout nouvel objet qui répond à la requête est automatiquement compris dans la définition du groupe). La définition d'appartenance par intension est adaptée pour grouper des éléments partageant un certain nombre de propriétés (par exemple, « l'ensemble des figures géométriques ayant 3 côtés »). Elle n'est toutefois pas recommandée pour grouper des éléments disparates où il serait très difficile ou inefficace de créer une restriction (par exemple, le regroupement de la figure géométrique « triangle vert », de la figure « carré rouge » et de la figure « rond bleu »). Dans ce dernier cas, nous préférerons la méthode par extension présentée ci-dessous ;

par extension : font partie d'un groupe tous les concepts faisant explicitement référence à ce groupe au moyen du lien *inGroup*. Dans ce cas-ci, aucune classe *MembershipRestriction* n'est définie. Si un nouveau concept est ajouté à la

terminologie sans spécifier son appartenance, il ne sera pas membre de ce groupe.

Les groupes de concepts peuvent être organisés au sein d'une hiérarchie de groupe au travers de la relation *Super-SubGroup*.

Ce patron de modélisation pour le groupement de concepts a été soumis[11] au portail Ontology Design Pattern (ODP).

6.3.4 Utilisation de métaclasses

L'ensemble des entités des quatre parties présentées ci-dessus, est étendu à partir de classes très génériques Object, link, Relation. Ces métaclasses sont le moyen de garantir l'extensibilité du modèle.

6.4 Artefacts spécifiques

Certains artefacts spécifiques à certains SOC, non pris en compte dans le modèle UniMoKR, doivent néanmoins être représentés afin de ne pas perdre d'information. En conséquence, un juste milieu doit être trouvé dans l'élaboration de notre modèle unificateur afin de représenter fidèlement les référentiels tout en factorisant le plus d'éléments communs. Nous illustrons l'extension spécifique à la terminologie CIM10 en figure 6.5. Nous avons pour cela défini une classe abstraite *CIM10Concept* qui représente un concept de la terminologie *CIM10* et qui étend la classe *Concept* du modèle commun. La classe *CIM10Concep*t hérite donc des propriétés et des relations de la classe *Concept* et permet de spécifier la relation *CIM10Exclusion* spécifique à *CIM10* (qui étend la métaclasse *Relation* du modèle commun), mais également de définir des attributs particuliers à cette terminologie. Ensuite, pour respecter la terminologie, nous avons défini une sous-classe de *CIM10Concept* pour chaque niveau de la hiérarchie de concepts. Outre le respect de la terminologie, cette architecture permet de prendre en compte les attributs spécifiques à chaque niveau comme par exemple le chapitre de la classification *CIM10chap*.

Cette modélisation respecte les bonnes pratiques de construction de SOC. Comme le montre la figure 6.6, les connaissances constitutives de la terminologie *CIM10* sont représentées comme des instances. Notons que la relation hiérarchique entre les concepts est représentée par une instance de la relation réifiée « Broader-NarrowerRelation » (qui représente un lien de subsomption ou de partition) et non par une relation de subsomption. Ceci évite des erreurs fréquentes liées à la transitivité de la relation de subsomption dans le cas où les concepts doivent être liés par

11. Voir http://ontologydesignpatterns.org/wiki/Submissions:ConceptGroup

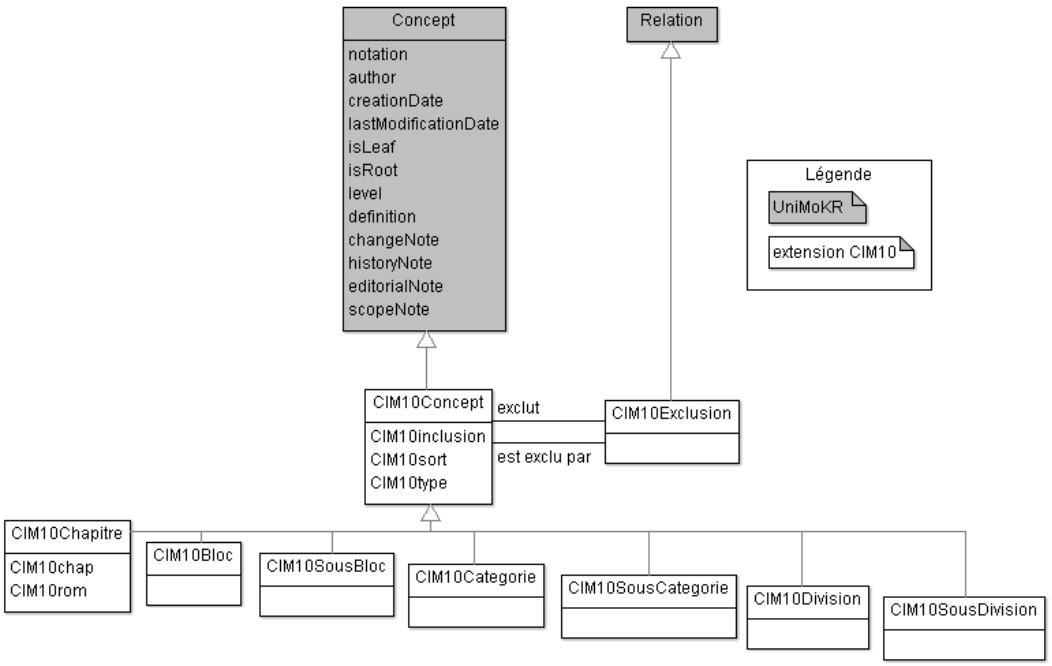

FIGURE 6.5 – Diagramme de classes UML de l'extension du modèle pour la terminologie CIM10.

une relation de partition [Aitken 2003]. Pour certains projets comme InterSTIS, nous avons mis en place une version simplifiée de notre modèle UniMoKR dans laquelle les termes sont représentés comme des attributs sur le concept. Cette simplification reflète la faible expressivité linguistique des SOC utilisés dans ce projet et permet en simplifiant le modèle, de faciliter également la maintenance des connaissances.

6.5 Langage de représentation

Au sortir de cette première étape, nous avons une description de l'arrangement entre les artefacts de notre modèle indépendant de la syntaxe finale de représentation. La seconde étape de notre méthode consiste en l'utilisation d'un ou plusieurs langages de représentation physique pour stocker, échanger et présenter le contenu de nos structurations de la connaissance. Plusieurs ensembles d'actions sur les SOC peuvent être identifiés (*cf.* figure 3.10) et pour chacun d'eux, des langages de représentation sont adaptés. Ainsi pour des actions d'édition (création, maintenance, traduction, alignement) nous avons choisi le langage OWL DL pour représenter notre modèle. Comme nous l'avons vu en section 4.2.2.3, ce langage a l'avantage (i) d'être communément utilisé dans les applications du web sémantique ; (ii) d'être fortement

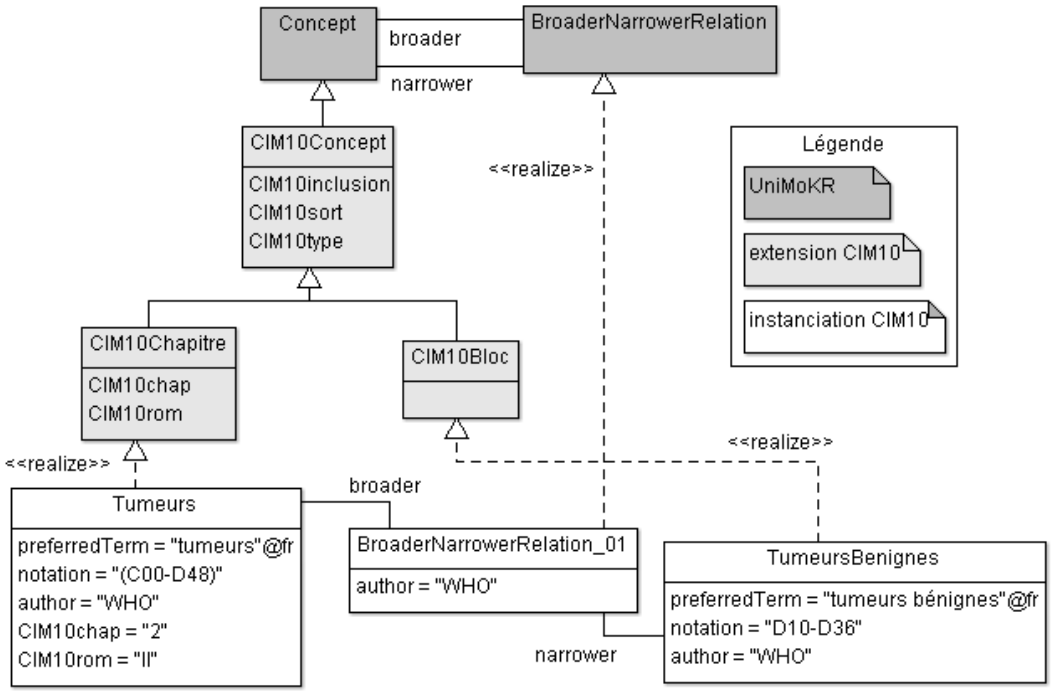

FIGURE 6.6 – Diagramme de classes UML de l'instanciation du modèle étendu CIM10.

outillé et donc de favoriser l'utilisation d'une ressource fondée sur ce langage ; (iii) de posséder plusieurs niveaux d'expressivité dont le niveau DL (Description Logic) qui utilise une sémantique formelle basée sur la logique des prédicats. A cet égard, OWL DL convient parfaitement à l'expressivité formelle nécessaire aux tâches d'édition et à l'interopérabilité sémantique souhaitée. OWL EL optimisé pour les terminologies est toutefois trop limitant pour l'expression de SOC plus complexes (*cf.* section 4.2.2.3).

Dans un système d'information, les SOC peuvent être instanciés pour représenter des cas réels tels que « l'appendicite de M. X » instance du concept « Appendicite ». Sur la base de ces cas, des raisonnements peuvent faire apparaître des comportements ou des conclusions intéressants. Toutefois, ces raisonnements ne sont possibles que dans la mesure où ces SOC possèdent une sémantique formelle. Si on prend l'exemple de la CIM-10, cette classification ne présente pas de sémantique formelle et a été conçue pour annoter des cas patients et non pour permettre le raisonnement. La description de notre modèle UniMoKR par le langage OWL permet l'affinement des types de classes et d'attributs pour modéliser un SOC (extension du modèle) et la représentation du contenu des SOC en tant qu'instance de ce modèle. Ce modèle que nous proposons est une ontologie bien formée pour représenter des SOC. Son

périmètre se limite toutefois à cette représentation et ne permet ni la représentation, ni le raisonnement sur des instances de SOC telles que « l'appendicite de M. X » instance du concept « Appendicite ». En reprenant l'exemple de la classification CIM-10, son intégration à notre modèle ne fait pas pour autant de ce SOC une ontologie. Cette intégration permet de préciser la sémantique des relations et des attributs qui composent la classification. Pour obtenir une ontologie bien formée à partir de la CIM-10, il faudrait par exemple différencier la relation *BroaderNarrowerRelation* en une relation transitive *is-a* et une relation partitive *part-of*. Cette modification changerait l'intention de ce SOC et donc son utilisation. Notre modèle UniMoKR a donc pour intention de représenter, stocker et mettre à disposition des SOC en formalisant leur expression sans modifier leur nature. Nous travaillons à un niveau méta d'abstraction par rapport aux SOC.

Pour des problématiques de distribution et d'utilisation, il est parfois demandé de fournir un SOC dans un format standard d'échange tel que SKOS ou encore au travers de web services comme CTS2 (*cf.* section 4.4.2). Pour exporter ces ressources, nous utilisons la méthode de transformation de modèles appliquée aux langages du web sémantique reposant sur RDF [Polleres 2007, Morbidoni 2007]. Cette capacité de transformation, facilitée par la confrontation de notre modèle aux standards existants, permet actuellement d'exporter les SOC aux formats SKOS, CTS2, IHE-Lab51. Ces transformations reposent sur l'utilisation de règles exprimées en SPARQL et sont détaillées au chapitre suivant.

Troisième partie

De l'utilisation de nos travaux

Chapitre 7

Intégration, services et interfaces

Sommaire

Ce chapitre présente les outils et techniques qui permettent à notre modèle de proposer des services aux utilisateurs. Nous commençons en section 7.1 par décrire notre méthode de transformation de modèle à l'aide du langage SPARQL. Cette technique, garante de l'interopérabilité, est primordiale dans l'environnement hétérogène de description des SOC. Toutefois cette méthode présente des limites que nous mettons en évidence. Nous détaillons en section 7.2 l'intégration de notre modèle à l'outil ITM. Notre travail de recherche, en collaboration avec l'entreprise MONDECA, a permis l'amélioration de la représentation des connaissances, des services et des interfaces du logiciel ITM que développe cette entreprise. Après avoir présenté l'outil, nous proposons des améliorations liées à l'intégration de notre modèle. Enfin, nous détaillons en section 7.3 l'intégration de notre modèle à un logiciel de stockage de triplets RDF.

7.1 Méthode de transformation de modèles

Les SOC sont disponibles et utilisés dans divers formats. Pouvoir les importer ou les exporter vers ou depuis notre modèle est primordial. Alors que certains formats comme Excel sont spécifiques et nécessitent une transformation particulière, les formats du Web sémantique (exprimés en RDF) peuvent être générés en effectuant des transformations de modèles exprimés avec UniMoKR.

7.1.1 Processus de transformation

7.1.1.1 Description générale

Dans nos travaux, nous avons utilisé une méthode de transformation de modèles (*cf.* section 2.3.2.2) qui repose sur le langage SPARQL (*cf.* section 4.4.1) pour représenter les règles de transformation. L'utilisation de requêtes SPARQL pour transformer les graphes RDF n'est pas nouvelle en soi : A. Polleres *et al.* ont décrit comment SPARQL peut servir aux passages entre vocabulaires exprimés en RDF [Polleres 2007]. Le projet DERI Pipes (sur lequel A. Polleres *et al.* s'appuient) a produit un moteur et une interface utilisateur graphique pour la transformation de données Web à base de transformation de graphes [Morbidoni 2007]. Nous exploitons aussi le potentiel de la clause « CONSTRUCT » de SPARQL. Cependant notre approche est différente : nous utilisons SPARQL dans sa spécification actuelle (1.0) [1] intégrée à un programme de traitement (les projets existants utilisent quant à eux, des extensions spécifiques de SPARQL ce qui rend impossible l'intégration de leur solution à un moteur SPARQL actuel). Notre programme de traitement sert à pallier les limites de SPARQL dans sa version actuelle, mais aussi à simplifier l'implémentation (grâce à des librairies existantes) et l'écriture des règles.

La figure 7.1 présente le schéma d'architecture de traitement des règles et de transformation du modèle que nous proposons. Notre moteur de transformation s'appuie sur un moteur de règles SPARQL externe auquel il est possible d'ajouter des traitements particuliers. Le moteur de transformation utilise un ensemble de règles *RuleSet* et un modèle source à transformer. Les règles sont ensuite interprétées par le moteur d'interprétation *RuleSet Parser* et appliquées au modèle source après des pré-traitements optionnels (comme l'application de règles d'inférence). L'exécution des règles génère un graphe temporaire *Temporary RDF statements*. Ce dernier se voit potentiellement appliquer des post-traitements tels que la concaténation de valeurs. Le résultat de cette opération est la production d'un modèle cible qui peut

1. Au moment de l'écriture de ce mémoire, la version 1.1 de SPARQL est en cours de standardisation et commence à être adoptée par les outils sesame et jena que nous utilisons ci-après.

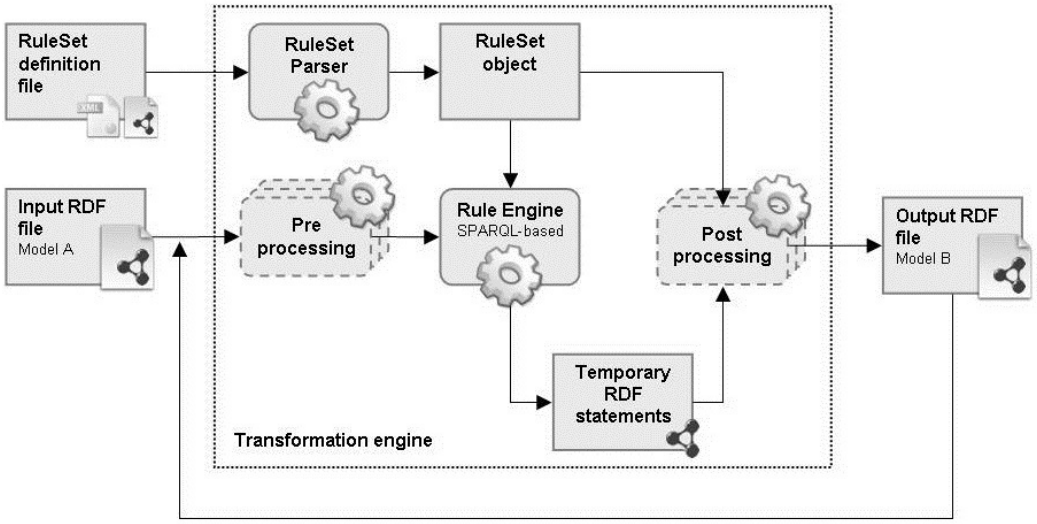

FIGURE 7.1 – Architecture du moteur de transformation SPARQL mis en place.

à son tour servir de source pour une nouvelle transformation.

Les avantages de cette architecture sont :

- de découpler l'interprétation des règles de leur exécution ; ceci permet l'utilisation de multiples formats de règles (XML, RDF, etc.) et facilite l'intégration directe de règles quand elles sont générées automatiquement ;
- d'utiliser un moteur de règles compatible avec SPARQL dans sa version 1.0 (Sesame ou Jena [Schenk 2008, Grobe 2009]) sans nécessiter d'extension au langage. De plus, la mise en place d'une étape de post-traitement manipulant directement le graphe temporaire, permet de simplifier et de réduire les coûts d'implémentation ;
- de permettre l'application de pré-traitements avant l'application des règles de transformation comme des procédures de nettoyage de données ou d'inférence ;
- de faire succéder plusieurs groupes de règles de transformation avant d'aboutir au modèle final. Ceci est très pratique lorsque l'on veut dans une première étape filtrer les données qui serviront de base à la transformation vers le modèle final.

7.1.1.2 Traitement des règles

Dans la suite de ce mémoire nous utilisons les préfixes suivants pour décrire les noms d'espaces associés :

```
@prefix ev: <http://eurovoc.europa.eu/schema#> .
@prefix eu: <http://eurovoc.europa.eu/> .
```

```
@prefix t3: <http://www.mondeca.com/system/t3#language> .
@prefix oc: <http://www.mondeca.com/system/ontology_creation#> .
@prefix iso: <http://psi.oasis-open.org/iso/639/#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix skos: <http://www.w3.org/2004/02/skos/core#> .
@prefix xl: <http://www.w3.org/2008/05/skos-xl#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
```

Une règle est composée des éléments suivants :

- un identifiant unique ;
- une description textuelle de son objectif ;
- la règle « CONSTRUCT » SPARQL ;
- des filtres facultatifs de post-traitement.

Dans sa forme la plus simple, une règle de transformation écrite en XML se présente comme ceci [2] :

```
<ruleSet>
    <rule id=\"A2-BN\">
        <description>Translation of the Eurovoc BN relations
        into SKOS broader properties.</description>
        <expression><![CDATA[
            PREFIX skos:<http://www.w3.org/2004/02/skos/core#>
            PREFIX t3:<http://www.mondeca.com/system/t3#>
            CONSTRUCT { ?n  skos:broader ?b }
            WHERE {
                ?a  a t3:BN .
                ?a  t3:bt  ?b.
                ?a  t3:nt  ?n.
            }
        ]]></expression>
    </rule>
    <!-- other rules here -->
</ruleSet>
```

Pour faciliter l'écriture des règles, nous avons mis en place des simplifications d'écriture.

Groupement des préfixes. Puisqu'elles s'appliquent à un même graphe source, toutes les règles utilisent les mêmes préfixes. Nous permettons donc la déclaration groupée des préfixes communs à toutes les règles d'un fichier de règles. Ceci évite de recopier les préfixes pour chaque règle et améliore leur lisibilité.

2. La règle « A2-BN » (BN pour Broader Narrower ; t3 fait référence à un nom d'espace de MONDECA que nous réutilisons pour l'identification de certains éléments de notre modèle UniMoKR) a pour objectif la transformation des entités b et n (liées à une instance de la classe *t3:BN* au moyen, respectivement, des prédicats *t3:bt* et *t3:nt*) en un triplet dans lequel n a pour concept plus général b.

Utilisation des opérateurs « from »,« to ». L'utilisation de ces opérateurs permet de simplifier la lecture des règles et factoriser les antécédents ou les conséquences de règles. Il est donc possible d'écrire une règle avec plusieurs antécédents et une conséquence [3]. Les deux parties « CONSTRUCT » et « WHERE » de chaque règle peuvent être écrites respectivement dans la partie « to » et « from ». Ces deux premières simplifications d'écriture sont présentées dans l'exemple suivant [4] :

```
<ruleSet>
    <common-prefixes><![CDATA[
            PREFIX skos:<http://www.w3.org/2004/02/skos/core#>
            PREFIX t3:<http://www.mondeca.com/system/t3#>
    ]]></common-prefixes>
    <translate id="A2-BN">
        <description>Translation of the Eurovoc BN relations
        into SKOS broader properties.</description>
        <from><![CDATA[
            ?a  a t3:BN .
            ?a  t3:bt  ?b.
            ?a  t3:nt  ?n.
        ]]></from>
        <to>?n  skos:broader ?b</to>
    </translate>
    <!-- other rules here -->
</ruleSet>
```

Modification simple de triplets. Bien qu'une règle de transformation puisse être complexe, il est courant qu'une règle serve simplement à modifier tous les triplets avec un sujet, prédicat ou objet précis. Cette écriture est possible dans notre programme et permet par exemple de recopier tous les triplets possédant le prédicat « `rdfs:label` » en modifiant ce prédicat par « `skos:prefLabel` ».

7.1.1.3 Post-traitements

Une règle de transformation peut-être associée à une chaîne ordonnée de post-traitements. Chaque post-traitement fait appel à une classe JAVA qui, à partir de triplets en entrée, produit des triplets en sortie. Les triplets générés par les règles sont soumis à cette chaîne ordonnée de post-traitements pour produire un graphe final. L'utilisation de ces post-traitements est une manière de compenser les limites de la version actuelle (1.0) de SPARQL que nous détaillerons dans la section 7.1.3.

3. Cette solution est plus performante que de faire une union des antécédents dans la requête.

4. La simplification d'écriture n'implique pas forcément d'avoir une expression plus courte mais vise à rendre l'écriture et la lisibilité des règles plus facile par un utilisateur. L'utilité des simplifications d'écriture devient évidente lorsque l'on crée un jeu de plusieurs dizaines de règles de transformation.

Les post-traitements disponibles dans notre architecture permettent d'effectuer les modifications suivantes sur les triplets RDF :

- appliquer un « chercher et remplacer » dans les URI des Ressources ;
- typer une valeur ;
- modifier une valeur littérale en fonction de son information de langue ;
- supprimer, modifier ou ajouter une information de langue à une valeur littérale ;
- remplacer la ressource objet d'un triplet par une valeur littérale formée à partir de son URI.

Les post-traitements sont déclarés séparément de la définition des règles dans lesquels ils sont référés :

```
<translate id="DataType4-definition-concepts">
    <description>Export of the 'definition' attribute</description>
    <from><![CDATA[
        ?x  skos:definition  ?y .
    ]]></from>
    <to>
        ?x  skos:definition  _:b.
        _:b  ev:noteLiteral  string(?y).
        _:b  ev:language  lang(?y).
    </to>
    <postprocessor>replaceObjectWithLanguage</postprocessor>
    <postprocessor>deleteNoteLiteralLanguage</postprocessor>
    <postprocessor>addNoteLiteralDatatypeCollector</postprocessor>
</translate>
```

Dans ce dernier exemple, trois post-traitements sont appliqués aux triplets générés par la règle : (i) remplacer l'objet ayant pour prédicat *ev: language* par l'information de langue ; (ii) supprimer l'information de langue aux objets ayant pour prédicat *ev: noteLiteral* ; et (iii) typer les objets ayant pour prédicat *ev: noteLiteral*. Le tableau 7.1 présente le résultat de la transformation par étapes d'un triplet en appliquant la règle *DataType4-definition-concepts* et ses post-traitements. Les tableaux de ce chapitre présentent des connaissances exprimées sous forme de triplets au format Notation 3 [5].

