Conception d'un modèle Web sémantique appliqué à la génomique fonctionnelle

Fleur Mougin

N^o d'ordre : 100

THÈSE

présentée

DEVANT L'UNIVERSITÉ DE RENNES I

pour obtenir

le grade de : ***DOCTEUR DE L'UNIVERSITÉ DE RENNES I***

Mention : Génie Biologique et Médical

PAR

Fleur Mougin

Laboratoire : EA 3888
« Modélisation des connaissances biomédicales »
(Faculté de Médecine, Rennes)

École doctorale : Vie, Agro, Santé (VAS)

Titre de la thèse :

Conception d'un modèle Web sémantique appliqué à la génomique fonctionnelle

Soutenue le 1er décembre 2006, devant la commission d'examen

COMPOSITION DU JURY :

M.	Dominique	LAVENIER	Président
M.	Pierre	LE BEUX	Directeurs de thèse
M.	Olivier	LORÉAL	
M.	Philippe	BESSIÈRES	Rapporteurs
M^{me}	Christine	FROIDEVAUX	
M^{me}	Anita	BURGUN	

Remerciements

Les travaux présentés dans cette thèse ont été réalisés au sein de l'Équipe d'Accueil 3888 ou Modélisation Conceptuelle des Connaissances Biomédicales, où j'ai commencé par faire des stages. C'est là que j'ai réellement découvert la recherche et l'intérêt qu'elle a éveillé en moi, malgré ce que j'avais pu dire jusqu'ici. J'y ai ainsi fait mes premiers pas en recherche et j'espère pouvoir y prolonger cette expérience particulièrement instructive et enrichissante.

Je remercie tout d'abord le directeur de cette thèse, Pierre Le Beux, pour m'avoir accueillie dans son laboratoire et pour avoir accepté de diriger mes travaux durant ces quatre années.

Mes remerciements vont également à Olivier Loréal de l'INSERM U522 qui a co-dirigé cette thèse. Même si la compréhension entre les mondes biologique et bioinformatique a parfois été délicate, j'ai vraiment beaucoup apprécié de discuter et d'échanger avec vous. Vous m'avez toujours donné des conseils, remarques et commentaires constructifs qui m'ont permis d'avancer petit à petit.

Je souhaite exprimer ma gratitude tout particulièrement à Anita Burgun sans qui ce travail n'aurait pas été ce qu'il est. Merci de m'avoir guidée, conseillée et encouragée tout au long de ma thèse. La confiance dont tu as fait preuve envers moi dès le début m'a beaucoup motivée.

Christine Froidevaux et Philippe Bessières m'ont fait l'honneur d'être les rapporteurs de cette thèse, et je les en remercie, de même que pour leur participation au Jury. Merci de l'intérêt que vous avez porté à mon travail et de vos remarques qui ont contribué à améliorer la qualité de ce manuscrit.

Je remercie également Dominique Lavenier d'avoir accepté de participer au Jury de soutenance.

Je tiens à remercier Olivier Bodenreider, chercheur à la National Library of Medicine, pour les conseils stimulants et les suggestions enrichissantes que j'ai eu l'honneur de recevoir de sa part. Nos collaborations et échanges m'ont beaucoup apporté, aussi bien scientifiquement que linguistiquement. Merci aussi de m'avoir accueillie au sein de votre institution.

Merci à Asuncion Gomez Perez qui m'a permis d'effectuer un stage de six mois au sein de son équipe Ontology Engineering Group du Laboratoire d'Intelligence Artificielle à Madrid. Je

remercie également Angel, Boris, Edwin, Mari Carmen, Oscar et Raul pour l'hospitalité dont vous avez fait preuve envers moi lors de ce séjour et pour m'avoir donné un aperçu non seulement de ce qu'est la recherche en Espagne mais aussi de la convivialité qui y règne.

Je tiens à remercier tous ceux du labo qui m'ont soutenue au quotidien : Gwenn, Julie, Marc, Nicolas et Olivier aussi bien par les discussions que j'ai eu la chance d'avoir avec eux, leurs suggestions ou contributions. Je pense ici en particulier à Olivier qui m'a donné de nombreux conseils instructifs et surtout à Julie qui, malgré les nombreuses obligations qui l'attendaient, a passé un temps précieux pour relire avec beaucoup d'attention mon manuscrit. Merci encore pour ton aide, sans tes corrections très détaillées et constructives, ce travail n'aurait pas été le même. C'est un réel plaisir de travailler avec vous tous et j'espère que ce n'est qu'un début !