7.1.2 Exemple de transformation du thésaurus Eurovoc de notre modèle vers SKOS

Dans le cadre du projet Eurovoc (*cf.* section 8.3), nous avons dû mettre en place un export du thésaurus au format SKOS. Le métamodèle cible SKOS étendu pour Eurovoc nous a été donné. Nous avons utilisé notre méthode de transformation de

5. Voir : http://www.w3.org/DesignIssues/Notation3

TABLEAU 7.1 – Résultat de l'application de la règle de transformation *DataType4-definition-concepts* et de ses post-traitements.

graphe RDF initial	`ev:termA skos:definition` « definition »@fr .
Résultat après application de la règle	`ev:termA skos:definition _:b` . `_:b ev:noteLiteral` « definition »@fr . `_:b ev:language` « definition »@fr .
Résultat après post-traitements	`ev:termA skos:definition _:b` . `_:b ev:noteLiteral` « definition »^^`rdf:XMLLiteral` . `_:b ev:language` « fr » .

modèle pour générer le modèle SKOS étendu d'Eurovoc. Cette transformation a nécessité l'application de 47 règles. Détaillons les règles qui concernent les relations entre concepts et termes illustrées en figure 7.2.

Les règles de transformation identifiées par le numéro 1 (1, 1' et 1") sont de simples transformations d'un élément en un autre. Les règles de transformation numéros 2 et 3 sont quant à elles plus compliquées :

- la règle de transformation #2 met en exergue la différence de niveau d'expression des deux métamodèles (UniMoKR et SKOS). Cette différence conduit à de possibles réductions de sens et dans ce cas à une plus grande difficulté de maintenance (causée par une redondance de l'information) ;
- la règle de transformation #3 révèle les difficultés de prise en compte de l'information de langue par le langage SPARQL dans sa version 1.0. Il est en effet impossible dans une règle de positionner une information de langue sur une valeur littérale sur la base d'une variable. Cette limite de SPARQL 1.0 à manipuler les URI et les chaînes de caractères sera discutée dans la section suivante.

7.1.3 Des limites et des contournements de SPARQL 1.0.

Durant l'écriture de nos règles de transformation, nous avons rencontré des limites liées à l'expressivité du langage SPARQL dans sa version courante 1.0. La plupart de ces obstacles, également repérés dans les travaux similaires présentés en début de section, ont été transmis au groupe de travail sur la future version de SPARQL (1.1). Les post-traitements nous ont permis, en attendant ces futures améliorations, de contourner ces obstacles [6].

6. Des outils comme Sesame ou Jena commencent à anticiper l'attribution du statut de standard à la version 1.1 de SPARQL et implémente déjà certaines fonctionnalités.

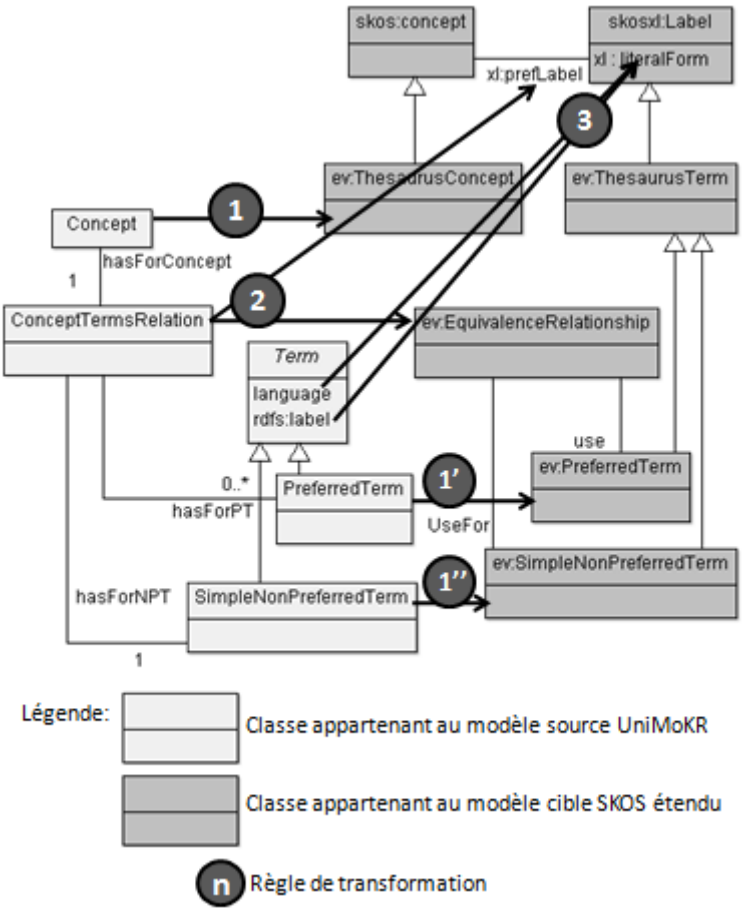

FIGURE 7.2 – Exemple de transformation de modèle depuis UniMoKR vers SKOS étendu pour Eurovoc. Les règles de transformation numéro 1, 1' et 1" sont des transformations simples entre deux éléments, la numéro 2 est une transformation d'un élément en plusieurs et la règle 3 est une transformation de plusieurs éléments en un (fusion des propriétés de libellé et de langue). Tous les éléments de ce schéma ont un identifiant unique (URI) et possèdent d'autres attributs et relations masqués pour une meilleure lisibilité.

Limite 1 - Construction d'URI. Considérons dans le tableau 7.2 le graphe RDF source et le graphe cible. Dans cet exemple, nous voulons réifier le libellé de `ev:TermA` et construire une nouvelle ressource RDF en formant son URI par la concaténation de l'URI `ev:TermA` et de la valeur de langue *fr*. Dans cet exemple atomique nous pourrions utiliser un nœud blanc (blank node) au lieu de construire un nouvel URI mais cette solution ne permettrait pas la réutilisation de cette ressource dans plusieurs règles. En effet, l'identifier de manière unique nous permet d'y faire référence dans plusieurs règles et ainsi compléter les informations qui concernent cette ressource. Le graphe cible que nous pro-

Tableau 7.2 – Limite 1 des transformations de modèles à base de SPARQL.

Graphe RDF source	Graphe RDF cible
`ev:termA t3:language iso:fra .` `iso:fra iso:code-a2` « fr » . `ev:termA rdfs:label` « Label A » .	`ev:termA xl:literalForm ev:termAfr .` `ev:termAfr rdfs:label` « Label A » .

Tableau 7.3 – Limite 2 des transformations de modèles à base de SPARQL

Graphe RDF source	Graphe RDF cible
`ev:termA skos:definition` « definition »@fr .	`ev:termA skos:definition _:b .` `_:b ev:noteLiteral` « definition »^^`rdf:XMLLiteral` . `_:b ev:language` « fr » .

posons n'est pas productible à partir de la version actuelle de SPARQL. En effet il est impossible de (i) manipuler une URI comme une valeur littérale et (ii) concaténer des chaînes de caractères pour produire une valeur littérale. Cette limite va être résolue par deux fonctions du futur langage SPARQL 1.1 que sont : les expressions de projet [W3C 2009b] et les agrégations [W3C 2009a].

Limite 2 - Manipulation de langues et des types de données. Considérons dans le tableau 7.3 le graphe RDF source et le graphe cible. Dans cet exemple, nous voulons réifier le prédicat `skos:definition`. Cette opération n'est pas possible dans la spécification courante de SPARQL puisque l'information de langue ne peut pas être manipulée comme une valeur littérale et ainsi être supprimée, ajoutée ou modifiée. De la même manière, il n'est pas possible de manipuler les types de données. Cette limite va être résolue par la fonction d'expressions de projet [W3C 2009b] du futur langage SPARQL 1.1.

Limite 3 - Opérateur de dénombrement « COUNT » Considérons dans le tableau 7.4 le graphe RDF source et le graphe cible. Dans cet exemple, nous voulons compter le nombre d'occurrences de la propriété `skos:definition` pour une ressource et stocker ce nombre en valeur d'une propriété. La version 1.0 de SPARQL ne permet pas l'utilisation de fonction d'agrégation (« count », « min », « max », « avg », etc.). Cette limite va être résolue par la fonction d'agrégation [W3C 2009a] proposée dans la future version (1.1) de SPARQL.

Limite 4 - Manipulation de listes. Considérons dans le tableau 7.5 le graphe RDF source et le graphe cible. Dans cet exemple, nous transformons un modèle OWL en un modèle proche du format des Topic Maps. Dans le graphe cible,

Tableau 7.4 – Limite 3 des transformations de modèles à base de SPARQL

Graphe RDF source	Graphe RDF cible
`ev:termA skos:definition` « definition »@fr . `ev:termA skos:definition` « definition »@en . `ev:termA skos:definition` « definition »@de . `ev:termA skos:definition` « definition »@it .	`ev:termA ev:definitionCount` « 4 »^^xsd :decimal .

nous voulons lier le domaine de la propriété `ev:definition` à la liste des classes `ev:Concept` et `ev:Term`. Dans le langage OWL (graphe source), cette définition se fait grâce à une union d'éléments d'une liste. Le langage SPARQL manque de fonctions pour manipuler simplement des listes ; les éléments `rdf:first` et `rdf:rest` doivent être pris en compte explicitement. Ceci rend impossible la gestion de listes dont le nombre de composants n'est pas connu. Actuellement il n'est pas possible d'utiliser une fonction récursive et nous devons tester si un élément existe en premier, ou en second, ou en troisième, ou en quatrième, etc. Voici un exemple de cette requête illustré sur le graphe source uniquement (partie « WHERE ») limité aux trois premiers éléments de la liste :

```
CONSTRUCT { <snip> }
WHERE {
    {
        { ?x a owl:DatatypeProperty }
        UNION
        ?x rdfs:domain ?d .
        ?d a owl:Class.
        ?d owl:unionOf ?union .
    {
        { ?union rdf:first ?y }
        UNION {
            ?union rdf:rest ?union2 .
            ?union2 rdf:first ?y
        }
        UNION {
            ?union rdf:rest ?union2 .
            ?union2 rdf:rest ?union3 .
            ?union3 rdf:first ?y
        }
        <etc.>
    }
}
```

Cette limite va être résolue par la fonction de chemin de propriétés [W3C 2010] proposée dans la future version (1.1) de SPARQL.

Tableau 7.5 – Limite 4 des transformations de modèles à base de SPARQL

Graphe RDF source	Graphe RDF cible
ev:definition a owl: DatatypeProperty .	
ev:definition rdfs:domain _:b .	_:b1 a oc:C.O.C_at .
_:b a owl:Class .	_:b1 oc:c.o.c_occurrence_rt ev:definition .
_:b owl:unionOf _:l1 .	_:b1 oc:c.o.c_class_rt ev:Concept .
_:l1 rdf:first ev:Concept .	_:b2 a oc:C.O.C_at .
_:l1 rdf:rest _:l2 .	_:b2 oc:c.o.c_occurrence_rt ev:definition .
_:l2 rdf:first ev:Term .	_:b2 oc:c.o.c_class_rt ev:Term .
_:l2 rdf:rest rdf:nil .	

7.1.4 Synthèse

La méthode que nous venons de détailler permet de transformer des SOC représentés grâce au modèle UniMoKR depuis et vers des modèles exprimés en RDF. Ceci nous permet de garantir une interopérabilité avec les langages du Web sémantique. Notre méthode repose sur le langage SPARQL dans sa version actuelle (1.0). Pour pallier les limites de ce langage et faciliter l'application des transformations, nous avons développé un programme informatique. Ce dernier permet d'appliquer des pré-traitements et post-traitements en plus des règles de transformation. Il permet également l'enchainement de plusieurs transformations successives. Même si la future version de SPARQL propose des corrections à ces limites, notre approche restera valable et utile. En effet, l'application de traitements avant ou après l'application des règles de transformation et l'enchainement de plusieurs transformations est complémentaire aux transformations.

7.2 Intégration à l'outil ITM

Notre modèle sert à la représentation de SOC mais ne propose pas nativement de services ou interfaces pour les gérer. C'est pourquoi nous intégrons le modèle UniMoKR à un outil qui peut offrir ces services pour interagir avec les SOC et la connaissance. L'outil ITM développé par MONDECA remplit ce rôle et sert à exprimer, représenter, stocker et exploiter des connaissances. Ces connaissances sont exprimées sous la forme d'un graphe et stockées dans un système de gestion de base de données relationnelle. Nous débutons par la présentation d' ITM avant de détailler

les apports de nos travaux à cet outil.

7.2.1 Présentation de l'outil

L'outil ITM est une application Web léger, c'est à dire qui ne nécessite pas l'installation d'un logiciel côté utilisateur si ce n'est un navigateur Web. Une description détaillée des interfaces de cet outil est donnée en annexe B, page 185.

La connaissance au sein de l'outil ITM est structurée en trois niveaux d'abstraction articulés par la relation classe-instance. Cette architecture est comparable aux niveaux de méta-modélisation de la démarche MDA (*cf.* section 2.3.2.1) :

- **le niveau méta** décrit le métamodèle réflexif des connaissances d'ITM ;
- **le niveau modèle** est défini en utilisant les primitives du niveau méta. Ce niveau est composé d'un modèle ontologie de domaine spécifiant la sémantique du vocabulaire utilisé pour décrire (au niveau sémantique) le contenu des informations gérées par la base. A cette ontologie de domaine peut être associée une « ontologie de services » spécifique à ITM et spécifiant la sémantique de comportements propres à l'outil ITM. Cette dernière ontologie permet par exemple de spécifier un patron de construction d'un libellé par concaténation d'autres valeurs ;
- **le niveau instances** contient les instances de la base qui sont contraintes par le modèle du niveau supérieur. On y trouve des annotations sémantiques qui décrivent le contenu des informations gérées par la base, des ressources terminologiques, la description de l'organisation d'un hôpital en structures, etc.

Le métamodèle du niveau méta varie très peu. Ce vocabulaire, avec l'ontologie de services, sont interprétés par ITM pour doter les bases de connaissances de la sémantique opérationnelle propre à l'outil.

7.2.2 L'intégration de connaissances à l'outil ITM

Le logiciel ITM est une application Web dont toutes les actions possibles sont présentées au travers d'API (Application Programming Interface) et de services Web. L'import d'un SOC se conformant à notre modèle UniMoKR passe par trois étapes successives :

1. l'import du modèle UniMoKR dans l'espace modèle d'ITM ;
2. l'import de l'extension du modèle UniMoKR pour la représentation des particularités du SOC dans l'espace modèle d'ITM ;

3. l'import du contenu du SOC dans l'espace instances d'ITM. Au cours de cette étape, la conformité des instances au(x) modèle(s) est vérifiée.

L'import d'un nouveau SOC est possible à tout moment en reproduisant les étapes 2 et 3. Ces connaissances sont importées au format standard OWL et stockées dans une base de données relationnelle grâce à une vue logicielle responsable de la transformation des données OWL vers la base de données. Des approches similaires sont décrites dans la littérature [Bellatreche 2004, Vysniauskas 2006].

7.2.3 Nos apports à l'outil

Le modèle. Le premier apport à l'outil ITM est de le doter d'un modèle capable de répondre aux besoin de représentation des clients de MONDECA. Cet apport sera argumenté et illustré au chapitre 8 grâce aux nombreux projets d'application que nous avons pu mener.

Les interfaces et comportements. Un second apport concerne les interfaces et comportements de l'outil. Notre modèle se propose d'offrir un noyau commun de représentation de SOC. Intégré à l'outil ITM, il est donc une partie du niveau modèle qui reste stable quel que soit le projet ou le domaine modélisé. Grâce à sa stabilité, il est possible d'associer des interfaces et des comportements dynamiquement contrôlés par des entités de notre modèle. Nous avons apporté des modifications aux interfaces utilisateurs pour répondre aux besoins exprimés par les clients et pour afficher des informations plus pertinentes. C'est le cas de la visualisation des instances de la classe *Group* affichées en figure 7.3 sous l'appellation « Vocabulary ». Cette figure présente une vue générique construite à partir des instances de la classe *Group* ou de ses sous-classes. L'imbrication des groupes par la relation *SuperSubGroup* est naturellement transcrite par une hiérarchie navigable d'entrées dans les interfaces. Nous avons également ajouté un nouveau comportement associé à l'accès d'un groupe dans cette hiérarchie. A cette action, est liée la visualisation des concepts membres de ce groupe potentiellement hiérarchisés selon la valeur de la propriété *structuringAssociationType*. Cette visualisation hiérarchique visible en figure 7.4 permet à son tour d'ajouter un comportement d'édition à chaque nœud de la hiérarchie mais aussi de pouvoir modifier la hiérarchie par un « glisser/déposer ».

Les services. Un troisième apport s'intéresse aux services tels que l'interrogation des connaissances en base ou encore l'import et l'export de SOC dans un langage différent. Ce dernier service est réalisé grâce à notre méthode de trans-

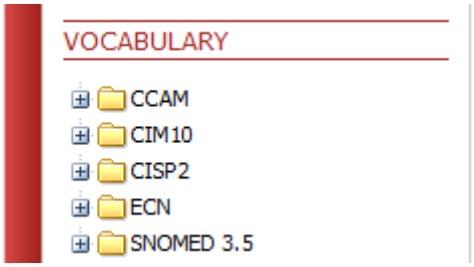

FIGURE 7.3 – ITM – Capture écran de la visualisation des groupes pour le projet InterSTIS (*cf.* section 8.1). « CCAM » est un exemple de groupe représentant une terminologie avec une définition en intention (*cf.* section 6.3.3). Tandis que « ECN » est un exemple de groupe représentant un ValueSet de concepts inter-terminologiques et est défini en extension (*cf.* section 6.3.3).

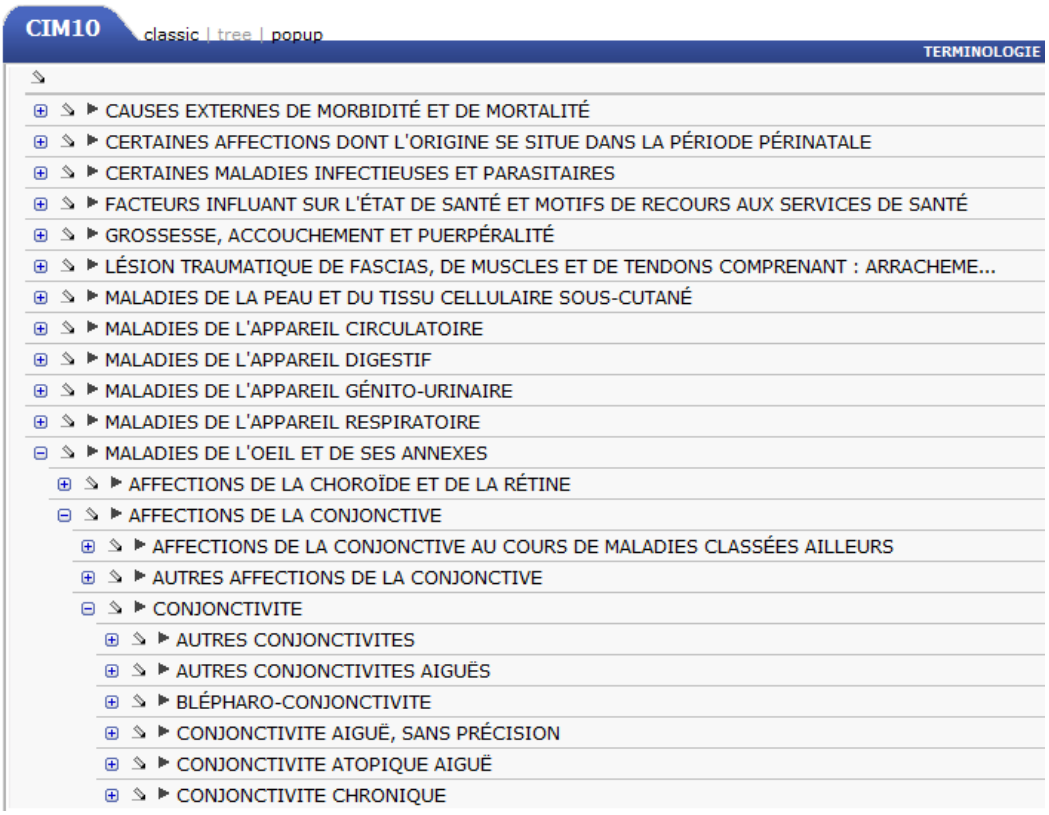

FIGURE 7.4 – ITM – Capture écran de la visualisation hiérarchique des concepts membres de la terminologie CIM-10. Cet exemple est issu du projet InterSTIS (*cf.* section 8.1).

formation de modèles (*cf.* section 7.1). Pour le développement de services d'interrogation des SOC et de leurs contenus, nous essayons le plus possible, de reprendre les spécifications de la norme CTS 2. Les services que nous avons mis

en place appartiennent au paquetage « Search / Access » (*cf.* section 4.4.2).

Par exemple, le service « Resolve Available Code Systems » a pour fonction de retourner la liste des SOC que contient un serveur multi-terminologique. Nous avons ensuite implémenté chaque service sous forme d'une requête SPARQL (pour interroger le modèle en OWL) et intégré dans un Web service (pour interroger la base de données). En reprenant l'exemple précédent, la requête SPARQL associée est la suivante :

```
SELECT DISTINCT ?codeSystems
WHERE {
    ?codeSystems rdf:type <http://www.mondeca.com/system/group#Group>.
}
```

7.3 Intégration à un entrepôt de données sémantiques

7.3.1 Présentation de la solution

Nous avons pris la décision d'utiliser le langage OWL pour représenter notre modèle et la connaissance des SOC (*cf.* section 6.5). Ce choix ouvre la possibilité d'utiliser des outils du Web sémantique tels que des entrepôts de données sémantiques. Ces entrepôts, aussi appelés « triplestore »(en français : entrepôt de triplets), stockent la connaissance directement sous forme de graphe RDF. Une étude approfondie des principaux triplestores a été réalisée par C. Bizer *et al.* [Bizer 2009b].

Cette intégration a pour but de démontrer la faisabilité d'intégration de notre modèle dans un environnement indépendant des outils proposées par MONDECA. Nous avons pour cela utilisé l'entrepôt de données sémantique Sesame [Schenk 2008] qui autorise des connections par le protocole et langage SPARQL. Pour démontrer la pertinence de notre modèle UniMoKR, nous avons développé une application Web appelée « Terminology Browser [7] » (en français : navigateur de terminologies). Cette application tire partie de la représentation unifiée des SOC et accède grâce à des services et interfaces génériques au contenu de n'importe quel SOC stocké dans l'entrepôt de données sémantique. La figure 7.5 illustre l'architecture mise en place.

7.3.2 Services et interfaces de navigation

Cette intégration utilise les SOC CIM-10, SNOMED 3.5, CISP 2 et les alignements entre ces SOC générés par le projet InterSTIS (*cf.* section 6.1) au format

7. L'application Terminology Browser est accessible à l'adresse http://client2.mondeca.com/semanticportalRDF/

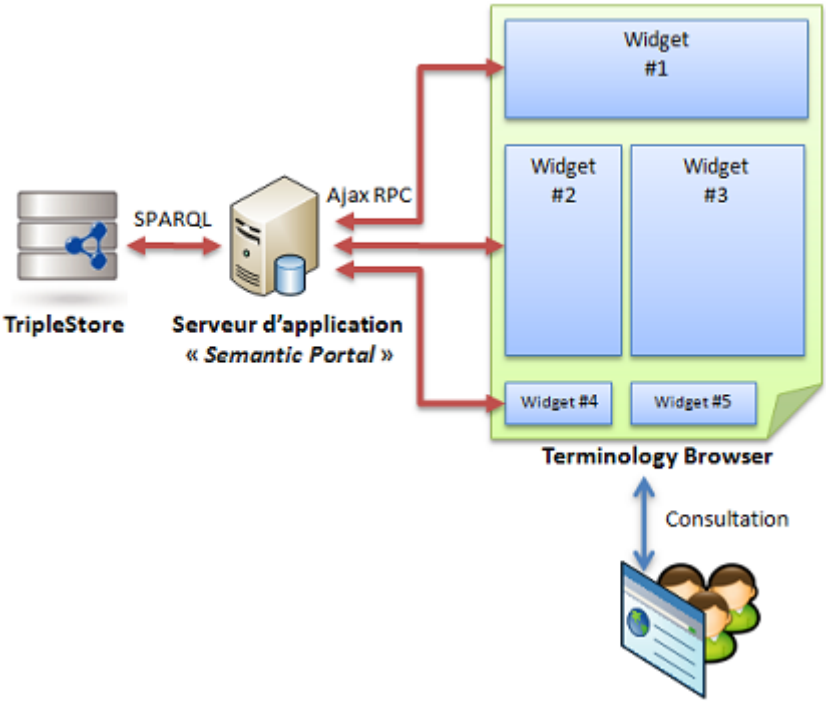

FIGURE 7.5 – Architecture mise en place pour l'intégration de notre modèle à un triplestore et à sa valorisation dans un portail Web.

OWL. Nous avons effectué une transformation des modèle de ces SOC vers le langage SKOS utilisé par notre application. Une fois qu'ils ont été transformés, nous avons importé (i) le modèle UniMoKR, (ii) les extensions de ce modèle pour les SOC et (iii) le contenu des SOC, dans l'entrepôt de données sémantiques Sesame. Cet outil met à disposition un point d'accès SPARQL que nous exploitons pour accéder au contenu de ce serveur. Nos données sont donc stockées directement en SKOS et sont représentées en triplets RDF.

Notre application Web est composée d'une partie serveur responsable de l'interrogation du triplestore et de la communication avec la deuxième partie, la partie client. Cette partie client est constituée d'agents logiciels appelés « widget ». Chaque widget est responsable de l'affichage d'une information précise comme l'arborescence d'une terminologie, le détail d'un concept, etc. Un widget génère et injecte dynamiquement du code HTML brut dans la page Web de l'application. Nous avons défini une librairie de widgets réutilisables et pertinents pour la navigation dans un serveur multi-terminologique. Le contenu et les actions des widgets déclenchent côté serveur des requêtes SPARQL sur l'entrepôt de données sémantique. Ces requêtes sont compatibles avec la norme CTS 2 (*cf.* section 4.4.2).

L'application Web « Terminology Browser » illustrée en figure 7.6 est un exemple

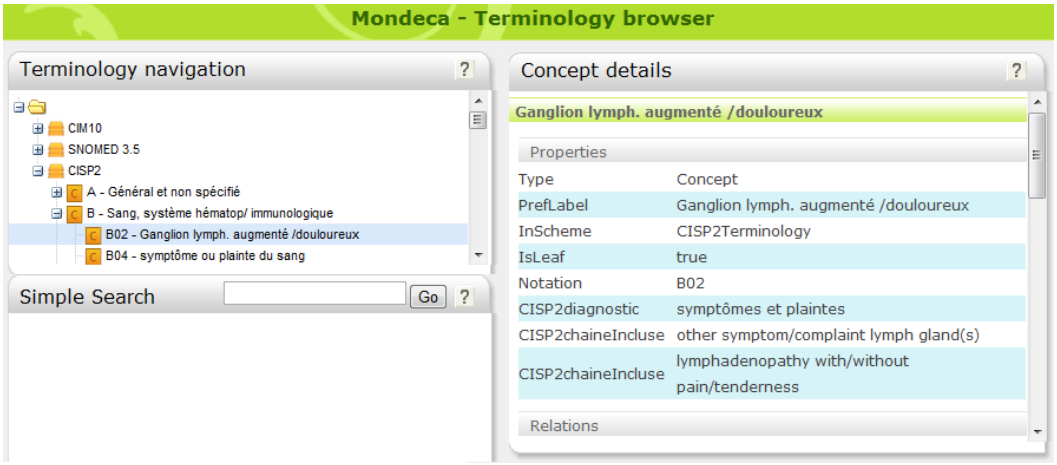

FIGURE 7.6 – Terminology Browser – Capture d'écran générale.

de mise en place de cette architecture. Dans cette application, nous mettons à disposition trois widgets :

- le widget de navigation des groupes. Ce widget affiche les instances de la classe *Group* de notre modèle UniMoKR et permet la navigation des concepts contenus dans chaque groupe en parcourant la relation de navigation spécifiée pour le groupe. Ce widget est illustré en figure 7.7.
- le widget de recherche. Ce widget permet d'effectuer une recherche textuelle et propose des résultats ainsi que leurs groupes d'appartenance. Ce widget est illustré en figure 7.7.
- le widget de présentation d'un concept. Ce widget présente les prédicats et valeurs associés à un concept. Ce widget est illustré en figure 7.8.

L'application côté serveur interprète ensuite chaque action des widgets. Ainsi la demande d'affichage du concept « Conjonctivite aiguë » identifié par l'URI `http://www.chu-rouen.fr/smts#CIM10_H10.3` et appartenant à la CIM-10, génère la requête SPARQL suivante :

```
SELECT DISTINCT ?predicate ?predicateLabel ?object ?objectLabel ?ConceptScheme ?ConceptSchemeLabel
WHERE {
    <http://www.chu-rouen.fr/smts#CIM10_H10.3> ?predicate ?object.
    OPTIONAL {
        ?predicate rdfs:label ?predicateLabel.
        FILTER (lang(?predicateLabel) = "fr").}
    OPTIONAL {
        ?object skos:prefLabel ?objectLabel.
        FILTER (lang(?objectLabel) = "fr").
        FILTER (isURI(?object)).}
    OPTIONAL {
```

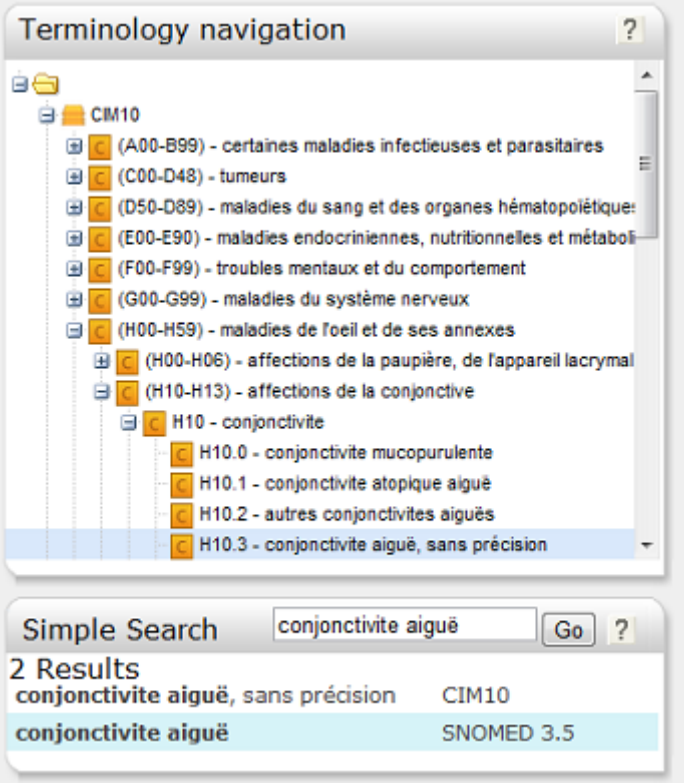

FIGURE 7.7 – Terminology Browser – Capture d'écran du widget de navigation hiérarchique et de recherche textuelle.

```
        ?object skos:inScheme ?ConceptScheme.
        ?ConceptScheme skos:prefLabel ?ConceptSchemeLabel.
        FILTER (lang(?ConceptSchemeLabel) = "fr").
        FILTER (isURI(?object)).}
}
```

Cette requête interroge tous les prédicats et objets associés à la ressource sujet « Conjonctivite aiguë ». La requête interroge également les libellés des prédicats et objets ainsi que les groupes auxquels le concept sujet appartient. Les paramètres d'identifiant du sujet, et de langue des libellés sont dynamiques et dépendent des paramètres et des actions de l'utilisateur dans le widget de présentation de concept.

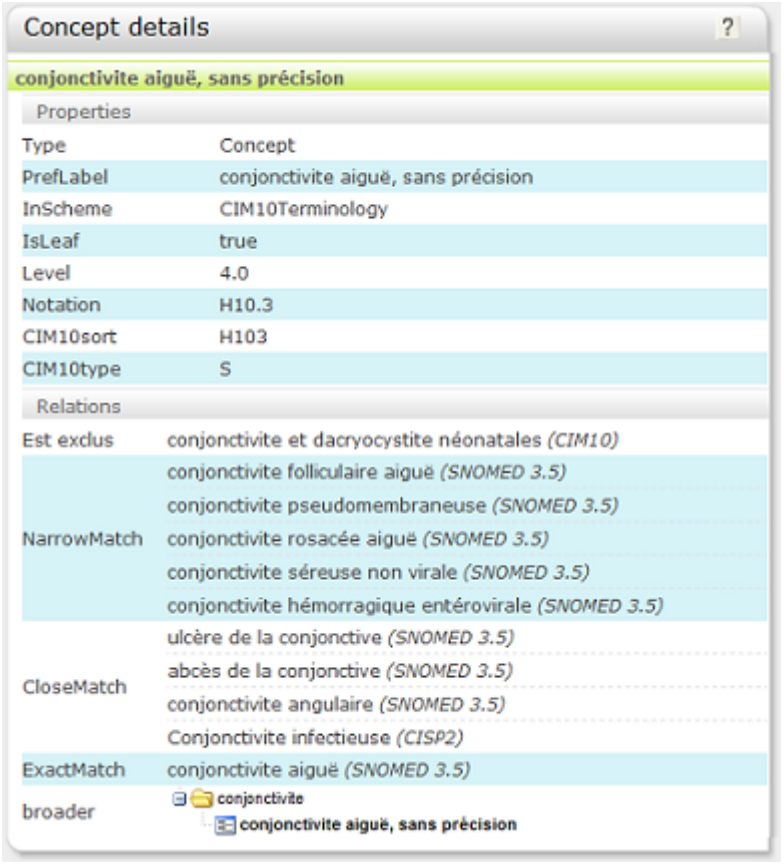

FIGURE 7.8 – Terminology Browser – Capture d'écran du widget de présentation d'un concept.

Chapitre 8

Mise en œuvre de nos travaux

Sommaire

Grâce à la convention CIFRE qui nous lie à la société MONDECA, notre travail de thèse a bénéficié de nombreuses mises en application tant dans des projets de recherche que dans des projets du secteur privé. Nous commençons par décrire l'élaboration et la mise en place de notre solution au sein du projet de recherche InterSTIS au cœur des problématiques de notre travail de thèse. Nous détaillons ensuite deux projets que sont la mise en place de notre solution pour les analyses biomédicales et pour l'édition du thésaurus Eurovoc. Ces deux projets confrontent directement notre modélisation avec des besoins d'utilisation en routine. Nous détaillerons plus finement le projet AnaBio dans lequel nous avons effectué l'ensemble des activités de modélisation, de reprise de données et de validation. Enfin nous présentons un projet de recherche pour la Lecture Rapide en Urgence du Dossier Informatisé du patient (LERUDI) dans lequel notre modèle a été utilisé pour l'élaboration d'une Ressource Termino-Ontologique.

8.1 Applications du projet InterSTIS

C'est dans le contexte du projet de recherche InterSTIS que nous avons élaboré une grande partie de notre modèle [1] [Joubert 2011]. Ce projet, présenté en section 6.1, a permis l'évaluation de nos travaux de recherche. En effet, trois démonstrateurs ont fait la preuve que la plateforme que nous avons mise en œuvre est opérationnelle.

8.1.1 Intégration de la terminologie TUV

Ce travail était l'occasion pour la société VIDAL de regrouper en un seul les différents thésaurus qu'elle exploite pour le traitement des informations relatives aux médicaments. Dans le cadre du projet, ce travail permet de démontrer que le modèle UniMoKR est capable d'intégrer une nouvelle terminologie qui est différente de celles initialement présentes. La Terminologie Unifiée Vidal (TUV) est le résultat de la fusion de quatre thésaurus internes à la société VIDAL servant à l'indexation des Résumés des Caractéristiques du Produit (RCP).

Le résultat, dont l'extension de modélisation est illustrée en figure 8.1, a été concluant. Les termes d'indexation initiaux *TUVTerme* ont été découpés en un ou plusieurs concepts élémentaires *TUVConcept*. Chaque concept possède un type *TUVType* (propriété intrinsèque du concept : pathologie, physiologie, traitement,

1. À l'exception du découpage termino-conceptuel.

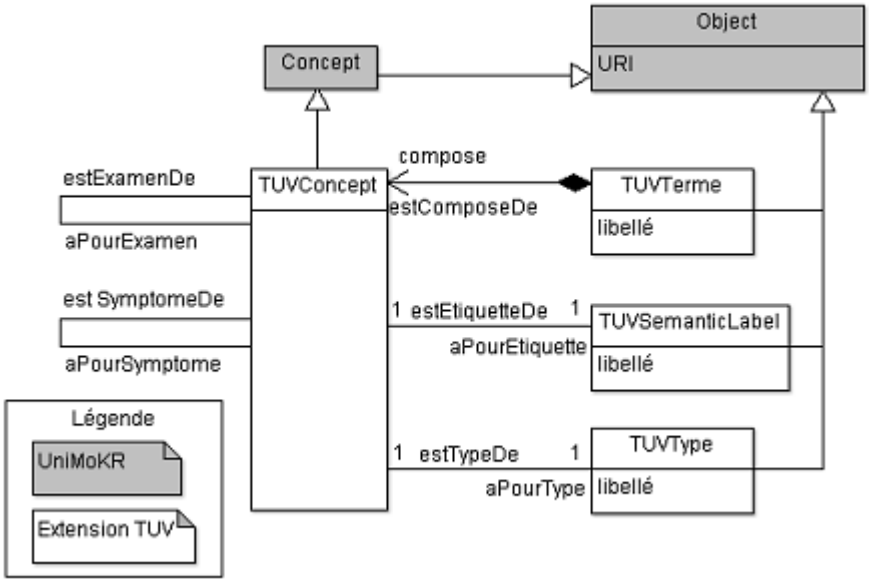

FIGURE 8.1 – Extension du modèle UniMoKR pour la terminologie TUV.

etc.) et un libellé sémantique (e.g. type de diagnostique, type de patient, etc.). Ce découpage a été validé par un expert. A la suite de cette validation, une mise en relation des concepts découpés a été réalisée. Le but de cette étape a consisté à identifier les relations sémantiques intervenant entre les concepts du TUV. Les trois relations sémantiques traitées dans le cadre du projet InterSTIS sont : *BroaderNarrowerRelation* (issue du modèle UniMoKR), *estExamenDe* et *EstSymptomeDe*. Ce modèle ainsi que des correspondances avec les SOC du projet d'InterSTIS ont été importés dans le serveur multi-terminologique.

8.1.2 Indexation de l'ECN et recherche de documents pédagogiques

Une seconde application des services offerts par le serveur multiterminologique d'InterSTIS a été réalisée par le laboratoire LabSTIC pour aider à l'indexation des items de l'Examen Classant National (ECN) en France [2]. Avec la promotion et le développement des contenus numériques pour l'apprentissage en médecine (comme dans d'autres disciplines universitaires), l'indexation des supports est vite devenue un enjeu méthodologique et technique majeur. L'indexation suppose deux actions successives : la première consiste à choisir un ensemble de descripteurs (concepts) ; la seconde consiste à attacher ces descripteurs à la ressource originale au moyen

2. Cette application est accessible à l'URL : http://bit.ly/nYeVZ8

d'une structure de métadonnées.

Avec le temps, le référentiel ECN est devenu un support « métier » structurant une grande partie des connaissances du cursus des études médicales. Il conditionne en effet (i) le travail des étudiants dans leur quête de ressources complémentaires aux supports traditionnels diffusés lors des enseignements ; (ii) le travail des enseignants dans leur façon de construire les documents d'apprentissage ; (iii) le référencement des recommandations de pratiques publiées par la Haute Autorité de Santé ; (iv) le référencement d'articles didactiques publiés dans certaines revues médicales ; (v) l'organisation de l'accès aux ressources dans les bibliothèques universitaires. Un exemple d'item qui compose l'ECN est « Tumeur du colon et du rectum ». Cet item recouvre un ensemble de connaissances qu'un étudiant en médecine doit maitriser.

Deux aspects majeurs de l'ECN ont été retenus dans le cadre du projet : la constitution puis l'import dans le serveur de la classification des items du programme de préparation aux Epreuves Classantes Nationales et la validation du modèle conceptuel et de la structure unifiée implémentée dans le serveur de SOC. Sur la base de la liste des objectifs de l'ECN, l'équipe LabSTIC a choisi pour chaque item ECN, les concepts qui le décrivent dans le MeSH, la CCAM et la CIM-10. L'équipe du LabSTIC a utilisé pour cela la littérature actuellement diffusée auprès des étudiants [Sroussi 2008, Fayssoil 2010] ainsi que deux portails terminologiques : le Portail Teminologique de Santé [3] et le portail terminologique multidisciplinaire TermSciences [Khayari 2006].

Un item ECN est donc, au sens du métamodèle UniMoKR, un groupe de concepts. Un item peut être contenu dans un module ou dans une partie. Nous modélisons donc un *Item*, un *Module* et une *Partie* comme des sous-classes de *ConceptGroup* comme l'illustre la figure 8.2. Les concepts décrivant chaque item n'ont pas besoin d'être re-déclarés au sein du serveur puisque les SOC auxquels ils appartiennent sont déjà présents. On ajoute juste un lien *inGroup* entre les concepts et les items. Les instances d' *Item*, de *Module* et de *Partie* sont reliées par la relation *SuperSubGroup*.

L'intégration de l'ECN dans le serveur multi-terminologique permet de disposer d'outils d'alignement et des SOC pour décrire chaque élément de cette classification. Ensuite, il est possible d'exporter l'ensemble de l'ECN et des concepts indexés. La figure 8.3 montre l'exemple de l'intégration de l'item 148 de l'ECN à l'outil de recherche pédagogique.

Cette démarche a permis de valider le modèle générique des objets du serveur

3. Le Portail Terminologique de Santé est développé par l'équipe du CiSMeF. http://pts.chu-rouen.fr/

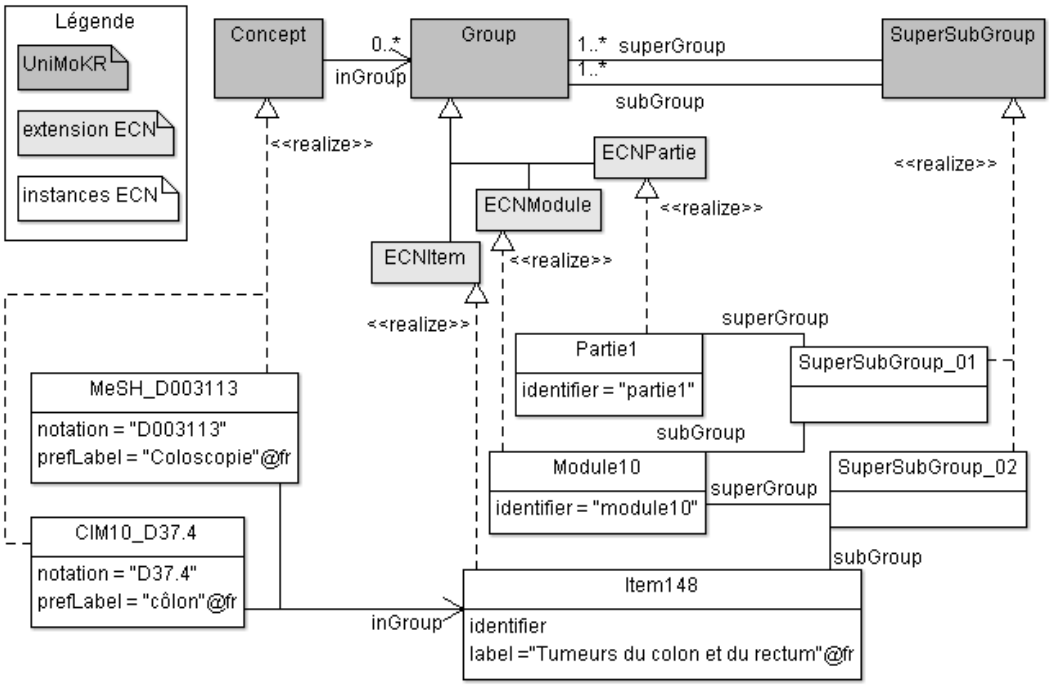

FIGURE 8.2 – Extension du modèle UniMoKR pour la représentation de l'ECN. Une partie de l'instanciation de ce modèle étendu est également illustrée.

d'InterSTIS et l'imbrication des groupes de concepts.

8.1.3 Indexation et affinement de recherche

Une troisième application des services offerts par le serveur multiterminologique d'InterSTIS a permis de démontrer son utilisation pour enrichir les possibilités d'indexation et de recherche ultérieure d'information dans le portail du projet WRAPIN. Les organismes HON et LERTIM ont participé au projet Européen WRAPIN [Gaudinat 2004] qui a abouti à la réalisation par HON d'un moteur de recherche [4] de sites Web de santé accrédités HON. Ce moteur de recherche fonctionne grâce à l'interprétation des requêtes des utilisateurs vers le catalogue de sites de santé accrédités. Ce moteur, illustré en figure 8.4 offre également la possibilité à l'utilisateur de reformuler ou préciser sa recherche grâce à des libellés de concepts ou des qualificatifs à joindre à la question initiale. Les concepts dont les libellés sont proposés pour reformuler la requête sont établis dynamiquement en fonction de la question.

Initialement, cette possibilité reposait uniquement sur le thésaurus MeSH. Pour cette dernière fonctionnalité, le moteur extrait des documents trouvés, un ensemble

4. Le moteur de recherche est accessible à l'URL suivante : http://bit.ly/o3hDgJ

FIGURE 8.3 – Ensemble des concepts qui décrivent l'item 148 de l'ECN. Cet exemple provient de l'outil développé par le LabSTIC accessible à l'URL : `http://bit.ly/nYeVZ8`

de mots pertinents faisant ou non partie de MeSH. Dans le projet InterSTIS, le but était d'étendre la fonctionnalité citée ci-dessus en proposant des libellés de concepts médicaux, en français uniquement, venant d'autres SOC en se fondant sur leurs correspondances avec MeSH. Les correspondances de 10 267 concepts MeSH avec CIM-10, SNOMED 3.5 et CISP 2 ont été utilisées dans le moteur de recherche WRAPIN pour améliorer la suggestion de termes. Concrètement, une liste de concepts MeSH est extraite depuis les résultats d'une recherche par WRAPIN et la correspondance dans les trois autres terminologies est proposée en vue d'affiner la recherche de l'utilisateur. L'utilisateur a la possibilité d'accéder à cette fonctionnalité dans la recherche avancée et visualise la hiérarchie des terminologies restreintes aux concepts

FIGURE 8.4 – Moteur de recherche Wrapin. Résultats de sites Web certifiés HON de la recherche « conjonctivite » en langue française.

pertinents. Cette fonctionnalité est illustrée en figure 8.5.

8.1.4 Discussion

Le projet InterSTIS nous a fourni le cadre pour élaborer notre modèle. Les trois applications réalisées ont permis de démontrer (i) l'utilisation de notre modèle UniMoKR et ses possibilités d'extension pour représenter des SOC différents, (ii) la capacité de notre modèle à représenter les correspondances, (iii) la possibilité d'exporter tout ou partie de SOC et (iv) l'utilité de disposer d'un serveur multi-terminologique. Lors de ce projet, nous avions dans un premier temps mis à disposition des services d'accès aux connaissances compatible CTS2 pour supporter les applications. Pour ces applications qui nécessitent un accès temps-réel aux connaissances du serveur, cette stratégie n'était pas adaptée. Nous avons ainsi mis en place des exports à fréquence régulière spécifiques à chaque application comprenant un ensemble de SOC. Cette dernière stratégie offre des performances meilleures

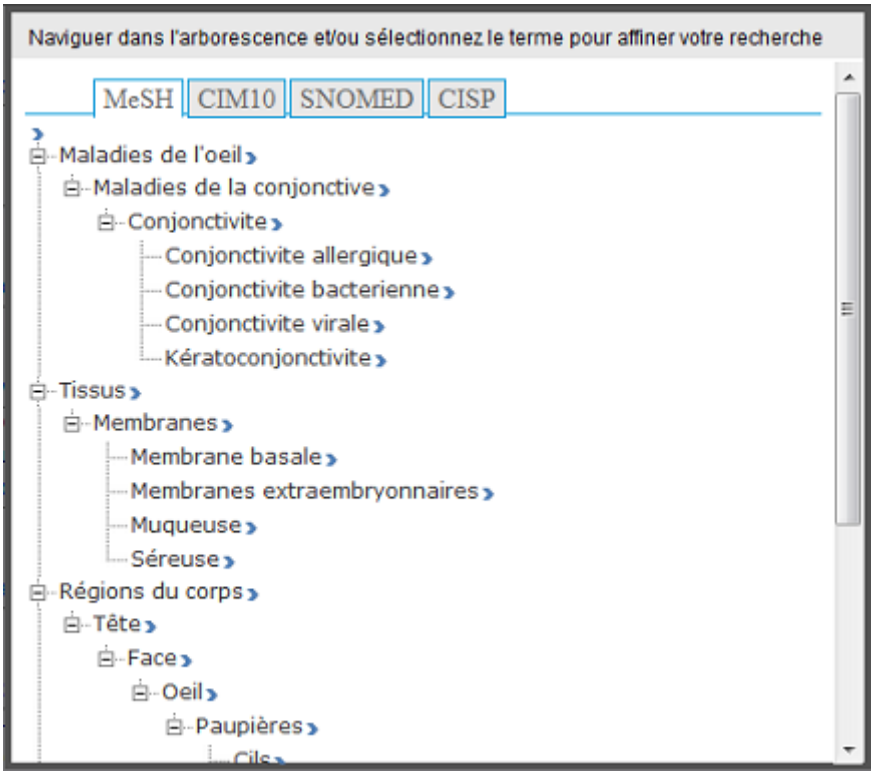

FIGURE 8.5 – Recherche avancée du moteur de recherche Wrapin pour le mot « conjonctivite ». Le résultat propose par SOC, les branches de la hiérarchie qui contiennent un concept répondant à cette recherche.

lors d'une architecture où le serveur multi-terminologique est distant.