J'exprime toute ma profonde et sincère reconnaissance à ma famille qui m'a toujours épaulée et encouragée. Merci à vous d'avoir supporté mes humeurs et de m'avoir fait confiance à tout moment.

Merci également à mes amis d'avoir été là pour m'écouter ou me changer les idées quand cela était nécessaire ou bien simplement pour partager un bon moment tous ensemble. La liste n'est pas exhaustive mais je citerais au moins : Agnès, Amandine, Cécile, Céline, Dedel, Franck, François, Gene, Greg, Jay, Jona, Mathilde, Matthias, Matthieu, Maya, Rémi, Sandrine, Seb God et Soph.

Un p'tit merci spécial à Nathalie Hernandez qui me comprend si bien et avec qui j'apprécie tant de partager mes impressions vis à vis de la recherche. Je ne désespère pas d'écrire un article avec toi un jour. Si notre collaboration professionnelle est aussi efficace qu'en Espagne et aussi constructive que notre symbiose amicale, ça pourrait bien faire des étincelles !

Cette liste serait bien entendu incomplète sans toi, Richard. Merci pour ton attention, ton écoute, ta patience et ton soutien de chaque instant. T'avoir à mes côtés est extrêmement réconfortant et rassurant, alors que ça ne devait pas être évident de re-signer pour tout ça. En résumé, merci d'être là, tout simplement...

À mon grand père,
À Sandrine.

Table des matières

Chapitre 1

Introduction

La **génomique fonctionnelle***[1] se définit par l'« étude et l'analyse directe du transcriptome* et du protéome* : elle vise à déterminer la fonction des gènes à partir de leurs produits d'expression (ARN* et protéines), ainsi qu'à étudier leur mode de régulation et leurs interactions »[2]. Elle opère en parallèle sur plusieurs centaines ou milliers de séquences d'ADN* et de protéines fournies par les projets de séquençage. En plus de leur impact déterminant dans le domaine biologique, les différentes approches méthodologiques et technologiques de génomique fonctionnelle sont utiles au domaine de la santé, pour le diagnostique et le traitement de certaines maladies et cancers.

Étant donné les grands volumes d'informations manipulés dans ce cadre, il est apparu indispensable de développer des sources capables de stocker, de rendre accessibles et de gérer ces données massives pour pouvoir ensuite les exploiter. Les biologistes, mais aussi les médecins, ont besoin de connaître et de disposer du maximum d'informations sur ces données dans le cadre de leurs travaux de recherche. Par exemple, à partir d'un gène impliqué dans une pathologie, les biologistes et médecins ont besoin de disposer d'informations à propos de cette maladie (ses manifestations, d'autres gènes potentiellement impliqués, etc) et à propos du gène lui-même (sa séquence, son polymorphisme, les voies métaboliques dans lesquelles il intervient, etc). En effet, l'interprétation des données expérimentales nécessite généralement de comparer des données cliniques et biologiques avec des ensembles de données déjà existantes, mais aussi avec des bases de connaissances de références.

La recherche dans le domaine de la génomique fonctionnelle nécessite donc d'accéder à une multitude de **sources de données** de différents types : entrepôts, bases et banques de données, bases de connaissances, ressources terminologiques, ontologies, etc. D'après le dernier état des lieux effectué par le journal Nucleic Acids Research, 858 sources de biologie moléculaire ont été recensées au 1er janvier 2006 [Galperin 06]. Celles-ci sont **distribuées** sur divers serveurs et évoluent très rapidement, indépendamment les unes des autres. Aujourd'hui, le problème qui est donc posé aux biologistes et médecins est celui d'une **collecte manuelle d'informations, qui ne peut être que partielle, très fastidieuse, voire même erronée**, étant donné le nombre de sources disponibles. À ce problème de distribution s'ajoute celui de la diversité des sources

[1] Les mots suivis d'un astérisque sont définis dans le glossaire

[2] Source InfoBioGen - http ://www.infobiogen.fr/glossaire/glossaire.php ?lettre=G#GENOMIQUE-FN

biologiques et médicales, ou biomédicales. En effet, celles-ci sont **hétérogènes à de nombreux niveaux** [Bry 03], [Froidevaux 02] :

- **Contenu**

  Ces différentes sources touchent à divers disciplines du domaine biomédical. Leur contenu diffère donc d'une source à l'autre. Par exemple, la source GenBank regroupe des données *biologiques* concernant les séquences nucléiques et protéiques [Benson 06], alors que OMIM (Online Mendelian Inheritance in Man) traite de données *médicales* en cataloguant des maladies génétiques [Hamosh 05].