8.2 AnaBio : un dictionnaire des Analyses Biomédicales

8.2.1 Présentation

8.2.1.1 Contexte

l'Assistance Publique-Hôpitaux de Paris (AP-HP) est le centre hospitalo-universitaire le plus grand d'Europe avec ses 12 groupes hospitaliers composés de 44 hôpitaux (23 000 lits, 1 000 000 patients hospitalisés et 4 000 000 patients externes par an). En 2005, l'AP-HP a décidé de développer un nouveau Système de Gestion de Laboratoire (SGL) commun aux 12 groupes hospitaliers et basé sur une application commune. C'est dans ce contexte qu'a été construit le dictionnaire d'analyses biologiques (AnaBio) commun à toute la chaine de traitement : prescription, analyse dans les SGL et transmission du résultat.

Utilisable par tous les SGL opérant dans 21 hôpitaux de l'établissement répartis

en 165 laboratoires, le référentiel sur lequel s'appuie ce dictionnaire doit idéalement rester indépendant des contraintes de ces outils. Toutefois, au-delà du couple [code, libellé], disponible en français dans le référentiel, il apparaît nécessaire de standardiser d'autres éléments, tels que le libellé d'affichage, le libellé d'édition, les unités et les codes mnémoniques de chaque analyse.

Afin d'assurer la communication des résultats de ces analyses aux professionnels de santé, quels que soient leur lieu et mode d'exercice, le dictionnaire doit s'appuyer sur une nomenclature partagée par le plus grand nombre. LOINC (Logical Observation Identifier Names and Codes) [McDonald 2003] possède aujourd'hui le plus fort développement international.

L'AP-HP choisit d'interfacer son dictionnaire d'analyses avec LOINC et d'en assurer la maintenance (Le choix de n'avoir pas utilisé directement LOINC sera discuté en section 8.2.2.1). C'est également le choix pris par d'autres hôpitaux montrant un interfaçage plus ou moins complet avec LOINC [Lin 2009]. La traduction en français des libellés de LOINC fait l'objet d'un travail coopératif entre la Société Française d'Informatique de Laboratoire (SFIL) et l'AP-HP.

8.2.1.2 Objectifs

L'objectif de ce projet est de mettre en place une plateforme de maintenance et de publication du dictionnaire des analyses biomédicales de l'AP-HP qui puisse satisfaire aussi bien les exigences liées au référentiel que celles liées au processus de maintenance et de diffusion. Le logiciel actuel de gestion du référentiel (tableur Excel®) montre des limites d'adaptation face à ces exigences et aux perspectives du dictionnaire. Pour atteindre cet objectif, nous utilisons le métamodèle UniMoKR (*cf.* section 6). Ce projet doit permettre :

- la représentation multi-terminologique et des correspondances ;
- la représentation de connaissances annexes (*e.g.* les contacts de chaque laboratoire) ;
- la mise à jour de terminologies (*e.g.* la mise à jour tous les six mois de LOINC) ;
- le stockage de la traduction de LOINC.

8.2.2 Mise en œuvre

8.2.2.1 Le dictionnaire des analyses biomédicales (AnaBio) et LOINC

Le dictionnaire des analyses biomédicales mis en place depuis 2006 à l'AP-HP contient plus de 39000 références utilisées par 165 laboratoires. L'élaboration du dictionnaire bénéficie de la liste des analyses réalisées par les laboratoires de l'AP-HP

engagés dans une démarche de renouvellement de leurs SGL. Chaque analyse est décrite en cinq axes (analytes, paramètres, milieux, techniques et unités), par un libellé (libellé AP) auquel est affecté un code alphanumérique à 5 caractères (index AP), et, le cas échéant un code LOINC. Chaque axe est structuré en 3 niveaux hiérarchiques. Des éléments nécessaires au paramétrage des SGL sont ajoutés : libellés d'affichage, libellés d'édition, codes d'appel des analyses (mnémoniques). Les composants en français du référentiel AnaBio s'appuient en partie sur les termes de la nomenclature Names-Lab [Cormont 2002]. L'ensemble des analyses du dictionnaire envoyées dans le serveur de résultats est ordonné. Il en découle une présentation électronique unique destinée aux cliniciens pour une consultation univoque des comptes-rendus de résultats. Le dictionnaire AnaBio est également lié à des données annexes comme la liste des structures hospitalières utilisatrices ainsi que leurs contacts.

Nous disposons des versions successives de LOINC depuis décembre 2005 (v2.16 à v2.34), librement téléchargées depuis le site Web[5]. La nomenclature LOINC comporte quatre parties, mais seule la classification des termes de laboratoire (Laboratory Term Classes) est utilisée dans ce travail. De 31 437 libellés d'analyses en 2005, elle passe à plus de 44 000 libellés en 2010, répartis dans 11 chapitres qui couvrent les six disciplines (biochimie, hématologie, immunologie, pharmaco-toxicologie, microbiologie et biologie moléculaire).

Le choix d'un dictionnaire interfacé avec la nomenclature LOINC est motivé par certaines inadéquations quant à l'utilisation du seul référentiel LOINC, parmi lesquels [Cormont 2008] :

- **la non exhaustivité** : LOINC est basé sur un système pré-coordonné de six axes. Il ne couvre pas la totalité des besoins de l'AP-HP et ses combinaisons ne sont pas toutes utiles à ses laboratoires.
- **la granularité** : un niveau de détail supérieur à celui de LOINC est parfois requis par les biologistes de l'AP-HP, pour lesquels certains axes doivent être décrits plus précisément dans des domaines spécialisés de la biologie.
- **les paramètres spécifiques** : Les contraintes de la réglementation française imposent aux SGL l'ajout d'attributs supplémentaires dans le référentiel (par exemple, le code NABM[6]).
- **la langue** : les données doivent être disponibles en français.

Le dictionnaire AnaBio offre la souplesse de gestion nécessaire à son utilisation quotidienne tout en conservant l'interopérabilité sémantique avec les autres organismes internationaux de santé grâce à son alignement avec LOINC. Le dictionnaire des

5. Voir http://www.loinc.org
6. Nomenclature des Actes de Biologie Médicale

analyses biomédicales est à cet égard, un parfait exemple d'une terminologie locale interfacée avec une terminologie de référence [Daniel 2009]. Ce choix implique un travail quotidien d'alignement entre ces deux référentiels pour prétendre à l'interopérabilité.

8.2.2.2 Interaction avec les acteurs de santé

Une cellule centralisée est chargée du recueil des analyses dans les laboratoires de l'institution, de l'enrichissement du dictionnaire, de sa maintenance (création/modification/suppression) et de son interfaçage avec LOINC (*cf.* figure 8.6). Elle assure la formation et l'assistance aux biologistes dans le choix de leurs analyses.

L'ensemble des analyses est soumis à la validation des représentants institutionnels de chaque discipline biologique avec l'objectif d'aboutir à une harmonisation des libellés d'édition et présentation des comptes rendus électroniques ou imprimés. Ce dictionnaire ainsi formé sert de noyau commun à l'ensemble du paramétrage des SGL de l'AP-HP. Toutes les mises à jour du dictionnaire dans les SGL sont bloquées en modification. Seule la cellule centralisée peut importer des mises à jour. Cette cellule chargée du référentiel assure la correspondance et traduit les libellés LOINC chaque fois que possible. Une collaboration étroite est établie avec le comité du Regenstrief Institute, auquel sont régulièrement soumises des analyses manquantes. Après validation par le comité, un code LOINC est retourné et la mise à jour du référentiel est effectuée. A ce jour, cette collaboration est à l'origine de la création de 2000 nouveaux codes LOINC.

8.2.2.3 Étapes du projet

Modélisation. Le travail de modélisation est réalisé en collaboration étroite avec l'unité de maintenance de la terminologie afin de comprendre l'utilité de chaque élément mais également d'appréhender les nouveaux besoins ayant un impact sur le modèle à produire (par exemple des propriétés de statut à ajouter pour permettre une meilleure traçabilité des éléments dans le temps). La structuration du tableur existant en onglets et en colonnes constitue une première étape d'organisation, de compréhension du domaine, et de prise en compte des besoins d'échanges de données. La figure 8.7 illustre l'extension du modèle UniMoKR pour le projet AnaBio avec la prise en compte de connaissances annexes au dictionnaire. Le dictionnaire AnaBio est constitué d'*Analyses* divisées en *Axes*. Chaque Analyse appartient à une *Discipline*. Les informations de *StructureHospitaliere*, d'*Hopital* et de*Contact* constituent des connaissances

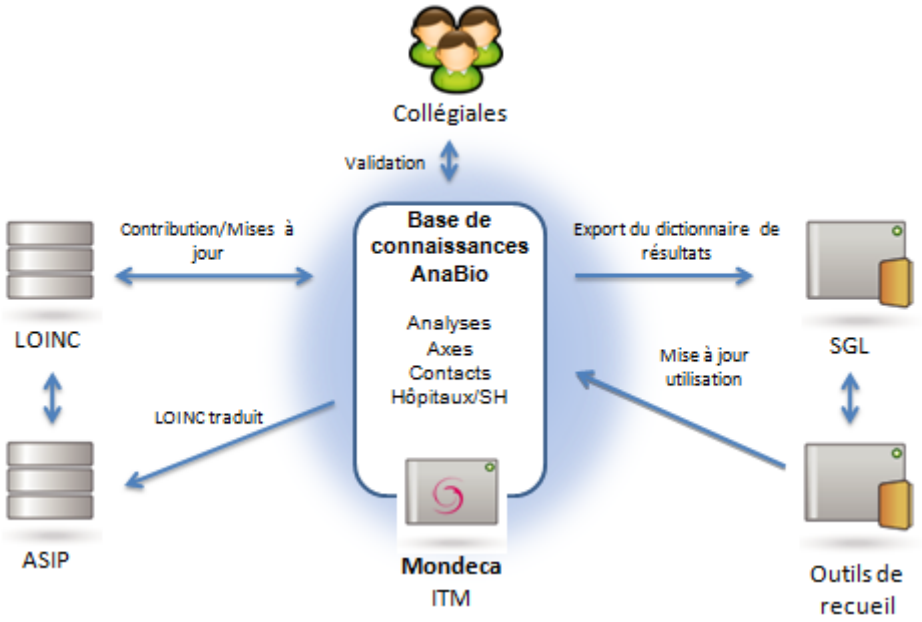

FIGURE 8.6 – Flux de données autour de la base de connaissances de biologie.

annexes au dictionnaire. Ces éléments ne sont pas représentés dans le modèle UniMoKR mais peuvent être liés à ce modèle ou à une extension.

Reprise de données. Parallèlement à la construction du modèle, débute la tâche de reprise de données. Ceci nécessite la transformation de l'intégralité des données du tableur afin de permettre leur intégration et leur mise en conformité avec le nouveau modèle formel. D'autres données sur les connaissances annexes sont fournies au format tableur. Au cours de cette reprise, certaines incohérences dans les données sont identifiées et corrigées (contrôles de doublons, erreurs orthographiques, contrôle d'intégrité sur certaines valeurs, etc.). La reprise de données s'appuie sur un outil JAVA développé spécifiquement. A partir d'une exploration du fichier XLS, il génère des fichiers au format RDF/XML conformes au schéma décrit dans le modèle et prêts à être intégrés. Cette étape permet d'une part de valider le modèle en le confrontant à des données et d'autre part d'améliorer la qualité des données contenues dans le dictionnaire tout en préservant les identifiants et le libellé AP-HP (concaténation des axes) qui doivent rester fixes dans l'ensemble du système d'information hospitalier.

Validation. Les phases de modélisation et de reprise de données permettent un travail d'affinement itératif. Pour être validées, les propositions de modélisation sont importées dans la plateforme avec les données reprises. Pendant cette

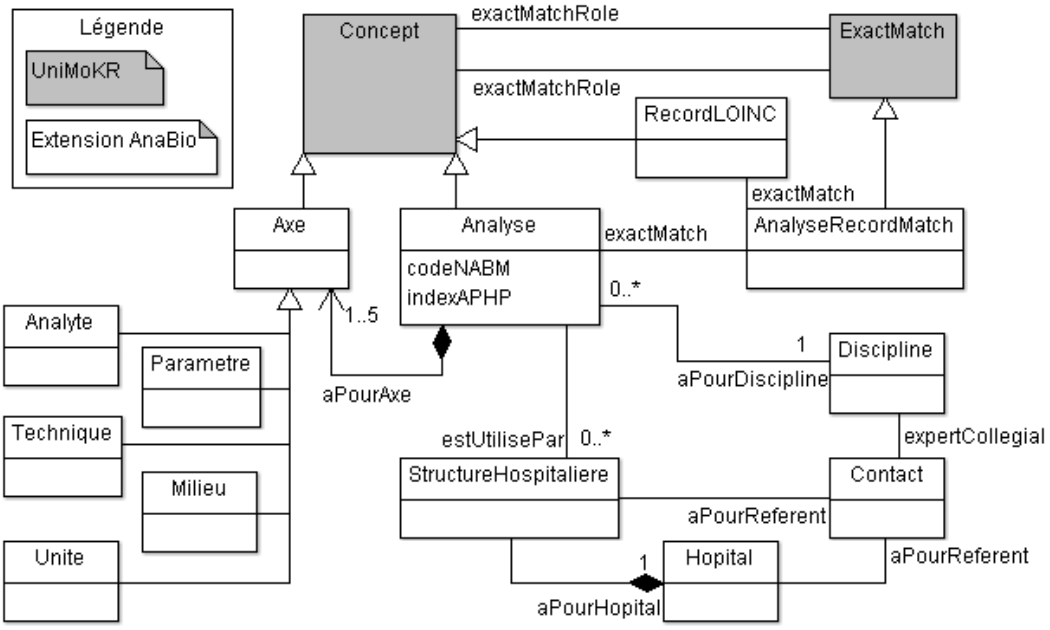

FIGURE 8.7 – Extension du modèle UniMoKR pour le projet AnaBio. Ce modèle simplifié, présente l'extension de modèle pour représenter le dictionnaire AnaBio ainsi que ses informations annexes.

phase, l'équipe de maintenance de la terminologie travaille en parallèle avec le tableur et l'outil ITM. Au terme de cette validation, des corrections et améliorations sont apportées au modèle et donc à la reprise de données. C'est au terme de six cycles de validation qu'intervient le déploiement de la plateforme dans l'environnement de production.

8.2.2.4 Solution intégrée à l'outil ITM

Nous avons intégré notre modèle au logiciel multiutilisateurs et multilingues ITM. Ce logiciel, comme nous l'avons vu en section 7.2, interprète des modèles formels (dont le format OWL) et exploite les restrictions exprimées dans ces modèles pour contraindre les connaissances du domaine représenté. Ce logiciel stocke les données du graphe de connaissances dans une base relationnelle (dans le cadre de ce projet : Oracle®) ce qui garantit la capacité à gérer le volume croissant des référentiels. L'interface de l'outil ITM est contrôlée par le modèle OWL qui a été modélisé. L'outil ITM est également doté d'un module de raisonnement qui permet l'exécution d'inférences ou de règles de contrôle. Il permet dans notre cas de générer un rapport de toutes les analyses ayant une valeur du libellé d'édition dépassant 40

caractères ou encore de mettre en évidence les doublons. Le module de génération de rapports statistiques permet quant à lui de générer un rapport présentant l'effectif correspondant à une liste de requêtes. Ces requêtes se basent sur la définition formelle du graphe de connaissances et permettent par exemple d'exprimer : « l'ensemble des analyses non fermées ayant une relation d'alignement avec un code LOINC » ou encore « l'ensemble des analyses de discipline Biochimie n'étant pas validées par la collégiale mais utilisées par au moins une structure hospitalière ».

8.2.3 Évaluation

8.2.3.1 Réponse de la solution aux exigences

La définition de règles métier s'appuie sur l'expressivité formelle du modèle exprimé en OWL. Ce modèle contient en effet des restrictions de cardinalité, de domaines et co-domaines. Par exemple, une *Analyse* s'adresse à une et une seule *Discipline* ; la relation *estUtiliséPar* a pour domaine une *Analyse* et pour co-domaine zéro ou plusieurs *StructureHosptitalières*. Ces restrictions contraignent les informations éditées par l'outil ITM et permettent à un module de raisonnement de valider l'intégrité des données de la base de connaissances. L'exigence quant à une traçabilité de validation d'une analyse est satisfaite par la définition d'attributs dans le modèle qui permettent de positionner différents états d'un objet. De plus, la modélisation, notamment par ses relations réifiées, permet la représentation de métadonnées sur les relations (créateur, date de création).

Le modèle retenu est générique : appliqué ici aux analyses biomédicales et à LOINC, il peut représenter des référentiels d'autres domaines. Sa caractéristique d'extension rend possible la mise en relation des données annexes avec le dictionnaire. Les demandes d'export et d'import comprennent, par exemple, la mise en place de nouvelles analyses dans une structure hospitalière, la validation par une collégiale des analyses d'une discipline ou encore les demandes de mise à jour de LOINC. Ces actions peuvent être générées automatiquement ou à la demande, dans des formats du Web sémantique (grâce à notre méthode décrite en section 7.1) ou dans des formats variés tels que XML, CSV, XLS, HTML, PDF (grâce aux services de l'application ITM). Les échanges de données avec les différents acteurs au sein des hôpitaux de Paris se font par des fichiers au format CSV intégrés par les responsables des différents systèmes d'application. À plus long terme, cette plateforme va pouvoir communiquer directement avec les systèmes d'information par des messages conformes au standard IHE tels que le décrit le profile LCSD (*cf.* section 4.1.2).

8.2.3.2 Amélioration de la qualité

Un point majeur des résultats de ce projet est l'amélioration de la qualité des données contenues dans le dictionnaire AnaBio. Le passage de données semi-structurées (tableur) à des données structurées (par le modèle formel) a imposé la correction des données considérées comme incohérentes. Il s'agit pour la plupart de différences de casse, d'orthographe ou d'absence de normalisation d'une valeur. Par exemple « cysterceques », « Cysterceque » et « Cysticerques anticorps » sont transformés en « Cysticerques anticorps ». Ces corrections ont pour but l'amélioration de la qualité des données.

Pour atteindre cet objectif, l'équipe responsable du dictionnaire AnaBio a défini l'ensemble normé des valeurs possibles pour chaque axe. Nous avons ensuite développé un programme détectant automatiquement les axes du dictionnaire ne faisant pas partie de cette liste autorisée et suggérant quand cela était possible la correction[7]. Nous avons utilisé cette détection à six reprises entre juillet 2009 et juillet 2010. Cette détection lors de la reprise de données a permis pour une certaine partie, d'effectuer des modifications dans le dictionnaire AnaBio ; les évolutions normales quotidiennes, la mise à jour de certaines disciplines et le nettoyage des données lors d'une détection manuelle ont été à l'origine de l'autre partie des modifications. Les modifications manuelles comprennent la fusion de valeurs d'axe synonymes (tels que « cobalamine » et « vitamine B12 ») ou encore la fusion de termes similaires (tels que « IL-2 » et « interleukine-2 »). Pour tenter de mesurer la part qui peut être imputable au logiciel de détection dans le cadre de la structuration des données en vue de son intégration dans un modèle formel, nous allons analyser l'évolution du référentiel.

Le nombre d'analyses et d'axes a globalement augmenté sur une année comme le montre la figure 8.8 : passant respectivement de 26 458 à 35 714 et de 103 758 à 144 126. Mais cette augmentation n'a pas été linéaire. Ainsi en étudiant le nombre de créations, suppressions et modifications, nous constatons deux phases importantes de nettoyage et une phase majeure d'enrichissement du dictionnaire :

Phase 1 (entre Août et Octobre 2009). De nombreux axes et analyses ont été supprimés accompagnés d'un nombre de créations moitié moindre. 96% des suppressions d'analyses correspondent à la discipline de microbiologie réparties en Virologie (47%), Bactériologie (33%) et Myco-Parasitologie (16%). 84% des créations d'analyses correspondent à la Bactériologie (48%) et à la Myco-Parasitologie (36%). Cette première phase est le début d'un travail effectué en

7. Cette suggestion était limitée aux problèmes de casse, d'ajout ou suppression d'un « s » final.

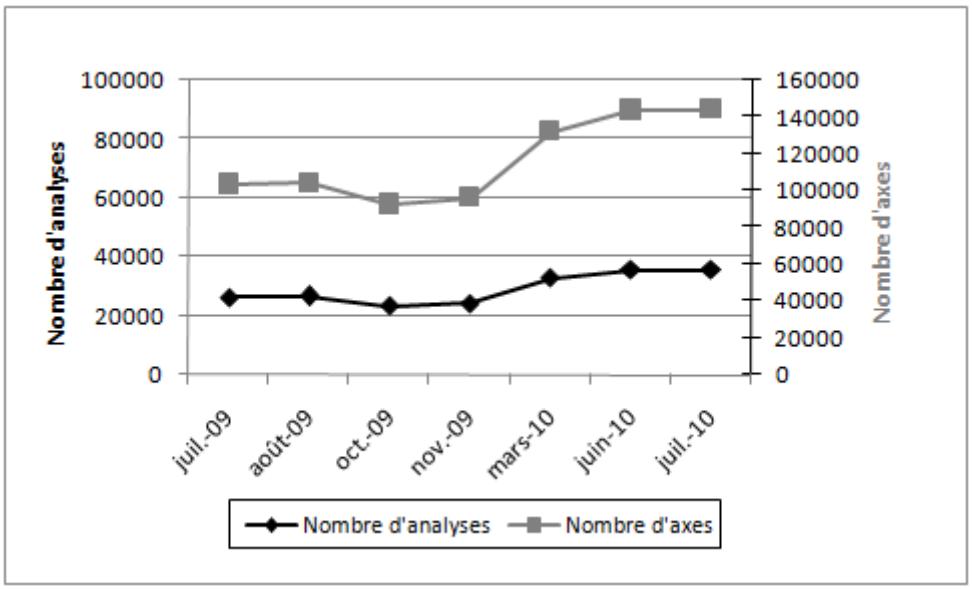

FIGURE 8.8 – Évolution du nombre des analyses /axes pendant la période du projet AnaBio.

parallèle par l'équipe en charge du dictionnaire AnaBio avec d'une part l'enrichissement continu du référentiel et d'autre part le nettoyage et l'amélioration de la qualité des données qui a été prédominante pour cette première phase. Au cours de cette phase, les représentants institutionnels de la microbiologie ont initié un travail de remise à plat des analyses de leur discipline donnant lieu à un grand nombre de suppressions.

Phase 2 (entre Novembre 2009 et Février 2010). Entre novembre 2009 et mars 2010, de nombreuses analyses ont été créées (8 719) correspondant pour 83% à la Virologie. Cette inflation du nombre d'axes et donc d'analyses s'explique d'une part par l'incorporation de nouvelles analyses et d'autre part par la décision de dupliquer les analyses en fonction de la technique.

Phase 3 (entre Mars 2010 et Juillet 2010) Un nombre plus modéré d'analyses ont été crées et supprimées. Si les suppressions correspondent pour 85% aux disciplines de Microbiologie (59%) et Pharmaco-Toxicologie (26%), les créations sont réparties entre plusieurs disciplines. Cette phase tend à consolider le dictionnaire en nettoyant les données en vue de son adéquation avec la modélisation effectuée et son intégration dans l'outil.

Dès Juillet 2009, le programme de détection a eu un impact majeur sur la recherche d'incohérences d'axes comme nous le montre la figure 8.9. À la première itération, nous avons ainsi détecté près de 11% d'axes incohérents. Lors des itérations suivantes, cette détection a représenté un pourcentage inférieur à 1% des actes

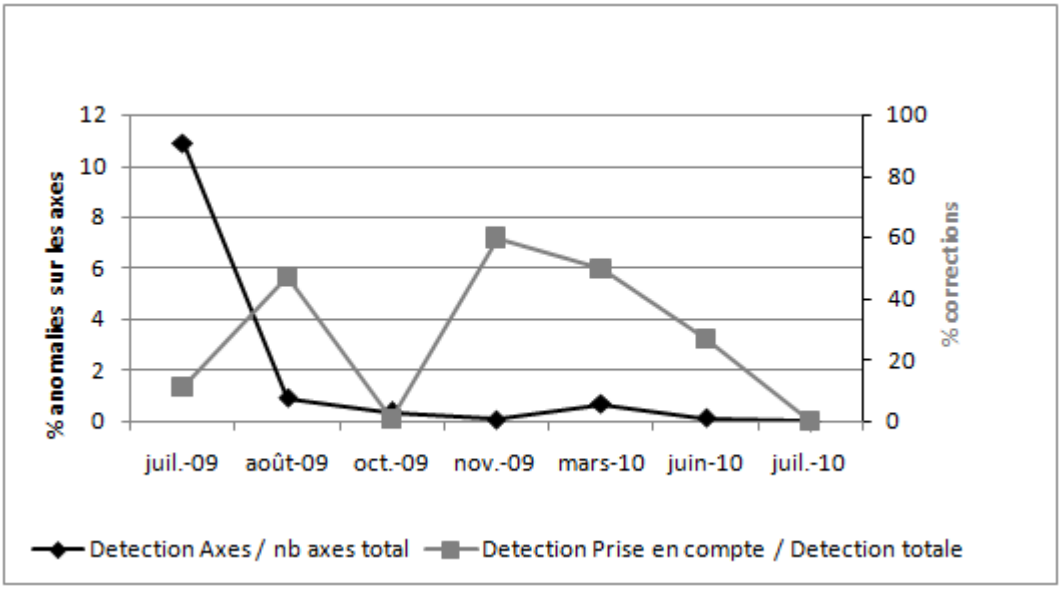

FIGURE 8.9 – Détection et prise en compte des anomalies d'axes sur le nombre total d'axes pendant la période du projet AnaBio.

pour arriver en juillet 2010 à 0%.

Il est intéressant de comprendre pourquoi certains axes détectés comme anormaux n'ont pas été corrigés par l'équipe AP-HP. Nous émettons plusieurs hypothèses quant à leur non prise en compte :

- leur suppression a été reportée à une date ultérieure suite au remplacement des valeurs d'une discipline ;
- la charge de travail était trop importante et leur correction a été repoussée à plus tard ;
- la valeur était correcte et a été ajoutée aux listes de valeurs d'axes autorisées.

On peut constater, en figure 8.10 qu'au début du projet, certaines anomalies détectées n'ont pas été prises en compte immédiatement. Ceci peut s'expliquer par le fait que la totalité des analyses et axes de la microbiologie devait être supprimée suite au travail effectué par les représentants institutionnels de cette discipline. Jusqu'au début de l'année 2010, en raison de l'activité de l'équipe de maintenance et face aux nouvelles introductions de disciplines, certaines modifications ont été reportées. Vers la fin du projet, les corrections étaient mises à jour de façon plus régulière. Le reste des valeurs détectées provenant pour la plupart des nouvelles analyses des disciplines, était correct et ajouté aux listes des valeurs d'axes autorisées.

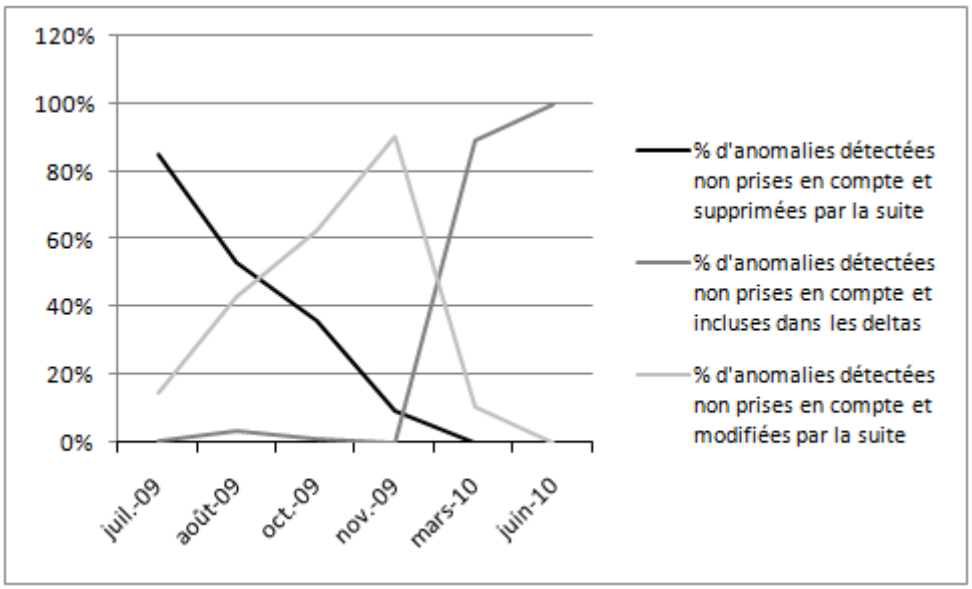

FIGURE 8.10 – Explication par les données de la non prise en compte des anomalies détectées par l'équipe AP-HP.

8.2.4 Discussion

Ce projet a démontré que le modèle UniMoKR est suffisamment générique pour représenter des types différents de terminologies incluant LOINC et AnaBio. Grâce au caractère extensible du modèle, il est possible de prendre en compte les spécificités du dictionnaire AnaBio mais aussi de lier ce dictionnaire aux connaissances annexes comme les structures hospitalières, les contacts, etc.

Le projet AnaBio a été pour nous l'occasion d'étudier l'impact de la modélisation et des outils utilisés pour représenter et maintenir un SOC. Une solution adaptée à la gestion de SOC (comme la nôtre) permet (au contraire de logiciels non adaptés) de contraindre les données saisies et ainsi interdire des données non conformes à la définition formelle du modèle. L'utilisation d'une telle solution est d'autant plus utile que le volume du SOC augmente et accroît le nombre d'incohérences. Au cours du projet AnaBio, 10% des données ont ainsi été corrigées.

Le passage d'un schéma de type tableur à un schéma formel a été l'occasion pour l'équipe de maintenance du dictionnaire d'exprimer explicitement les contraintes sur les connaissances qu'ils manipulent. Cette explicitation facilite la compréhension et la prise en main de ce SOC par des personnes externes à cette équipe ou par de nouveaux collaborateurs.

Un dernier point concerne l'utilisation d'un modèle commun de représentation de SOC. La compréhension du modèle UniMoKR pour l'utilisation du dictionnaire

AnaBio facilite grandement la compréhension d'autres SOC dont la modélisation est fondée sur ce modèle. Seules les extensions spécifiques à chaque SOC demeurent à assimiler.

8.3 Eurovoc

8.3.1 Présentation

8.3.1.1 Contexte

L'Office des Publications de l'Union Européenne est en charge de l'édition des publications des institutions des Communautés européennes et de l'Union Européenne. Il propose en outre des services dont la mise à disposition du thésaurus Eurovoc (*cf.* section 3.4.5). Eurovoc est un thésaurus multilingue, actuellement disponible dans 22 langues officielles, dans une langue d'un pays candidat (croate), et dans la langue d'un pays tiers (serbe). Ce thésaurus a été construit spécifiquement pour le traitement des informations documentaires par les institutions de l'UE. Eurovoc est un thésaurus pluridisciplinaire couvrant des domaines qui sont suffisamment larges pour englober à la fois des points de vue communautaires et des points de vue nationaux, avec une certaine emphase sur les activités parlementaires.

L'Office des publications a décidé en 2008, de remplacer son système de gestion de thesaurus pour la maintenance et la diffusion du thésaurus Eurovoc.

8.3.1.2 Objectifs

L'objectif de ce projet est de mettre en place un système de gestion terminologique qui réponde aux besoins spécifiques d'Eurovoc :

- la gestion du modèle conceptuel Eurovoc multilingue ;
- la prise en compte dans l'édition, des processus de traduction et de validation ;
- la diffusion et la publication du thésaurus sur un site Web public.

Ce thésaurus nécessite la prise en compte d'un processus de traduction et de validation, illustré en figure 8.11, mais également d'une traçabilité complète qui permet par exemple de connaître l'auteur d'une traduction. L'aspect linguistique est très développé dans le thésaurus Eurovoc afin de satisfaire au mieux à sa tâche première de traitement de l'information documentaire.

8.3.2 Mise en œuvre

L'outil ITM a été retenu pour satisfaire ces objectifs. Notre modèle UniMoKR, adapté à la représentation de SOC, a été utilisé pour modéliser le thésaurus Eurovoc.

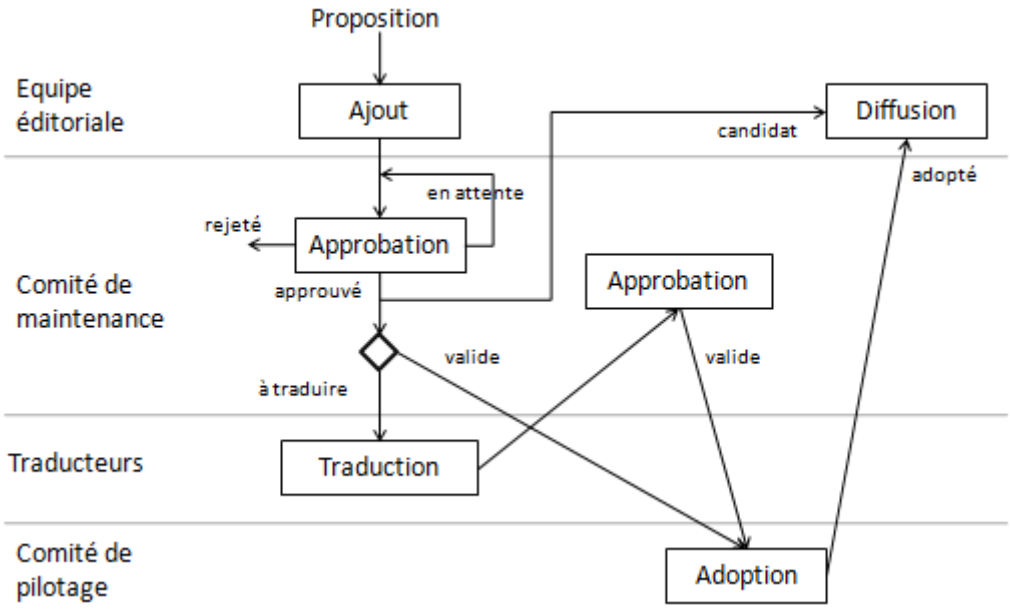

FIGURE 8.11 – Processus de traduction et de validation tel que formulé par l'équipe en charge du thésaurus Eurovoc. L'ajout d'un concept dans le thésaurus passe par les étapes de proposition, d'approbation, de traduction, d'adoption et de diffusion. Pour être diffusé dans la prochaine version du thésaurus, un concept doit être accepté en tant que candidat à la diffusion puis être adopté après approbation de la traduction de ses termes dans les 24 langues du thésaurus.

C'est dans le cadre de ce projet que nous avons affiné la description des termes associés aux concepts. En effet la définition d'attributs sur le concept n'était pas suffisant pour supporter le processus de traduction. Cette séparation des niveaux terminologique et conceptuel a abouti au modèle final UniMoKR présenté en section 6.3.1.

Domaines et microthésaurus. Le thésaurus Eurovoc couvre tous les domaines d'activité des institutions européennes. Il est divisé en 21 domaines de connaissances (en anglais, « domain ») et en 127 sous-catégories appelées microthésaurus (en anglais, « microthesaurus »). Certains domaines sont plus développés que d'autres parce qu'ils sont plus étroitement liés aux centres d'intérêts de l'Union Européenne. Ces domaines et microthésaurus sont autant de sous ensembles du thésaurus Eurovoc. Nous avons donc modélisé les entités *Domain* et *Microthesaurus* comme sous-classes de l'élément *Group* de notre modèle UniMoKR. Cette modélisation est illustrée en figure 8.12. Chaque instance de la classe *Domain* (par exemple « Droit » dont le code est « 12 ») sera liée à l'instance de la classe *Terminology* qui représente le thésaurus Eurovoc par la

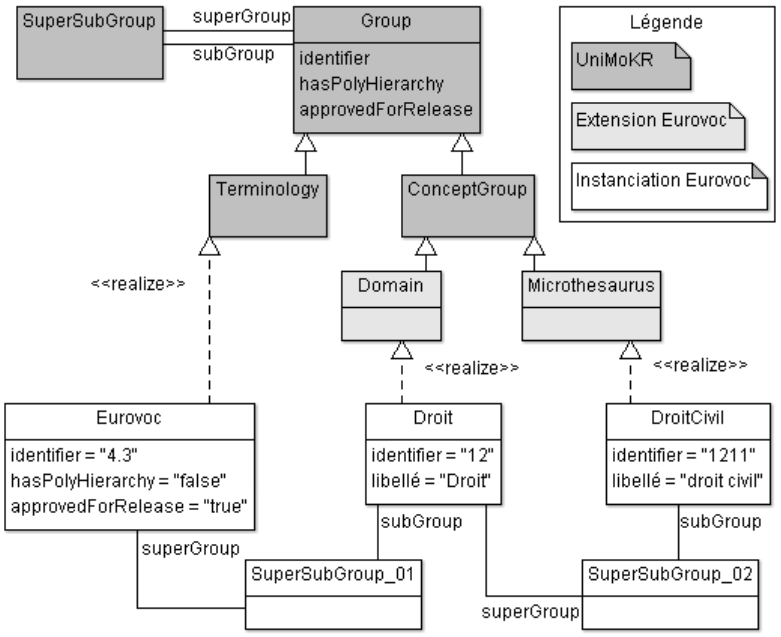

FIGURE 8.12 – Extension du modèle UniMoKR pour Eurovoc. Cette figure illustre également l'instanciation de ce modèle étendu.

relation *SuperSubGroup* où l'instance de *Domain* est sous-groupe de l'instance de *Terminology* (dans notre exemple, « Droit » est un sous-groupe de « Eurovoc »). Il en est de même entre les instances de la classe *Microthesaurus* et celles de la classe *Domain* (par exemple le microthésaurus « Droit civil » est sous-groupe du domaine « Droit »). Cette modélisation hiérarchique entre le thésaurus Eurovoc et ses sous-ensembles est valorisée dans les interfaces utilisateurs par une visualisation arborescente qui permet d'accéder facilement aux connaissances d'un domaine particulier. Ce point est illustré en figure 8.13.

Concepts et Termes. Le thésaurus Eurovoc présente une description fine des concepts et termes. Cette description, conforme aux recommandations de la nouvelle norme ISO 25964 pour l'élaboration de thésaurus (*cf.* section 4.3.3), est similaire à la modélisation des relations entre concepts et termes que nous avons définie dans notre modèle UniMoKR. L'idée directrice est que la hiérarchie des éléments d'Eurovoc soit identique quel que soit la langue : c'est-à-dire une hiérarchie de concepts indépendants d'empreintes linguistiques. A chaque concept sont associés un terme préféré et des termes non préférés (synonymes) pour le désigner dans une langue. Cette représentation termino-conceptuelle fine est présentée en section 6.3.1.

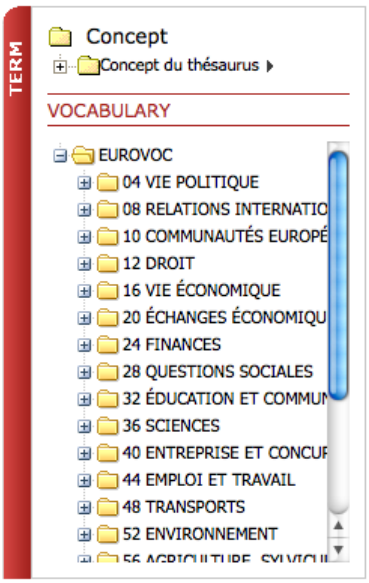

FIGURE 8.13 – Capture d'écran ITM : visualisation de la hiérarchie des groupes de concepts.

Pour satisfaire au besoin de représentation terminologique du projet Eurovoc, nous avons étendu le modèle UniMoKR d'une relation de traduction entre termes. La figure 8.14 illustre le résultat de la représentation des données d'Eurovoc conforme à notre modèle intégré au logiciel ITM. On y voit le concept désigné par le terme préféré en Français « cryptographie ». Ce concept est déclaré explicitement comme membre du groupe « 3236 informatique et traitement des données » (microthésaurus). Le concept porte le code (*notation*) « 6778 » et possède une note éditoriale (*editorial note*). Enfin ce concept possède un terme préféré pour chaque langue et des termes non préférés dans certaines langues dont deux en Français que sont « chiffrement » et « cryptage ».

Traduction et validation. Le processus de traduction et de validation est très important pour la maintenance d'un thésaurus où des équipes internationales interviennent. La majorité de ce processus est supportée par l'outil ITM. Toutefois, nous avons ajouté des attributs pour capturer les changements d'état de validation des éléments. La figure 8.15 illustre le terme préféré « cryptographie ». Ce terme possède l'attribut de diffusion avec la version (*released with version*) 4 et supérieure du thésaurus Eurovoc.

Pour supporter le processus de demande/proposition/validation associé aux traductions, nous avons ajouté des attributs de validation sur les termes et

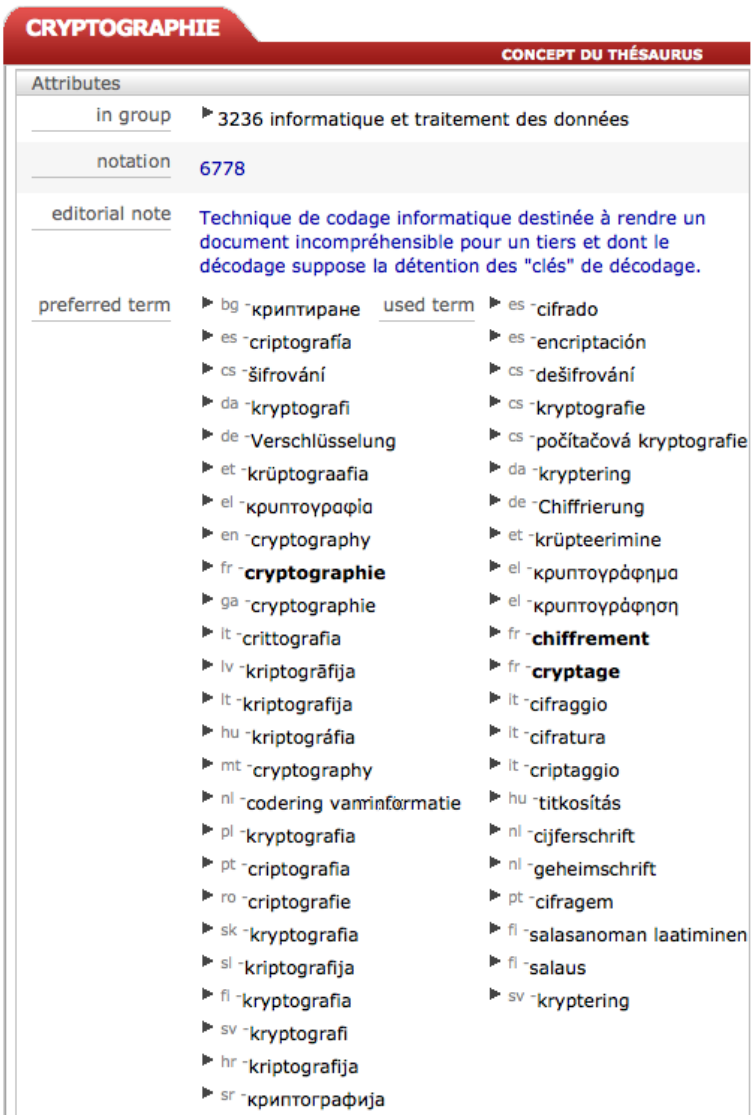

FIGURE 8.14 – Capture d'écran ITM : visualisation d'un concept et de ses termes associés. L'élément *SimpleNonPreferredTerm* du modèle UniMoKR a été renommé « used term » pour ce projet.

concepts. Ces attributs permettent l'ajout de métadonnées sur chaque élément (*e.g.* proposé, validé, à traduire, en traduction, etc.).

8.3.3 Résultats et Discussion

Notre modèle intégré à l'outil ITM est aujourd'hui utilisé au quotidien pour maintenir le thésaurus Européen Eurovoc. Une analyse a été menée pour tester la

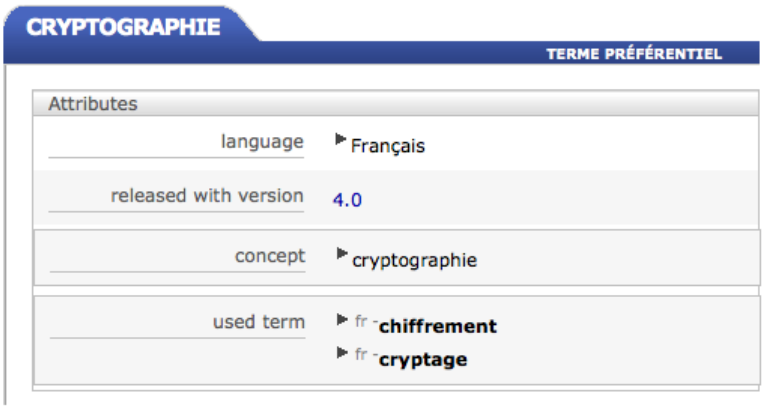

FIGURE 8.15 – Capture d'écran ITM : visualisation d'un terme préféré et de ses termes synonymes et concept associés. L'élément *SimpleNonPreferredTerm* du modèle UniMoKR a été renommé « used term » pour ce projet.

possibilité de réutiliser et étendre notre modèle pour d'autres thésaurus de l'Union Européenne. Le résultat étant concluant, nous commençons ces nouveaux projets de mise en œuvre de notre solution.

Le projet Eurovoc nous a confrontés à une description terminologique fine et nous a permis de faire évoluer notre modèle UniMoKR pour prendre en compte la définition indépendante de termes. Ce projet a démontré l'utilité des groupes de concepts et leur imbrication. Toutefois, il a mis en évidence le besoin de représenter le versioning de SOC. Nous rediscuterons dans le chapitre 9, de cette perspective.

8.4 LERUDI

8.4.1 Présentation

8.4.1.1 Contexte

Comment un médecin urgentiste peut-il connaître rapidement les informations essentielles du dossier d'un patient de façon à garantir la meilleure prise en charge de celui-ci ? Pour répondre à cette question, un groupe de travail a décidé de lancer un projet de prototype d'outil d'aide à la consultation du dossier patient en situation d'urgence. Ce groupe réunit des représentants du Service d'Aide Médicale Urgente (SAMU) de France et de la Société Française de Médecine d'Urgence (SFMU), des chercheurs de l'Agence des Systèmes d'Information Partagés de Santé (ASIP-Santé), de l'INSERM, du CISMeF et du LERTIM. Cet outil informatique doit permettre d'extraire les informations jugées les plus importantes d'un dossier patient préala-

blement indexé et ainsi améliorer la fiabilité du diagnostic lors de la prise en charge du patient en situation d'urgence hospitalière.

8.4.1.2 Objectifs

Les objectifs de ce projet sont de :

- créer et ré-utiliser les Systèmes d'Organisation de la Connaissance nécessaires pour représenter et caractériser l'information des documents des patients ;
- élaborer une Ressource Termino-Ontologique (RTO) à partir des SOC préalablement identifiés ;
- indexer et annoter les documents grâce à cette RTO ;
- développer des interfaces utilisateurs qui s'appuient sur les documents annotés pour proposer les informations selon leur importance, de manière condensée et ergonomique.

Notre responsabilité dans ce projet se limite aux deux premiers points et s'intéresse à la construction d'une RTO à partir de SOC et à la mise à disposition de cette ressource aux autres activités du projet.

8.4.2 Mise en œuvre

La figure 8.16 montre les différentes utilisations de la RTO pour la réalisation du projet LERUDI. La qualité des informations finales proposées aux urgentistes dépend essentiellement des qualité et richesse de cette RTO. En effet les phases d'annotation, d'inférence et d'indexation reposent sur la structuration formelle et la richesse linguistique de la RTO.

Exemple de l'importance de la structuration formelle de la RTO.

Prenons l'exemple d'une question importante que se pose l'urgentiste au sujet d'un patient : « Mon patient a-t-il déjà été infecté par une entérobactérie ? ». Considérons que ce patient a un document annoté avec le concept « Salmonelle ». Pour que le système puisse déduire que la salmonelle est une entérobactérie, il faut que la RTO déclare que le concept « Salmonelle » entretient une relation transitive de spécialisation avec le concept « Entérobactérie ». Ainsi la réponse à la question de l'urgentiste sera positive même si le document du patient n'est pas directement annoté avec le concept plus général d'entérobactérie.