  D'autre part, des sources traitant de la même entité biomédicale **ne fournissent pas forcément des données qui se situent au même niveau**. Entrez Gene, par exemple, contient des informations *généralistes* au sujet des gènes [Maglott 05] tandis que KEGG fournit *plus spécifiquement* des informations sur les voies métaboliques et réseaux de régulation et en conséquence les gènes qui y sont impliqués [Kanehisa 06]. Par ailleurs, certaines sources contiennent des informations *spécifiques à un organisme*, comme MGD (Mouse Genome Database) [Blake 06] qui concerne la souris en particulier, alors que d'autres sont *générales*, c'est le cas notamment de Swiss-Prot [Boeckmann 03], base de connaissances universelle de protéines.

- **Syntaxe**

  Au niveau syntaxique, on distingue également des différences entre les sources. En effet, celles-ci sont **implémentées de manières diverses** ; certaines sous forme de base de données relationnelles (par exemple, Entrez Gene) ou plus simplement par des fichiers plats (par exemple, Swiss-Prot). D'autre part, les interfaces Web des sources **n'utilisent pas le même format pour décrire leurs données**. Par exemple, DDBJ (DNA Data Bank of Japan) [Okubo 06] fournit des résultats de requêtes au format XML*[3] (eXtensible Markup Language - ou langage de balisage extensible) [Bray 00] tandis que Swiss-Prot se limite au format texte.

- **Sémantique**

  L'hétérogénéité sémantique entre les sources est la plus complexe à résoudre. En fonction des sources, on rencontre des **conflits sémantiques** dûs à la diversité des modes de désignation des entités du domaine et à l'interprétation de certaines notions fondamentales qui sont propres à chacun. L'exemple traditionnellement utilisé pour illustrer cet aspect est la définition même d'un gène [Schulze-Kremer 02] : dans la source GDB (Genome Data Base) [Letovsky 98], il est défini comme un « fragment d'ADN qui peut être transcrit et traduit en une protéine » alors que dans GenBank, il est considéré comme un « fragment d'ADN qui porte un trait génétique ou un phénotype (incluant des régions codantes non structurelles d'ADN, comme les introns ou les promoteurs) ». Cela pose des problèmes dans la mesure où un même terme est utilisé pour décrire deux notions différentes. Un autre exemple est l'unité de mesure de la distance génétique (kilobases ou centimorgans) qui n'est pas toujours la même, il est donc nécessaire de faire des conversions pour pouvoir comparer ce type de données.

  Il existe aussi des **différences de granularité dans la représentation des données** se trouvant dans les sources. Un exemple concerne deux sources de contenu identique :

[3] http ://www.w3.org/XML/

```
<OrgName>
  <OrgName_name>
   <OrgName_name_binomial>
    <BinomialOrgName>
      <BinomialOrgName_genus>Homo</BinomialOrgName_genus>
      <BinomialOrgName_species>sapiens</BinomialOrgName_species>
    </BinomialOrgName>
   </OrgName_name_binomial>
  </OrgName_name>
  <OrgName_lineage>Eukaryota; Metazoa; Chordata; Craniata;
Vertebrata; Euteleostomi;
 Mammalia; Eutheria; Primates; Catarrhini; Hominidae;
Homo</OrgName_lineage>
   <OrgName_gcode>1</OrgName_gcode>
   <OrgName_mgcode>2</OrgName_mgcode>
   <OrgName_div>PRI</OrgName_div>
</OrgName>
```

```
<ORGANISM>Homo sapiens</ORGANISM>
<TAXONOMY>Eukaryota; Metazoa; Chordata; Craniata; Vertebrata;
Euteleostomi;Mammalia;
Eutheria; Primates; Catarrhini; Hominidae; Homo.</TAXONOMY>
```

FIG. 1.1 – **Extraits de fichiers résultats au format XML des sources GenBank et DDBJ.** L'extrait de GenBank apparaît avec un fond blanc et celui de DDBJ avec un fond gris : bien que contenant les mêmes informations, leur représentation de l'organisme Homo Sapiens est différente. DDBJ utilise un niveau de granularité plus fin que Genbank.