Exemple de l'importance de la richesse linguistique de la RTO.

L'annotation des syntagmes nominaux « paracétamol », « Dafalgan » et « paraml. » nécessite que la RTO comporte un concept unique représentant

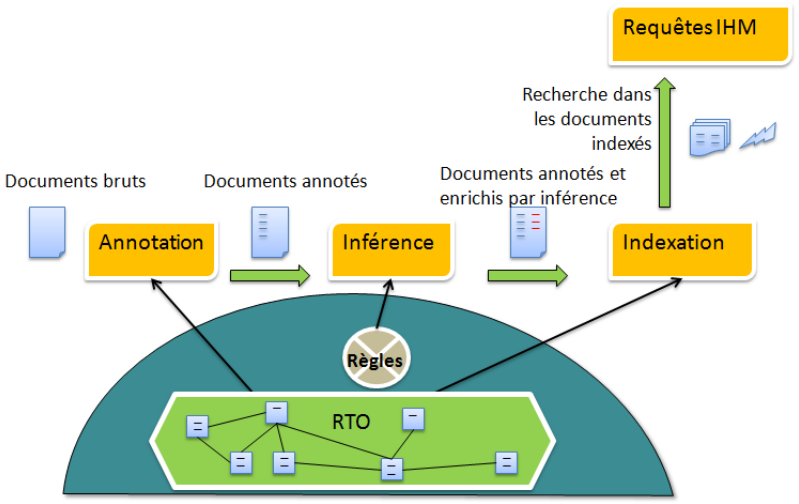

FIGURE 8.16 – Utilisation de la RTO dans le projet LERUDI. La Ressource Termino-Ontologique supporte les actions d'annotation, d'inférence et d'indexation.

ces trois syntagmes et que l'on dispose des termes relatifs à la molécule chimique (paracétamol) et à la spécialité médicamenteuse[8] ou à son nom de fantaisie (Dafalgan).

La RTO est élaborée à partir d'une ontologie du domaine des urgences alignée à 10 SOC pertinents pour ce domaine dont CIM-10, SNOMED 3.5, MedDRA, ATC. Les SOC constituent un vocabulaire contrôlé, sur lequel s'appuient les fonctions d'analyse (annotation) des documents des patients. Les SOC médicaux sont utilisés pour fournir des lexicalisations pour les concepts de l'ontologie et augmenter les chances de détection dans les documents traités. Outre la détection des concepts par l'annotateur sémantique, l'ontologie des urgences est utilisée pour typer l'information extraite. Toute expression annotée est indexée avec le type du concept. On se sert de ces types pour filtrer et trier l'information présentée à l'utilisateur. Par exemple, on peut ne présenter que les expressions de type pathologie ou médicament.

Les référentiels sémantiques fournis par les laboratoires auprès de l'ASIP-Santé sont intégrés au sein d'un unique serveur de référentiels dans l'outil ITM de MON-

8. En pharmacologie, une spécialité désigne une forme de médicament proposée par une marque *e.g.* « DAFALGAN 1 g, comprimé effervescent ». « DAFALGAN » est le nom de fantaisie du produit vendu en pharmacie (terme recommandé par l'Agence Française de Sécurité Sanitaire des Produits de Santé : AFSSAPS). Cette spécialité est composée de la substance chimique active « paracétamol ». Pour plus d'information, se reporter au site Web de l'AFSSAPS http://bit.ly/rhWZtr

DECA. Nous avons utilisé notre modèle UniMoKR intégré à l'outil ITM pour la représentation de ces SOC. La plupart des terminologies utilisées par ce projet ont déjà été modélisées dans le cadre du projet InterSTIS. Il s'agit d'étendre le modèle UniMoKR pour prendre en compte les spécificités de chaque SOC. L'ontologie des urgences développée par l'INSERM est dans un format OWL DL. Nous avons donc appliqué une transformation de modèle pour qu'il soit compatible avec notre propre modèle. Cette transformation a consisté à adapter la représentation en classes de l'ontologie à une représentation en concepts. Les laboratoires CISMeF, LERTIM et INSERM ont exécuté des alignements pour la mise en correspondance de l'ontologie avec les autres SOC. Ces correspondances ont également été importées dans le serveur.

8.4.3 Résultats et discussion

L'ontologie des urgences est le principal référentiel manipulé dans le projet. C'est à travers lui et ses correspondances, que l'on accède aux termes contenus dans les autres SOC. Grâce aux correspondances entre l'ontologie et les terminologies, il est possible de construire une nouvelle ressource RTO qui conserve l'ontologie en ajoutant à ses concepts les termes trouvés sur les terminologies alignées. Le fait de disposer, en plus de l'ontologie, de nombreuses manières d'exprimer un concept, augmente les chances de détection des syntagmes pertinents dans les documents. Pour construire et mettre à disposition la RTO aux autres acteurs du projet, nous procédons en deux étapes :

Enrichissement lexical des concepts. Une première étape est l'enrichissement des concepts de l'ontologie des urgences grâce aux termes des concepts des SOC mis en correspondance. Les alignements effectués entre l'ontologie des urgences et les autres SOC permettent d'établir des liens d'équivalence *exactMatch*[9] entre concepts. La RTO est construite automatiquement à l'export grâce à un pré-traitement qui consiste à ajouter des termes aux concepts de l'ontologie des urgences (*cf.* figure 8.17). Pour cela, nous regardons chaque concept de cette ontologie ; pour ceux qui ont une correspondance avec un autre concept d'un SOC, nous recopions le ou les termes du concept aligné sur le concept de l'ontologie. De cette manière, nous avons des concepts avec des formes lexicales variées de désignation. Pour nous donner la possibilité de détecter les noms de fantaisie (marque) des médicaments, nous utilisons

9. Nous n'avons volontairement utilisé que les liens de stricte équivalence pour éviter d'ajouter des termes trop éloignés et donc qui amènent du bruit dans les détections futures.

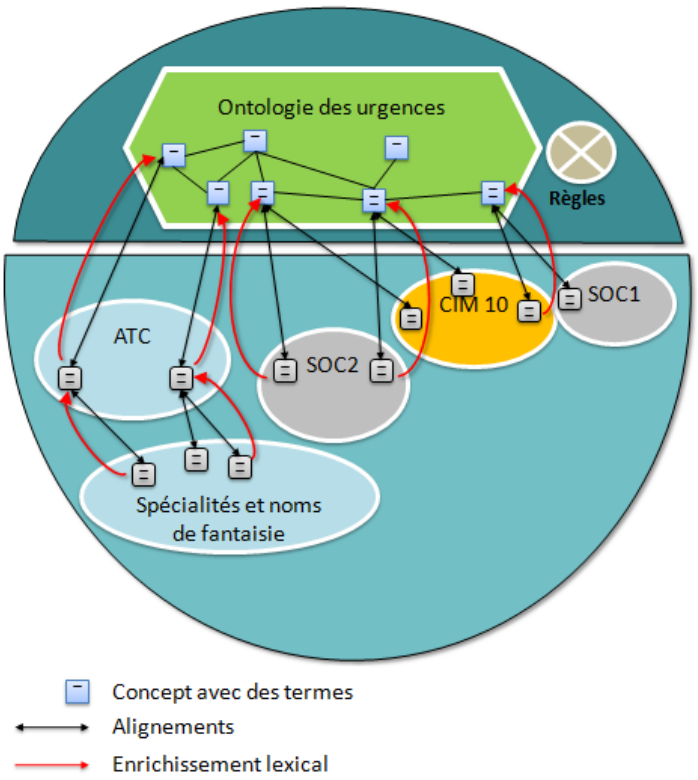

FIGURE 8.17 – Processus d'enrichissement de l'ontologie des urgences à partir des SOC. Grâce aux correspondances entre les concepts de l'ontologie des urgences et des concepts des autres SOC, nous recopions les termes de ces SOC sur les concepts de l'ontologie. Ce processus conduit à l'élaboration d'une ontologie enrichie appelée RTO.

les tables de correspondance [10] entre les substances chimiques décrites dans l'ATC et les noms de fantaisie, mises à disposition par l'Agence Française de Sécurité SAnitaire des Produits de Santé (AFSSAPS). Nous ajoutons les noms de fantaisie sur les concepts ATC avant de recopier les termes de l'ATC (et donc ceux des noms de fantaisie) sur l'ontologie.

Export SKOS. Une deuxième étape lors de l'export de la RTO optimisée pour l'annotation et l'indexation, est sa mise à disposition au format SKOS. Maintenant que les concepts de l'ontologie des urgences sont enrichis par des formes lexicales provenant des termes associés aux concepts des autres SOC, nous pouvons effectuer une transformation de modèles de cette ontologie enrichie vers une représentation de la RTO en SKOS. Nous utilisons pour cela la méthode

10. Ces tables sont mises à disposition par l'Agence Française de Sécurité SAnitaire des Produits de Santé (AFSSAPS) http://bit.ly/qg2755

de transformation de modèles décrite en section 7.1.

L'intégration des SOC sous un même format et le service de transformation RDF (capable d'effectuer des pré-traitements) permet de générer une ressource termino-ontologique qui comporte une lexicalisation suffisante pour supporter les actions d'indexation, d'inférence et d'indexation des dossiers patients. Ce projet a démontré l'intégration de multiples SOC et la capacité à mettre à disposition une ressource construite à partir de différents traitements de requêtes et de transformations.

Chapitre 9

Conclusion et perspectives

Nous venons de présenter dans cette thèse le fruit de nos travaux de recherche : la création d'un modèle de représentation commune de SOC. Ce modèle enrichit l'outil ITM en proposant un ensemble de services et de méthodes pour la gestion efficace de SOC. Cette démarche, appliquée notamment à la discipline médicale, s'appuie sur les théories, les méthodes, les techniques et les projets développés tant en Ingénierie des Connaissances, qu'en Ingénierie des Modèles. Nos travaux s'intègrent dans l'approche du Web sémantique.

Pour répondre à la problématique énoncée en section 5.4, nous soutenons la thèse suivante :

L'élaboration d'un modèle de représentation commune de SOC est une solution adaptée pour (i) pallier l'hétérogénéité de ces référentiels, (ii) favoriser l'interopérabilité sémantique au sein d'un Système d'Information et (iii) proposer des services unifiés quelque soit le SOC.

Pour débuter cette thèse, nous avons présenté les notions qui constituent les fondements de notre recherche. À partir de l'étude des modèles et des méthodes de modélisation, nous avons construit notre méthodologie de travail. L'analyse de la représentation et de l'organisation de la connaissance nous a permis d'identifier les éléments de notre modèle. Grâce aux langages, recommandations et projets existants dans la littérature, nous avons architecturé notre modèle et développé des services d'accès aux SOC.

La méthodologie que nous avons mise en place repose essentiellement sur l'approche MDA de l'Ingénierie des Modèles. Dans un premier temps, cette approche nous a permis d'élaborer le modèle UniMoKR grâce aux projets, langages et bonnes pratiques existants sans se figer dans un langage informatique particulier. Lors de cette étape, nous avons pu mener en parallèle la composition de notre modèle et l'étude des langages formels. Dans un deuxième temps, nous avons représenté le modèle UniMoKR avec le langage OWL DL. Ce choix a été motivé par son statut standard, son adoption par la communauté d'utilisateurs, par le nombre des outils disponibles et sa compatibilité avec les technologies du Web sémantique. En effet, nous devons nous intégrer au maximum dans la vision du Web sémantique pour le

partage de connaissances organisées au sein de SOC. Dans un troisième et dernier temps, nous avons mis en place des règles de transformation pour passer de notre modèle exprimé en OWL DL vers des modèles qui reposent sur RDF tels que SKOS.

Notre travail de modélisation a été soumis au groupe de recherche pour l'élaboration de la nouvelle norme ISO 25964. Ces experts ont ajouté à la spécification de cette future norme, la primitive de groupes de concepts que nous avons définie. Fondé sur de bonnes pratiques, notre modèle a donné lieu à la proposition de deux patrons de modélisation sur le portail Ontology Design Pattern [1] pour l'articulation termino-conceptuelle et pour la gestion des groupements de concepts. Le modèle UniMoKR est une solution efficace pour mettre à disposition des systèmes d'organisation de la connaissance à priori non interopérables.

Nos travaux ont été intégrés avec succès au sein de quatre applications, tant dans des projets de recherche (InterSTIS et LERUDI) que dans des projets commercialisés (AnaBio et Eurovoc). Ces résultats et l'utilisation commerciale de notre modèle couplé à l'outil ITM, confortent notre thèse. Des travaux de recherches sur l'interopérabilité avec les standards d'échanges d'informations sont utiles pour nos projets. Nous poursuivrons ces efforts notamment dans le domaine de la santé avec les standards comme HL7 et IHE.

Les SOC sont trop souvent considérés comme une simple liste d'éléments que l'on peut maintenir avec n'importe quel outil ou encore comme un dictionnaire qui peut être utilisé tel quel dans un Système d'Information. Ces idées reçues cachent une tout autre réalité : un SOC est un objet de connaissance complexe qui prend place dans un processus éditorial lui aussi complexe et repose sur des outils dédiés. Détaillons ces deux points :

L'importance d'un outil adapté. La connaissance contenue dans un SOC est présente sous forme d'un graphe qui possède des objets, des relations sémantiques, des propriétés. La représentation d'un graphe en constante évolution se prête difficilement à la gestion par un outil tel qu'un tableur. En effet la structure figée et les cases d'un tableur limitent l'expressivité de représentation des SOC (*e.g.* relations n-aires, relation hiérarchique) et sont la source d'ambiguïtés d'interprétation. Le projet AnaBio est représentatif de nombreux projets que nous avons rencontrés. Dès lors que le processus éditorial d'un SOC (i) fait intervenir plusieurs utilisateurs et la notion de multilinguisme, (ii) utilise une hiérarchie ou des relations entre les concepts, (iii) est trop volumineux, un outil non adapté (dans ce cas de type tableur) ne peut plus supporter ce processus. Un SOC est un graphe de connaissances ; sa représentation sous

1. Voir : http://ontologydesignpatterns.org

forme de table (dans un tableur ou en base de données relationnel) n'est pas adaptée. Le modèle UniMoKR que nous avons défini est exprimé dans un formalisme de représentation par réseaux sémantiques de connaissances. Un outil capable de gérer ce type de représentation et sa sémantique formelle associée est pleinement adapté à la représentation des connaissances contenues dans un SOC. Le logiciel ITM est un de ces outils.

Un processus éditorial complexe. L'intégration et l'utilisation d'un SOC nécessite également un traitement de la connaissance de ce SOC pour le rendre disponible aux formats classiques du SI. L'activité de publication permet de mettre les connaissances d'un SOC au format adéquat pour leur pleine utilisation.

Actuellement, nous prenons en compte les différentes versions de SOC comme des images séparées sans rapport entre leurs éléments constitutifs. Cette approche ne permet pas de suivre l'évolution d'un concept à travers les version d'un SOC. En revanche nous prenons en compte la dépréciation d'un concept en faveur d'un autre. Il serait intéressant d'analyser comment concilier la gestion du versionning à notre modèle.

Détaillons maintenant comment nos travaux répondent aux questions soulevées en section 5.2 :

- *Comment placer ces SOC à un même niveau d'interopérabilité pour pouvoir lier leurs connaissances ?* La solution que nous mettons en place repose sur une approche centralisée par harmonisation. Nous avons défini le modèle UniMoKR pour la représentation de SOC. Ce modèle sert de langage pivot et permet l'interopérabilité syntaxique et sémantique entre ces référentiels. Cette approche nécessite la transformation des données de leur format d'origine en données respectant le modèle que nous proposons. Notre modèle propose la représentation du contenu des SOC sous forme d'instances. Le modèle UniMoKR est donc à un niveau méta d'abstraction par rapport au contenu des SOC. Les instances de ces SOC ne rentrent pas dans le périmètre du modèle UniMoKR et sont généralement gérées dans un autre processus dans un Système d'Information. On y distingue le serveur multi-terminologique rassemblant l'ensemble des SOC utiles au système et l'application d'un SOC dans son contexte avec ses instances.
- *Comment prendre en compte l'hétérogénéité des SOC sans altérer l'information spécifique à chacun d'eux ?* Le modèle UniMoKR que nous proposons utilise les bonnes pratiques de représentation des connaissances et factorise les éléments communs des SOC. Néanmoins chaque SOC possède des spécifi-

cités qu'il nous faut représenter. La propriété d'extensibilité de notre modèle permet la représentation de ces spécificités intégrées au modèle noyau. La solution que nous proposons fournit une représentation unifiée de SOC et a donc la possibilité d'offrir des services identiques quel que soit le référentiel interrogé. Cette représentation unifiée nécessite toutefois la transformation des données d'origine. Même si notre modèle est extensible, certaines transformations peuvent réduire l'expressivité du modèle d'origine. Ces cas révèlent des possibilités de perfectionnement de notre modèle commun et méritent d'être étudiés avec attention.

- *Quel langage de représentation utiliser pour rester compatible avec les standards existants ?* Nous utilisons le langage OWL DL qui grâce à sa sémantique formelle et ses primitives permet de représenter les éléments de notre modèle et leur sens associé. Ce langage a l'avantage d'être le standard principal de représentation en Ingénierie des Connaissances pour la représentation d'ontologies (SOC dont l'expressivité formelle est la plus complexe) et notamment pour le Web sémantique. Il bénéficie également d'une large communauté d'utilisateurs et de nombreux outils. Exprimer notre modèle avec OWL rend possible la transformation de SOC depuis notre modèle vers des modèles standards tels que SKOS ou IHE.
- *Comment gérer ces différents référentiels en sachant les recouvrements potentiels ?* L'approche que nous utilisons pour l'intégration des SOC respecte l'indépendance des concepts similaires ou équivalents. Nous considérons les correspondances entre concepts de SOC comme des artefacts distincts. À ce titre un jeu de correspondance est également intégré tout en conservant son indépendance. Cette indépendance permet l'évolution autonome de SOC ou de correspondances. Nous avons adopté la modélisation proposée par le standard SKOS et l'avons étendue (par la réification des relations de correspondance) pour qu'elle convienne à l'activité d'édition.
- *Quel outil choisir pour l'intégration et l'interprétation de notre modèle formel ?* Un modèle seul ne permet pas de proposer d'enregistrer, de manipuler des informations ni d'offrir des services. C'est le couplage du modèle à un outil qui permet de rendre ces actions effectives. Dans le cadre de notre thèse en collaboration avec l'entreprise MONDECA, nous avons utilisé l'outil ITM qui permet l'interprétation de notre modèle formel. Toutefois nous avons étudié la faisabilité d'intégration de notre modèle au sein d'outils ouverts.
- *Quels services d'édition et d'accès à l'information peut-on proposer en intégrant notre modèle à un outil dédié ?* Les services d'édition sont gérés nati-

vement au sein de l'outil ITM qui propose des interfaces et des services et qui possède le moyen de stocker la connaissance issue de notre modèle. Nous avons développé un ensemble de services dédiés à l'accès de SOC et de leur contenu. Ces services respectent les recommandations de CTS 2. Le fait que notre modèle propose un noyau unique quel que soit le SOC, permet de fournir des services identiques pour chacun d'eux. Ce résultat est très important dans la mise en place de notre solution car il réduit les coûts d'interopérabilité entre logiciels (une seule communication à configurer quel que soit le référentiel).

– *Comment importer ou exporter tout ou partie des SOC vers et depuis notre modèle ?* Nous avons enfin utilisé le langage SPARQL opérant sur un graphe RDF pour l'expression de règles de transformation de modèles. Ces transformations permettent d'importer ou d'exporter tout ou partie de SOC depuis (et vers) notre modèle vers (ou depuis) tout langage dont l'expression est fondée sur RDF.

Pour conclure ce chapitre, nous revenons sur nos principales contributions, d'ordres méthodologique, technique et pratique :

Contributions méthodologiques. L'originalité de notre recherche réside en l'utilisation conjointe de méthodologies issues de l'Ingénierie des Connaissances, de l'Ingénierie des Modèles et de la sémantique. Notre approche tend à rendre générique la modélisation de SOC et à mener à des apports dans ces communautés. Ainsi deux sous-parties de notre modèle ont été proposées en bonnes pratiques de modélisation. L'une d'elles a même été intégrée à la prochaine norme ISO 25964.

Contributions techniques. Notre travail de recherche a également un caractère original de par son intégration au sein de l'entreprise MONDECA. L'utilisation de notre métamodèle intégré à l'outil ITM a permis le développement de nouvelles techniques de transformation de modèles qui permettent des conversions de connaissances dans divers formats standards. Par ailleurs nous avons développé des interfaces d'accès à l'information originale et intuitive qui ont été déjà commercialisées avec la suite logicielle de MONDECA.

Contributions pratiques. Notre travail de recherche a abouti à de nombreuses mises en application tant dans le domaine de la recherche que dans des cadres privés. L'opérationnalisation de notre solution par des institutions comme l'Assistance Publique - Hôpitaux de Paris, l'Office des publications Européen ou l'ASIP est pour nous un résultat très important. La diversité des domaines de mise en œuvre est un indicateur de la généricité de notre solution et une garantie des possibilités de son adaptation future.

Les résultats obtenus jusqu'à présent concernant l'utilisation du modèle UniMoKR et des services que nous avons développés nous encouragent à poursuivre les recherches et à améliorer les solutions que nous proposons. Nous poursuivrons cet axe de recherche dans le cadre de futurs projets tant dans notre entreprise MONDECA que dans notre laboratoire de recherche INSERM.

Bibliographie

[Adamczewski 2002] Georges Adamczewski. *Qu'est-ce qu'un concept ?, URL : http ://www.biblioconcept.com/textes/concept.htm*, 2002. 101

[AFN 1981] *Z 47-100 :1981*, 1981. 32

[Aitken 2003] JS Aitken, BL Webber et JBL Bard. *Part-of relations in anatomy ontologies : a proposal for RDFS and OWL formalisations.* In Pacific Symposium on Biocomputing 2004 : Hawaii, USA, 6-10 January 2004, page 166. World Scientific Pub Co Inc, 2003. 106

[Amann 2010] B. Amann et I. Fundulaki. *Integrating ontologies and thesauri to build RDF schemas.* Research and Advanced Technology for Digital Libraries, pages 672–672, 2010. 35

[Amardeilh 2009] F. Amardeilh, C. Bousquet, S. Guillemin-Lanne, M. Wiss-Thébault, L. Guillot, D. Delamarre, L.L. Louet et A. Burgun. *A Knowledge Management Platform for Documentation of Case Reports in Pharmacovigilance.* Medical Informatics Europe, 2009. 47

[Auer 2007] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak et Z. Ives. *Dbpedia : A nucleus for a web of open data.* The Semantic Web, pages 722–735, 2007. 17

[Aussenac-Gilles 2005a] N. Aussenac-Gilles et D. Sörgel. *Text analysis for ontology and terminology engineering.* Applied Ontology, vol. 1, no. 1, pages 35–46, 2005. 99

[Aussenac-Gilles 2005b] Nathalie Aussenac-Gilles. *Méthodes ascendantes pour l'ingénierie des connaissances.* Habilitation à diriger des recherches, Université Paul Sabatier, Toulouse, France, décembre 2005. 6

[Austin 1968] C.J. Austin. *MEDLARS, 1963-1967.* 1968. 39

[Baader 2003] F. Baader. The description logic handbook : theory, implementation, and applications. Cambridge Univ Pr, 2003. 57

[Baader 2007] F. Baader, D. Calvanese, D.L. McGuinness, D. Nardi et P.F. Patel-Schneider. The description logic handbook. Cambridge university press, 2007. 63

[Bachimont 2000] B. Bachimont. *Engagement sémantique et engagement ontologique : conception et réalisation d'ontologies en ingénierie des connaissances.* Ingénierie des connaissances : évolutions récentes et nouveaux défis, pages 305–323, 2000. 75

[Bachimont 2004] B. Bachimont. *Arts et sciences du numérique : Ingénierie des connaissances et critique de la raison computationnelle.* Mémoire d'Habilitation à Diriger la Recherche, 2004. 4, 6, 12

[Baget 2002] Jean-François Baget et Marie-Laure Mugnier. *Extensions of Simple Conceptual Graphs : the Complexity of Rules and Constraints.* J. Artif. Intell. Res. (JAIR), vol. 16, pages 425–465, 2002. 62

[Baget 2005] Jean-François Baget. *RDF Entailment as a Graph Homomorphism.* In International Semantic Web Conference, pages 82–96, 2005. 64

[Baget 2008] Jean-François Baget, Michel Leclère, Marie-Laure Mugnier et Eric Salvat. *DL-SR : a Lite DL with Expressive Rules : Preliminary Results.* In Description Logics, 2008. 64

[Baneyx 2007] A. Baneyx. *Construire une ontologie de la Pneumologie : Aspects théoriques, modèles et expérimentations.* PhD thesis, Université Pierre et Marie Curie - Paris 6, 2007. 37

[Bellatreche 2004] L. Bellatreche, G. Pierra, D.N. Xuan, D. Hondjack et Y.A. Ameur. *An a priori approach for automatic integration of heterogeneous and autonomous databases.* In Database and Expert Systems Applications, pages 475–485. Springer, 2004. 123

[Berners-Lee 1998] T. Berners-Lee. *Semantic web road map.* 1998. 16

[Berners-Lee 2001] T. Berners-Lee, J. Hendler et O. Lassila. *The semantic web.* Scientific American, vol. 284, no. 5, pages 34–43, 2001. 15, 57

[Bézivin 2001] J. Bézivin. *From object composition to model transformation with the MDA.* In Proceedings of TOOLS, pages 350–354, 2001. 22

[Bézivin 2004] J. Bézivin. *Sur les principes de base de l'ingénierie des modèles.* RSTI-L'Objet, vol. 10, no. 4, pages 145–157, 2004. 20, 21

[Bézivin 2005] J. Bézivin. *On the unification power of models.* Software and Systems Modeling, vol. 4, no. 2, pages 171–188, 2005. 5, 23

[Biezunski 1999] M. Biezunski, M. Bryan et S. Newcomb. *ISO/IEC 13250 : 2000 Topic Maps : Information Technology–Document Description and Markup Language.* International Organization for Standarization (ISO) and International Electrotechnical Commission (IEC), 1999. 58

[Binding 2006] C. Binding et D. Tudhope. *KOS at your service : programmatic access to knowledge organisation systems.* Journal of Digital Information, vol. 4, no. 4, 2006. 3, 31

[Bizer 2009a] C. Bizer, T. Heath et T. Berners-Lee. *Linked data-the story so far.* Int. J. Semantic Web Inf. Syst., vol. 5, no. 3, pages 1–22, 2009. 16

[Bizer 2009b] C. Bizer et A. Schultz. *The berlin sparql benchmark.* Int. J. Semantic Web Inf. Syst, vol. 5, no. 2, pages 1–24, 2009. 125

[Bodenreider 2004] O. Bodenreider. *The unified medical language system (UMLS) : integrating biomedical terminology.* Nucleic acids research, vol. 32, no. suppl 1, page D267, 2004. 74

[Bodenreider 2006] O. Bodenreider. *Lexical, terminological and ontological resources for biological text mining.* Text mining for biology and biomedicine, pages 43–66, 2006. 43

[Bodenreider 2008] O. Bodenreider. *Biomedical ontologies in action : role in knowledge management, data integration and decision support.* Yearb Med Inform, vol. 67, page 79, 2008. 43

[Bourigault 2004] D. Bourigault, N. Aussenac-Gilles et J. Charlet. *Construction de ressources terminologiques ou ontologiques à partir de textes : un cadre unificateur pour trois études de cas.* Revue d'Intelligence Artificielle, vol. 18, no. 4, page 24, 2004. 31

[Brickley 2004] D. Brickley. *RDF vocabulary description language 1.0 : RDF schema.* http ://www. w3. org/tr/rdf-schema/, 2004. 59

[BS8723 2008] BS8723. *Structured vocabularies for information retrieval, Part 4 : Interoperability between vocabularies,*, 2008. 67

[Carloni 2009] O. Carloni, M. Leclère et M.L. Mugnier. *Introducing reasoning into an industrial knowledge management tool.* Applied Intelligence, vol. 31, no. 3, pages 211–224, 2009. 64

[Carroll 1995] J.M. Carroll. *Scenario-based design : envisioning work and technology in system development.* 1995. 29

[Charlet 1996] J. Charlet, B. Bachimont, J. Bouaud et P. Zweigenbaum. *Ontologie et réutilisabilité : expérience et discussion.* Acquisition et Ingénierie des connaissances : tendances actuelles, pages 69–87, 1996. 75

[Charlet 2002] Jean Charlet. *L'Ingénierie des connaissances : développements, résultats et perspectives pour la gestion des connaissances médicales.* Rapport technique, Université Paris 6, décembre 2002. 4, 12, 35

[Charlet 2006] J. Charlet, B. Bachimont et M.C. Jaulent. *Building medical ontologies by terminology extraction from texts : an experiment for the intensive care units.* Computers in biology and medicine, vol. 36, no. 7-8, pages 857–870, 2006. 20, 35

[Choi 2006] N. Choi, I.Y. Song et H. Han. *A survey on ontology mapping.* ACM Sigmod Record, vol. 35, no. 3, pages 34–41, 2006. 30

[Chute 1999] CG Chute, PL Elkin, DD Sherertz et MS Tuttle. *Desiderata for a clinical terminology server.* In Proceedings of the AMIA Symposium, page 42. American Medical Informatics Association, 1999. 6, 84

[Chute 2000] C.G. Chute. *Clinical Classification and Terminology.* Journal of the American Medical Informatics Association, vol. 7, no. 3, page 298, 2000. 6

[Cimino 2001] J.J. Cimino*et al.* *Terminology tools : state of the art and practical lessons.* Methods of information in medicine, vol. 40, no. 4, pages 298–306, 2001. 30

[Cimino 2009] J.J. Cimino, T.F. Hayamizu, O. Bodenreider, B. Davis, G.A. Stafford et M. Ringwald. *The caBIG terminology review process.* Journal of biomedical informatics, vol. 42, no. 3, pages 571–580, 2009. 77

[Clarke 2008] S.G.D. Clarke. *ISO 2788+ ISO 5964+ Much Energy= ISO 25964.* Bulletin of the American Society for Information Science and Technology, vol. 35, no. 1, pages 31–33, 2008. 68

[Cormont 2002] S Cormont, A Erman, Y Burckel et A Carayon. *Names-Lab : a model for the standardization of biology message exchanges.* Ann Biol Clin, vol. Mar-Apr, pages 173–81, 2002. 140

[Cormont 2008] Sylvie Cormont, Antoine Buemi, Thierry Horeau, Pierre Zweigenbaum et Éric Lepage. *Construction of a dictionary of laboratory tests mapped to LOINC at AP-HP.* In AMIA, 2008. 140

[Côté 1993] R Côté, D Rothwell, J Palotay, R Beckett et L Brochu. *The systematized nomenclature of human and veterinary medicine.* Rapport technique, SNOMED International, Northfield, IL : College of American Pathologists, 1993. 36

[Cournot 1851] A.A. Cournot. Essai sur les fondements de nos connaissances et sur les caractères de la critique philosophique, volume 1. Librairie de L. Hachette, 1851. 31

[Daniel 2009] C. Daniel, A. Buemi, L. Mazuel, D. Ouagne et J. Charlet. *Functional requirements of terminology services for coupling interface terminologies to reference terminologies.* In Studies in health technology and informatics, volume 150, page 205, 2009. 87, 141

[Davies 2003] J. Davies, D. Fensel et F. Van Harmelen. Towards the semantic web : ontology-driven knowledge management. Wiley, 2003. 35

[Degoulet 1997] P. Degoulet, M. Fieschi et C. Attali. *Les enjeux de l'interopérabilité sémantique dans les systèmes d'information de santé*. Volume 9 Springer-Verlag France, Paris, 1997. 45

[Degoulet 1998] P. Degoulet et M. Fieschi. Informatique médicale. Elsevier Masson, 1998. 4

[Dolin 2001] R.H. Dolin, K. Spackman, A. Abilla, C. Correia, B. Goldberg, D. Konicek, J. Lukoff et C.B. Lundberg. *The SNOMED RT Procedure Model*. In Proceedings of the AMIA Symposium, page 139. American Medical Informatics Association, 2001. 37

[Dolin 2006] R.H. Dolin, L. Alschuler, S. Boyer, C. Beebe, F.M. Behlen, P.V. Biron et A. Shabo Shvo. *HL7 clinical document architecture, release 2*. Journal of the American Medical Informatics Association, vol. 13, no. 1, page 30, 2006. 53

[Dubuc 1977] R. Dubuc. *Qu'est ce que la terminologie*. Le banque des mots, vol. 13, 1977. 32

[Dusserre 1985] L. Dusserre et H. Ducrot. L'informatique mÃ©dicale. Presses universitaires de France, 1985. 4

[Euzenat 2004] J. Euzenat. *An API for ontology alignment*. The Semantic Web–ISWC 2004, pages 698–712, 2004. 30

[Euzenat 2007] J. Euzenat et P. Shvaiko. Ontology matching. Springer-Verlag New York Inc, 2007. 30, 87

[Favre 2006] J.M. Favre, J. Estublier et M. Blay-Fornarino. *L'ingénierie dirigée par les modèles : au-delà du MDA*. Hermes-Lavoisier, Cachan, France, 2006. 5

[Fayet-Scribe 1997] Sylvie Fayet-Scribe. *Chronologie des supports, des dispositifs spatiaux, des outils de repérage de l'information.*, Décembre 1997. 13

[Fayssoil 2010] A. Fayssoil. *Réflexes et mots-clés pour les ECN*. 2010. 134

[Fleurey 2006] F. Fleurey. *Langage et méthode pour une ingénierie des modèles fiable*. 2006. 5

[Fung 2005] K.W. Fung et O. Bodenreider. *Utilizing the UMLS for semantic mapping between terminologies*. In AMIA Annual Symposium Proceedings, volume 2005, page 266. American Medical Informatics Association, 2005. 74, 96

[Ganascia 1998] J.G. Ganascia. Le petit trésor, dictionnaire de l'informatique et des sciences de l'information. Flammarion, 1998. 12

[Gandon 2002] F. Gandon. *Ontology engineering : A survey and a return on experience.* 2002. 27, 29, 35

[Gandon 2008] F. Gandon et A. Giboin. *Vers des ontologies à l'état sauvage.* 2008. 36

[Gaudinat 2004] A. Gaudinat, M. Joubert, S. Aymard, L. Falco, C. Boyer et M. Fieschi. *WRAPIN : new generation health search engine using UMLS knowledge sources for MeSH term extraction from health documentation.* Medinfo, vol. 2004, pages 356–60, 2004. 135

[Giannangelo 2006] K. Giannangelo. Healthcare code sets, clinical terminologies, and classification systems. American Health Information Management Association (AHIMA), 2006. 44

[Gilchrist 2003] A. Gilchrist. *Thesauri, taxonomies and ontologies–an etymological note.* Journal of documentation, vol. 59, no. 1, pages 7–18, 2003. 33

[Goble 1994] C. Goble, P. Crowther et D. Solomon. *A medical terminology server.* In Database and Expert Systems Applications, pages 661–670. Springer, 1994. 76

[Golbreich 2007] C. Golbreich, M. Horridge, I. Horrocks, B. Motik et R. Shearer. *OBO and OWL : Leveraging semantic web technologies for the life sciences.* The Semantic Web, pages 169–182, 2007. 61

[Greenberg 2001] J. Greenberg. *Automatic query expansion via lexical–semantic relationships.* Journal of the American Society for Information Science and Technology, vol. 52, no. 5, pages 402–415, 2001. 44

[Grobe 2009] M. Grobe. *RDF, Jena, SparQL and the'Semantic Web'.* In Proceedings of the 37th annual ACM SIGUCCS fall conference, pages 131–138. ACM, 2009. 113

[Gruber 1993] T.R. Gruber. *A translation approach to portable ontology specifications.* Knowledge acquisition, vol. 5, pages 199–199, 1993. 34

[Guarino 1995] N. Guarino et P. Giaretta. *Ontologies and knowledge bases : Towards a terminological clarification.* Towards Very Large Knowledge Bases Knowledge Building and Knowledge Sharing, vol. 1, no. 9, pages 25–32, 1995. 34

[Hacid 2004] M.S. Hacid et C. Reynaud. *L'intégration de sources de données.* Revue I3 (Information Interaction Intelligence, vol. 4, no. 2, 2004. 46

[Hayes 2004] P. Hayes et B. McBride. *RDF semantics.* W3C recommendation, vol. 10, pages 38–45, 2004. 57, 59

[Heath 2011] Tom Heath et Christian Bizer. Linked data : Evolving the web into a global data space : Theory and technology, volume 1. Morgan & Claypool Publishers, 2011. 15, 35

[Heiler 1995] S. Heiler. *Semantic interoperability.* ACM Computing Surveys (CSUR), vol. 27, no. 2, pages 271–273, 1995. 46

[Hodge 2000] G. Hodge. Systems of knowledge organization for digital libraries. Citeseer, 2000. 3, 31

[Imel 2002] M. Imel. *A closer look : the SNOMED clinical terms to ICD-9-CM mapping.* Journal of AHIMA/American Health Information Management Association, vol. 73, no. 6, page 66, 2002. 96

[Ingenerf 1998] N. Ingenerf et W. Giere. *Concept-oriented standardization and statistics-oriented classification : continuing the classification versus nomenclature controversy.* Meth Inform Med, vol. 37, pages 527–39, 1998. 31, 49

[Jamoulle 2000] M. Jamoulle, M. Roland, J. Humbert et J.F. Brûlet. *Traitement de l'information médicale par la Classification Internationale des Soins Primaires 2ème version (CISP-2), assorti d'un glossaire de médecine générale, préparé par le Comité International de Classification de la Wonca.* Care Edition, Bruxelles, 2000. 74

[Joubert 2011] Michel Joubert*et al.* *Interopérabilité sémantique de terminologies de santé francophones.* IRBM, 2011. 96, 132

[Kayser 1997] D. Kayser. La représentation des connaissances. Hermes, 1997. 12

[Khayari 2006] M. Khayari, S. Schneider, I. Kramer et L. Romary. *Unification of multi-lingual scientific terminological resources using the ISO 16642 standard. The TermSciences initiative.* Arxiv preprint cs/0604027, 2006. 134

[Kleppe 2003] A.G. Kleppe, J. Warmer et W. Bast. Mda explained : the model driven architecture : practice and promise. Addison-Wesley Longman Publishing Co., Inc., 2003. 22

[Klyne 2004] G. Klyne et J.J. Carroll. *Resource description framework (RDF) : Concepts and abstract syntax. W3C Recommendation, 10 February 2004.* World Wide Web Consortium, 2004. 16

[Lassila 1999] O. Lassila et R.R. Swick. *Resource description framework (RDF) model and syntax.* World Wide Web Consortium, http ://www. w3. org/TR/WD-rdf-syntax, 1999. 59

[Laublet 2002] P. Laublet, C. Reynaud et J. Charlet. *Sur quelques aspects du Web sémantique.* Assises du GDR I, vol. 3, 2002. 46

[Le Bozec 2001] C.D. Le Bozec. *Gestion des connaissances multi-expertes en imagerie médicale" IDEM : images et diagnostics par l'exemple en médecine*. PhD thesis, Université Paris 6, 2001. 37

[Lefèvre 2000] Philippe Lefèvre. La recherche d'informations (du texte intégral au thésaurus). Hermès Science Publications, 2000. 32

[Lerat 1989] P. Lerat. *Les fondements théoriques de la terminologie*. La Banque des mots, pages 51–62, 1989. 26

[Limpens 2008] F. Limpens, F. Gandon et M. Buffa. *Rapprocher les ontologies et les folksonomies pour la gestion des connaissances partagées : un état de l'art.* 2008. 36

[Lin 2009] MC Lin, DJ Vreeman, CJ McDonald et SM Huff. *A Characterization of Local LOINC Mapping for Laboratory Tests in Three Large Institutions*. Methods Inf Med, vol. 2010, page 49, 2009. 139

[Lindberg 1993] D. Lindberg, BL Humphreys et AT McCray. *The unified medical language system*. Methods of Information in Medicine, vol. 32, no. 4, page 281, 1993. 73, 89

[LMF 2008] *Language resource management - Lexical markup framework (LMF)*, 2008. 69

[Lovis 2009] C. Lovis, T. DOUGLAS, E. Pasche, P. Ruch, D. Colaert et K. Stroetmann. *DebugIT : Building a European distributed clinical data mining network to foster the fight against microbial diseases*. Studies in health technology and informatics, vol. 148, page 50, 2009. 5

[Lussier 1998] YA Lussier, DJ Rothwell et RA Cote. *The SNOMED model : a knowledge source for the controlled terminology of the computerized patient record*. Methods of information in medicine, vol. 37, no. 2, pages 161–164, 1998. 32

[Mazuel 2009] L. Mazuel et J. Charlet. *Alignement entre des ontologies de domaine et la Snomed : trois études de cas*. 2009. 30, 87

[McBride 2004] B. McBride. *The resource description framework (RDF) and its vocabulary description language RDFS*. Handbook on Ontologies, pages 51–66, 2004. 60

[McDonald 2003] C.J. McDonald, S.M. Huff, J.G. Suico, G. Hill, D. Leavelle, R. Aller, A. Forrey, K. Mercer, G. DeMoor, J. Hook*et al.* *LOINC, a universal standard for identifying laboratory observations : a 5-year update*. Clinical chemistry, vol. 49, no. 4, page 624, 2003. 139

[McGuinness 2002] D.L. McGuinness, R. Fikes, J. Hendler et L.A. Stein. *DAML+OIL : an ontology language for the Semantic Web.* IEEE Intelligent Systems,, pages 72–80, 2002. 60

[Merabti 2010] T. Merabti. *Méthodes pour la mise en relations des terminologies médicales : contribution a l'interopérabilité sémantique Inter et Intra terminologique.* PhD thesis, Université de Rouen, 2010. 30, 74, 97

[Merrill 2010] G.H. Merrill. *Ontological realism : Methodology or misdirection ?* Applied Ontology, vol. 5, no. 2, pages 79–108, 2010. 99

[Mika 2005] P. Mika. *Ontologies are us : A unified model of social networks and semantics.* The Semantic Web–ISWC 2005, pages 522–536, 2005. 36

[Miles 2006] A. Miles. *SKOS : requirements for standardization.* In DC-2006 : Proceedings of the International Conference on Dublin Core and Metadata Aplications, pages 55–64, 2006. 64, 89

[Miller 2000] Paul Miller. *Interoperability : What is it and Why should I Want it ?* Ariadne, vol. 24, 2000. 45

[Minsky 1974] M. Minsky. *A framework for representing knowledge.* 1974. 56, 62

[Morbidoni 2007] C Morbidoni, A Polleres, G Tummarello et D Le Phuoc. *Semantic Web Pipes.* Rapport technique, DERI, November 2007. 71, 108, 112

[Motik 2009] B. Motik, B.C. Grau, I. Horrocks, Z. Wu, A. Fokoue et C. Lutz. *OWL 2 Web Ontology Language : Profiles.* W3C Recommendation, vol. 27, 2009. 61

[Nardi 2003] D. Nardi, R.J. Brachman*et al.* *An introduction to description logics.* The description logic handbook : theory, implementation, and applications, pages 1–40, 2003. 60

[Nowlan 1994] WA Nowlan, AL Rector, TW Rush et WD Solomon. *From terminology to terminology services.* In Proceedings of the Annual Symposium on Computer Application in Medical Care, page 150. American Medical Informatics Association, 1994. 83

[Noy 2004] N. Noy et A. Rector. *Defining N-ary relations on the Semantic Web : Use with individuals.* W3C Working Draft, vol. 21, 2004. 102

[Noy 2009] N.F. Noy, N.H. Shah, P.L. Whetzel, B. Dai, M. Dorf, N. Griffith, C. Jonquet, D.L. Rubin, M.A. Storey, C.G. Chute*et al.* *BioPortal : ontologies and integrated data resources at the click of a mouse.* Nucleic acids research, vol. 37, no. suppl 2, page W170, 2009. 78

[Oltramari 2002] A. Oltramari, A. Gangemi, N. Guarino et C. Masolo. *Restructuring WordNet's top-level : The OntoClean approach.* LREC2002, Las Palmas, Spain, 2002. 49

[OMG 2001] OMG. *MDA Guide Version 1.0.1*, 2001. 19

[Otman 1994] G. Otman. *Pourquoi parler de connaissances terminologiques et de bases de connaissances terminologiques.* La banque des mots, pages 5–27, 1994. 26

[Paquin 2010] Louis-Claude Paquin. *Le passage des termes aux concepts, URL : http ://www.ling.uqam.ca/sato/publications/bibliographie/Termes.htm.* Rapport technique, Centre d'analyse de textes par ordinateur Université du Québec Ã Montréal, 2010. 100

[Park 2003] J. Park et S. Hunting. Xml topic maps : creating and using topic maps for the web. Addison-Wesley Professional, 2003. 58

[Patel-Schneider 2004] P.F. Patel-Schneider, P. Hayes, I. Horrocks*et al. OWL web ontology language semantics and abstract syntax.* W3C recommendation, vol. 10, 2004. 57, 60

[Pathak 2009] J. Pathak, H.R. Solbrig, J.D. Buntrock, T.M. Johnson et C.G. Chute. *LexGrid : A Framework for Representing, Storing, and Querying Biomedical Terminologies from Simple to Sublime.* Journal of the American Medical Informatics Association, vol. 16, no. 3, pages 305–315, 2009. 78

[Pepper 2001] S. Pepper et G. Moore. *XML topic maps (XTM) 1.0.* TopicMaps.org Specification xtm1, 2001. 58

[Polleres 2007] A. Polleres, F. Scharffe et R. Schindlauer. *SPARQL++ for mapping between RDF vocabularies.* Lecture Notes in Computer Science, vol. 4803, page 878, 2007. 71, 108, 112

[Prud'hommeaux 2008] E. Prud'hommeaux et A. Seaborne. *SPARQL query language for RDF. W3C Recommendation 15 January 2008.* World Wide Web Consortium, 2008. 71

[Rastier 1995] F. Rastier. *Le terme : entre ontologie et linguistique Texto ! : http ://www.revue-texto.net/index.php ?id=568*, 1995. 26, 100

[Rector 1992] AL Rector, WA Nowlan et S. Kay. *Conceptual knowledge : the core of medical information systems.* Proc MEDINFO, pages 1420–6, 1992. 76

[Rector 1993] A. Rector et WA Nowlan. *The GALEN representation and integration language (GRAIL) kernel, version 1.* The GALEN Consortium for the EC AIM Programme.(Available from Medical Informatics Group, University of Manchester), 1993. 76

[Rector 1998] A.L. Rector. *Thesauri and formal classifications : terminologies for people and machines.* Methods of Information in Medicine, vol. 37, pages 501–509, 1998. 75, 89

[Rector 2006] Alan Rector, R. Qamar et T. Marley. *Binding Ontologies & Coding systems to Electronic Health Records and Messages.* In Proceedings of KR-MED, page 11. Citeseer, 2006. 103

[Roget 1856] P.M. Roget. Thesaurus of english words and phrases. Gould and Lincoln, 1856. 13

[Rosse 2003] C. Rosse, J.L.V. Mejino*et al.* *A reference ontology for biomedical informatics : the Foundational Model of Anatomy.* Journal of biomedical informatics, vol. 36, no. 6, pages 478–500, 2003. 36

[Rothenberg 1989] J. Rothenberg et United States. Defense Advanced Research Projects Agency. The nature of modeling. Citeseer, 1989. 20

[Schenk 2008] S. Schenk et J. Petrák. *Sesame RDF repository extensions for remote querying.* In ZNALOSTI Conf, 2008. 113, 125

[Scherrer 1997] J.R. Scherrer. *Concepts, knowledge and language in health-care information systems : follow-up 30 months later.* In Proceedings of the IMIA Conference on Natural Language and Medical Concept Representation ; Jacksonville, Florida, pages 5–8. Citeseer, 1997. 26

[Schober 2010] D. Schober, M. Boeker, J. Bullenkamp, C. Huszka, K. Depraetere, D. Teodoro, N. Nadah, R. Choquet, C. Daniel et S. Schulz. *The DebugIT Core Ontology : semantic integration of antibiotics resistance patterns.* MEDINFO, pages 1060–1064, 2010. 46

[Sherman 2000] C. Sherman. *Humans Do It Better : Inside the Open Directory Project.* Online, 2000. 33

[Slodzian 2000] Monique Slodzian. *Wordnet : what about its linguistic relevancy ?* In Rose Dieng, editeur, Proceedings of the EKAW 2000 Workshop on Ontologies and Texts, volume 51, Juan-les-Pins, France, 2000. 26, 49, 100

[Smith 2004] B. Smith. *Beyond concepts : ontology as reality representation.* In Formal Ontology In Information Systems : Proceedings of the Third International Conference (FOIS-2004), pages 73–84. IOS Press, 2004. 99

[Soley 2000] R. Soley et the OMG staff. *Model driven architecture.* vol. 308, page 308, 2000. 20

[Soria 2009] C. Soria, M. Monachini et P. Vossen. *Wordnet-LMF : fleshing out a standardized format for wordnet interoperability.* In Proceeding of the 2009

international workshop on Intercultural collaboration, pages 139–146. ACM, 2009. 69

[Sowa 1984] J.F. Sowa. *Conceptual structures : information processing in mind and machine*. 1984. 57, 62

[Sroussi 2008] M. Sroussi. Mots clés des ecn. S-éditions, 2008. 134

[Stearns 2001] M.Q. Stearns, C. Price, K.A. Spackman et A.Y. Wang. *SNOMED clinical terms : overview of the development process and project status.* In Proceedings of the AMIA Symposium, page 662. American Medical Informatics Association, 2001. 36

[Sun 2004] Y. Sun. *Methods for automated concept mapping between medical databases.* Journal of biomedical informatics, vol. 37, no. 3, pages 162–178, 2004. 30

[Tao 2009] C. Tao, J. Pathak, H. Solbrig et C. Chute. *LexOWL : A bridge from LexGrid to OWL.* 2009. 77

[Tirmizi 2011] S. Tirmizi, S. Aitken, D. Moreira, C. Mungall, J. Sequeda, N. Shah et D. Miranker. *Mapping between the OBO and OWL ontology languages.* Journal of biomedical semantics, vol. 2, no. Suppl 1, page S3, 2011. 61

[Uschold 1995] M. Uschold et M. King. *Towards a methodology for building ontologies.* In Workshop on basic ontological issues in knowledge sharing, volume 80. Citeseer, 1995. 29

[Uschold 1996] M. Uschold et M. Gruninger. *Ontologies : Principles, methods and application.* The Knowledge Engineering Review, vol. 11, no. 02, pages 93–136, 1996. 29, 49

[Vandenbussche 2009] Pierre-Yves Vandenbussche et Jean Charlet. *Méta-modèle général de description de ressources terminologiques et ontologiques.* In Ingénierie de la Connaissance (IC), 2009. 95

[Vysniauskas 2006] E. Vysniauskas et L. Nemuraite. *Transforming ontology representation from OWL to relational database.* Information Technology and Control, vol. 35, no. 3A, pages 333–343, 2006. 123

[W3C 2009a] W3C. *W3C Working Draft 2 July 2009, SPARQL New Features and Rationale. Aggregates :* `http://www.w3.org/TR/sparql-features/#Aggregates`. 2009. 119

[W3C 2009b] W3C. *W3C Working Draft 2 July 2009, SPARQL New Features and Rationale. Project Expressions :* `http://www.w3.org/TR/sparql-features/#Project_expressions`. 2009. 119

[W3C 2010] W3C. *W3C Working Draft 26 January 2010, SPARQL 1.1 Property Paths : http://www.w3.org/TR/sparql11-property-paths/*. 2010. 120

[Wang 2006] H.H. Wang, N. Noy, A. Rector, M. Musen, T. Redmond, D. Rubin, S. Tu, T. Tudorache, N. Drummond, M. Horridge*et al.* *Frames and OWL side by side.* In Presentation Abstracts, page 54. Citeseer, 2006. 78

[Wittgenstein 1953] L. Wittgenstein. *Philosophical investigations, trans.* GEM Anscombe, vol. 261, 1953. 49

[Zweigenbaum 1995] P. Zweigenbaum, B. Bachimont, J. Bouaud, J. Charlet et J.F. Boisvieux. *A multi-lingual architecture for building a normalised conceptual representation from medical language.* In Proceedings of the Annual Symposium on Computer Application in Medical Care, page 357. American Medical Informatics Association, 1995. 36

[Zweigenbaum 1999] P. Zweigenbaum. *Encoder l'information médicale : des terminologies aux systèmes de représentation des connaissances.* Innovation Stratégique en Information de Santé, vol. 2, 1999. 5, 26

Annexe A

La métaphore de l'apiculture

Il existe un rapport étroit entre la modélisation et l'architecture : l'organisation. De même que nous essayons de dompter la connaissance et de l'organiser dans des SOC, un procédé analogue est appliqué dans l'organisation artificielle d'une ruche. Essayons de comprendre notre travail au travers d'une métaphore sur l'apiculture qui, je l'espère, donnera au lecteur l'irrésistible envie de mieux connaître cette activité. Cette métaphore est illustrée dans les figures A.1 et A.2.

Un essaim (colonie d'abeilles) à l'état sauvage construit des pains de cire (formés d'alvéoles de cire qui contiennent le miel). Cette configuration facilite la défense des réserves de nourriture par la colonie mais ne permet ni un accès aisé à chaque alvéole pour récolter le miel ni un grand développement de l'essaim. Nous pouvons comparer cette organisation à l'élaboration non outillée de SOC : cette solution convient pour de petits volumes où l'accès aux connaissances n'est pas un obstacle et où l'absence d'outils est une économie (financière et de formation).

Imaginons notre travail de modélisation des connaissances comme celui de la construction d'une ruche : le travail commence avec les apiculteurs (que l'on peut assimiler aux modélisateurs) ; c'est-à-dire nous. Notre objectif est de construire une ruche (serveur multi-terminologique) la mieux agencée possible pour faciliter toutes actions. La ruche est composée de cadres (SOC) sur lesquels sont fixés des plaques de cire (modèle UniMoKR) avec des alvéoles préformées. Cette architecture multi-cadres (multi-terminologiques) permet d'extraire et d'accéder plus facilement au miel (connaissances contenues dans les SOC). Notre premier travail est donc de construire des plaques de cire avec alvéoles préformés identiques (modèle UniMoKR) pour les placer dans chaque cadre (SOC). Les abeilles (responsables du SOC) vont alors étirer la cire (extension du modèle UniMoKR) de la plaque de cire pour agrandir l'alvéole et pouvoir accueillir le miel (connaissance). Les abeilles permettent grâce à leur jabot (moteur de règle) et l'enzyme invertase (règle de transformation) de transformer (transformation de modèles) les nectar et miellat (modèle source) en miel (modèle cible).

Notons que si les cadres sont bien agencés au sein de la ruche (bonne modélisa-

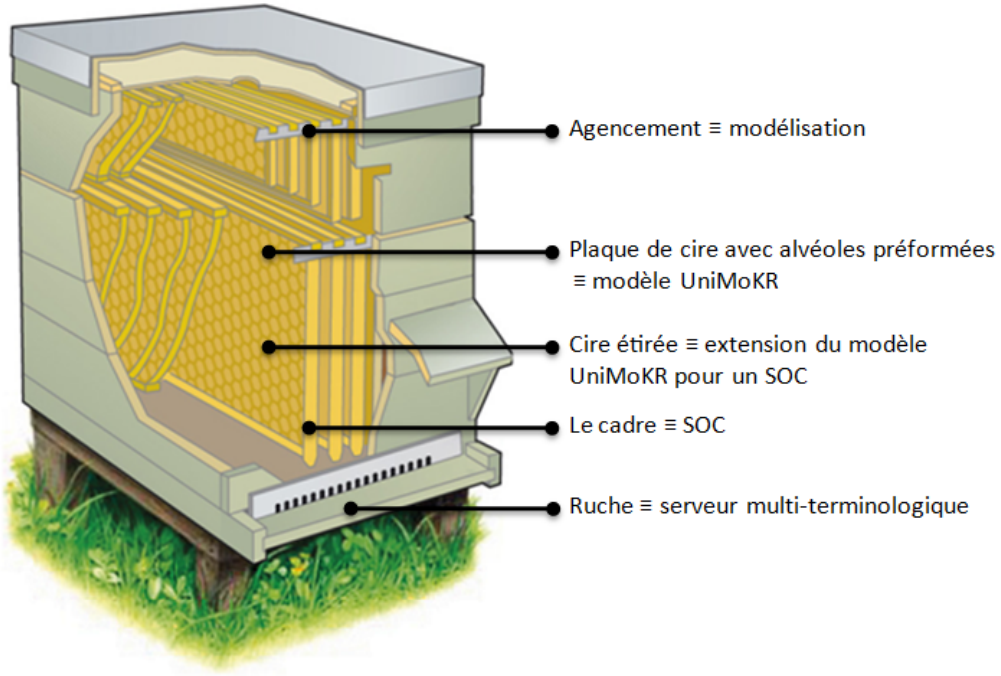

FIGURE A.1 – Métaphore de la ruche 1.

tion), la circulation des abeilles est facilitée [1] (l'accès aux connaissances). Ce choix artificiel de construction de la ruche (modélisation) permet et aide à l'ajout et au retrait d'un cadre (ajout, mise à jour, export d'un SOC).

Finalement, pour revenir à la réalité, notre travail de thèse a pour objectif de trouver une modélisation (comparable à l'architecture de la ruche) qui :

- soit adaptée pour la représentation de SOC (stockage du miel) ;
- facilite l'accès aux connaissances (accès au miel) ;
- soit souple pour permettre l'ajout, la mise à jour et le retrait de tout ou partie de SOC (manipulation aisée des cadres par l'apiculteur).

1. En acceptant la métaphore inverse et sachant que pour produire 500 grammes de miel, les abeilles doivent effectuer plus de 17 000 voyages et visiter 8 700 000 fleurs, le tout représentant plus de 7 000 heures de travail, on peut évaluer le travail que représente la construction d'un SOC !

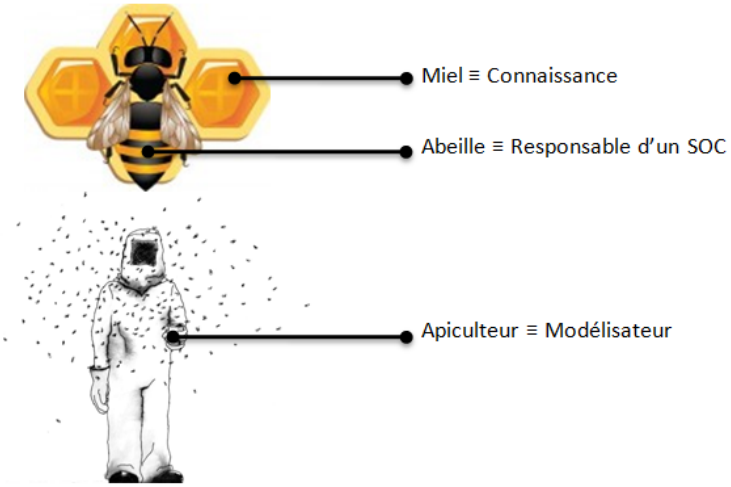

FIGURE A.2 – Métaphore de la ruche 2.

Annexe B

Les interfaces principales de l'outil ITM

Cette annexe détaille l'accès aux SOC contenus dans l'outil ITM. Ces connaissances sont séparées en espaces de travail différents selon leur niveau de description (méta, modèle et instances). Nous détaillons ces différents niveaux ainsi que les interfaces et services à disposition pour visualiser et éditer des connaissances.

B.1 Accès aux espaces de travail d'ITM

On accède aux connaissances stockées par l'outil au travers d'espaces de travail. Un espace de travail permet d'affecter des droits aux utilisateurs pour un ensemble de connaissances. La figure B.1 présente l'interface des espaces de travail du projet InterSTIS. On y distingue les espaces relevant du niveau modèle [1] (dans ce cas l'espace nommé « ontologie ») et ceux relevant du niveau instances (dans ce cas, l'espace nommé « terminologies »). Les interfaces au sein de chaque espace sont génériques et construites dynamiquement en fonction des connaissances des niveaux courants et supérieurs (par exemple les interfaces du niveau modèle vont être construites à partir des connaissances du niveau modèle et du niveau méta).

B.2 Le niveau modèle

Le niveau modèle dispose des primitives du méta-modèle d'ITM pour construire le modèle. Ces éléments visibles en figure B.2 permettent de représenter l'ontologie de domaine en utilisant des classes, attributs, relations, etc. définis au niveau méta.

Dans le projet InterSTIS, la construction de l'ontologie de domaine est passée par la définition d'une classe *Concept CIM10* qui est sous-classe de la classe *Concept* et appartient à notre modèle UniMoKR. Cette classe *Concept CIM10* hérite des propriétés et contraintes appliquées à la classe *Concept*. Comme nous l'avons montré

1. L'architecture d'ITM est partagée entre les niveaux méta, modèle et instances (*cf.* section 7.2.1).

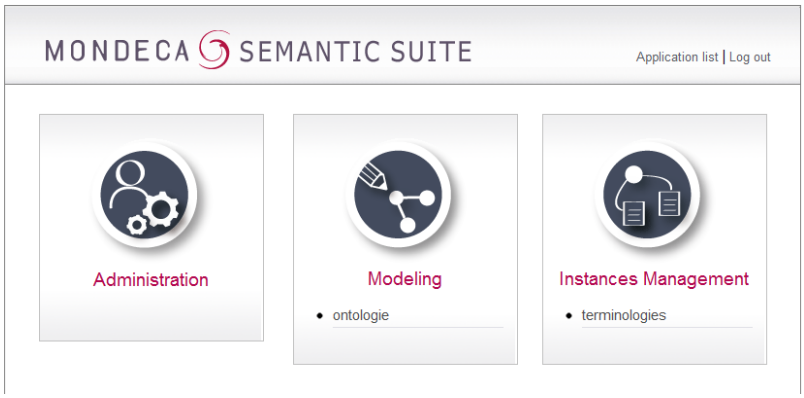

FIGURE B.1 – ITM – Interface des espaces de travail du projet InterSTIS. Seuls les espaces appartenant au niveau modèle (nommé dans cette figure « Modeling ») et au niveau instances sont présents dans cet écran. Le niveau méta n'est disponible qu'au super utilisateur. L'espaces « ontologie » contient le modèle UniMoKR et ses extension pour la prise en compte des SOC du projet. L'espace « terminologies » contient le contenu des SOC du projet.

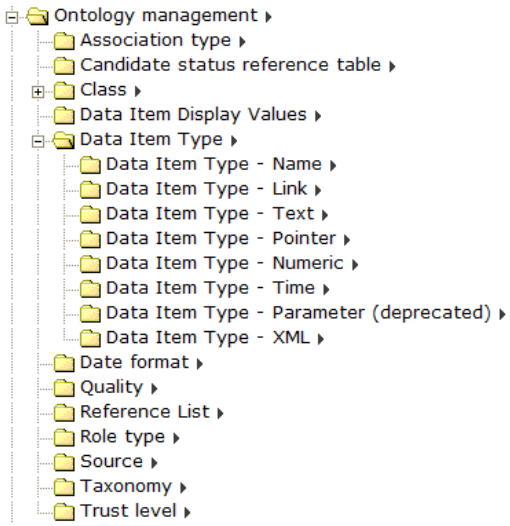

FIGURE B.2 – ITM – Interface des classes du niveau méta qui permettent la construction du niveau modèle. Les éléments principaux sont : *Class*, *Association type* et *Data Item Type* (équivalent à *Property* en RDF).

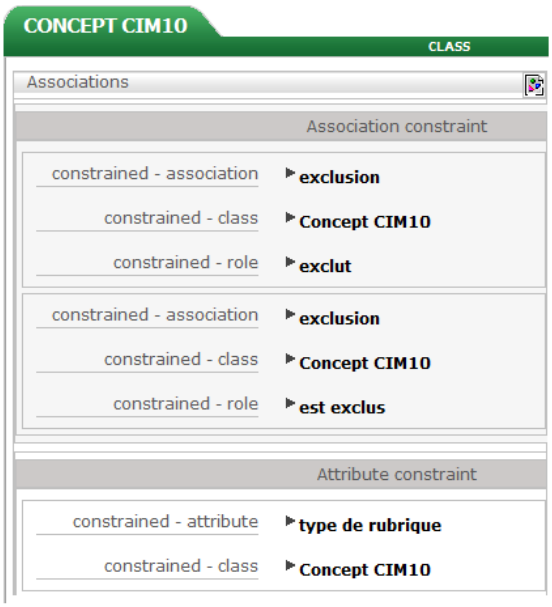

FIGURE B.3 – ITM – Interface de la classe *Concept CIM10* appartenant au modèle de la classification CIM-10. Cette classe est une extension du modèle UniMoKR et permet de représenter les spécificités de cette SOC comme l'attribut *type de rubrique*.

en section 6.4, la représentation de la classification CIM-10 nécessite l'extension de notre modèle commun UniMoKR pour prendre en compte les spécificités de ce SOC. Ainsi la classe *Concept CIM10* est le domaine de l'attribut *type de rubrique* ou encore les domaine et co-domaine de la relation *exclusion* (*cf.* figure B.3).

B.3 Le niveau instances

De même qu'au niveau modèle, le niveau instances dispose des primitives définies au niveau modèle pour construire des connaissances (*cf.* figure B.4). Chaque *Group* de notre modèle UniMoKR est accessible et permet d'accéder à une vue arborescente de leur contenu(si une relation de navigation *structuringAssociationType* est spécifiée (*cf.* figure B.5).

Il est possible d'accéder aux concepts d'un SOC au moyen d'une recherche textuelle, de requêtes de type graphe ou encore par navigation. La visualisation du concept *conjonctivite aiguë, sans précision* appartenant à la CIM-10 est présentée en figure B.6. Ce concept possède des attributs définis dans le modèle UniMoKR comme *notation* mais aussi des attributs spécifiques à la CIM-10 comme *type de rubrique*.

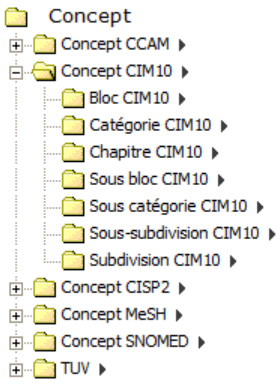

FIGURE B.4 – ITM – Interface des classes du niveau modèle du projet InterSTIS qui permettent la construction du niveau instances. On y retrouve la hiérarchie des classes de concept CIM-10.

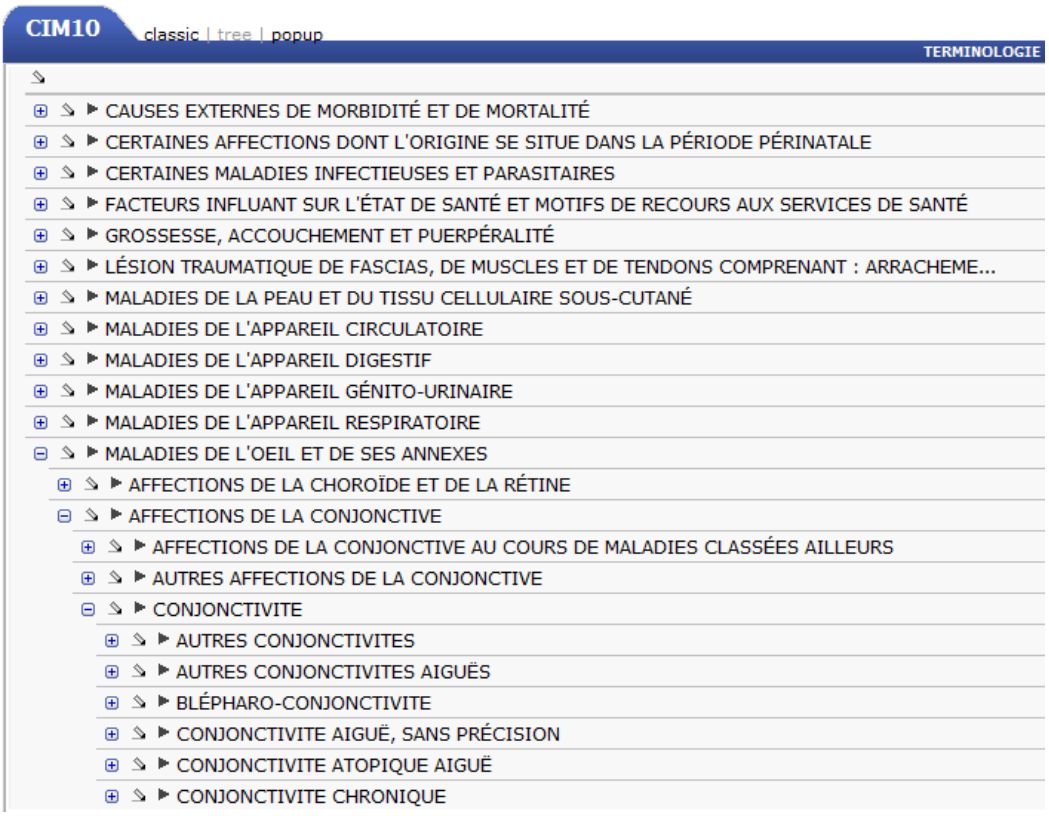

FIGURE B.5 – ITM – Interface de la vue hiérarchique de la terminologie CIM10.

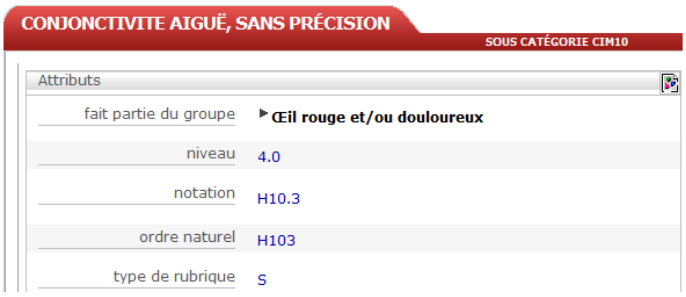

FIGURE B.6 – ITM – Interface de la visualisation du concept *Conjonctivite aiguë, sans précision* appartenant à la CIM-10. Cette figure présente l'ensemble des attributs de ce concept. L'attribut *notation* décrit dans le modèle UniMoKR a pour ce concept, la valeur « H10.3 ».

FIGURE B.7 – ITM – Interface d'édition du concept *Conjonctivite aiguë, sans précision* appartenant à la CIM-10. Les contraintes exprimés sur le modèle sont, à ce niveau instance, interprétés pour générer l'écran d'édition et contraindre la saisie par un utilisateur.

L'outil ITM permet l'édition des connaissances. Les interfaces d'édition sont générées et contrôlées par les contraintes exprimées au niveau modèle. Une illustration est donnée en figure B.7 et montre les attributs modifiables pour le concept *conjonctivite aiguë, sans précision*. On remarque que les restrictions de cardinalité définies dans l'ontologie de domaine contraignent la saisie (ici l'attribut *Name* est obligatoire : une valeur ou plus). Le type des attributs au niveau modèle influe sur l'apparence et le type de données que l'on peut saisir. De même les contraintes de l'ontologie de services permettent d'indiquer par exemple que l'attribut *abréviation* possède une langue.

Résumé : Ce travail de thèse, réalisé au sein de l'entreprise MONDECA et du laboratoire de recherche INSERM, est né du besoin de disposer d'un serveur capable de supporter le processus éditorial de Systèmes d'Organisation de Connaissances (SOC) et soulève la problématique suivante : comment harmoniser la représentation des SOC et de leurs correspondances afin de proposer des services unifiés qui supportent l'édition, la publication et l'utilisation efficaces des connaissances de ces référentiels ?

Pour répondre à cette problématique, nous soutenons la thèse que l'élaboration d'un modèle de représentation commune de SOC est une solution adaptée pour (i) pallier l'hétérogénéité de ces référentiels, (ii) favoriser l'interopérabilité sémantique au sein d'un Système d'Information et (iii) proposer des services unifiés quel que soit le SOC. Nous utilisons pour cela des méthodes propres à l'Ingénierie des Connaissances couplées à celles de l'Ingénierie des modèles. Les contributions présentées se concentrent sur trois axes.

Dans un premier axe, nous souhaitons obtenir une solution de modélisation de SOC la plus générique possible et qui puisse être étendue pour prendre en compte les spécificités de chacun des référentiels. Nous proposons donc un modèle extensible commun de représentation, nommé UniMoKR, construit à partir des standards, recommandations et projets existants. Notre modèle a été proposé et intégré en partie dans la future norme ISO 25964 qui porte sur la représentation des terminologies. Nous avons également soumis deux patrons de modélisation d'ontologie au portail Ontology Design Pattern.

Le second axe est consacré à la proposition de services unifiés qui reposent sur cette modélisation. Parmi ces services nous distinguons l'export de tout ou partie de SOC dans un format standard d'échange ou encore des services Web de gestion de terminologies. Pour mettre ces services à disposition, nous préconisons la méthode de transformation de modèles qui utilise le langage SPARQL pour l'expression des règles de transformation.

Dans un troisième axe, nous présentons l'application de notre solution testée et commercialisée pour divers projets dans différents domaines d'applications. Nous montrons ici la faisabilité de notre approche, ainsi que l'amélioration que la représentation formelle de notre modèle apporte à la qualité des informations. Ces implémentations ont permis d'effectuer une validation en condition d'utilisation.

Mots clés : Systèmes d'Organisation de la Connaissance, modélisation, MDA, OWL, serveur multi-terminologique