GenBank et DDBJ qui fournissent des données sur les séquences nucléiques et protéiques au format XML. Mais ce dernier n'a pas la même forme dans les deux sources, reflétant ainsi deux modes de représentation différents pour des informations totalement similaires. En effet, leur représentation en XML de l'*Homo Sapiens* est un organisme décrit en genre et espèce dans GenBank et simplement un organisme dans DDBJ (Figure 1.1).

Les technologies du Web fournissent une réponse au problème de distribution. Cependant, si elles rendent possible l'accès aux différentes sources de données, et en ce sens c'est un succès, cet accès est encore manuel et la mise en correspondance des informations présentes dans les différentes sources requiert l'intervention humaine. Automatiser ces accès et combiner les résultats nécessitent de résoudre les différents niveaux d'hétérogénéité identifiés précédemment. Ces besoins ne sont cependant pas spécifiques à la génomique fonctionnelle et font actuellement l'objet de recherches et développements technologiques dont pourrait profiter notre domaine. C'est dans ce cadre notamment que la notion de **Web sémantique** a été introduite [Berners-Lee 01]. L'objectif général est de rendre le contenu des pages Web interprétables par les machines, et plus uniquement par les hommes. Ainsi, des outils logiciels pourront réaliser des tâches compliquées, et ce de manière automatique, afin de faciliter le travail des utilisateurs nécessitant d'accéder à des informations présentes sur Internet. Plus spécifiquement, un des intérêts majeurs du Web sémantique en génomique fonctionnelle est d'apporter suffisamment de renseignements sur les ressources, de décrire leur contenu de manière à la fois formelle et signifiante, de telle sorte que des programmes de recherche sur le Web puissent sélectionner de manière automatique les informations pertinentes pour une question donnée et les combiner. On entend par ressource* n'importe quelle entité informatique (document électronique, image, service, collection d'autres ressources, etc) ayant une identité[4].

[4]http ://www.gbiv.com/protocols/uri/rfc/rfc2396.html

L'un des axes fondamentaux du Web sémantique est l'intégration automatique d'informations provenant de sources hétérogènes [Laublet 02]. L'objectif est de faciliter les tâches de recherche et de collecte de données réalisées par les utilisateurs au cours de leurs travaux en mutualisant les informations existant dans les sources de données pertinentes. De plus, la combinaison et une représentation plus formelle des données issues de sources hétérogènes devraient permettre de découvrir de nouvelles connaissances à partir de l'existant. Ainsi, l'idée est de créer un système d'intégration donnant l'illusion d'interroger un système homogène, global et centralisé, plutôt qu'une multitude de sources de données distribuées.

Ce travail de thèse vise à aider les biologistes et médecins à accéder aux informations disponibles dans les multiples sources de données en génomique fonctionnelle. Pour cela, nous proposons une approche pour la conception d'un modèle Web sémantique, et plus précisément d'un système d'intégration de sources de données biomédicales.

Le manuscrit est organisé comme suit :

- la section suivante positionne le sujet dans son contexte, présente les approches d'intégration existantes pour la réalisation de systèmes, introduit certains de ces différents systèmes, et aborde finalement la problématique de mise en correspondance de schémas qui est nécessaire lors de la conception de tels systèmes ;
- la section 3 annonce les objectifs de notre travail ;
- la section 4 présente le matériel utilisé et les méthodes mises en œuvre pour la conception de notre système d'intégration ;
- la section 5 donne les résultats obtenus grâce aux méthodes détaillées à la section précédente ;
- la section 6 présente le système que nous avons développé au travers d'exemples de requêtes et aborde les questions de son évolution ;
- la section 7 positionne notre travail par rapport aux travaux existants en recensant les apports, les limites et les perspectives de ce travail ;
- enfin, la dernière section conclut ce travail.

Chapitre 2

État de l'art

Au cours de leurs travaux, les biologistes et médecins doivent rechercher des informations dans des sources de données qui sont réparties sur Internet et hétérogènes à de multiples niveaux. Dans ce cadre, nous présentons l'intérêt d'utiliser les technologies du Web sémantique. Nous détaillons ensuite les différentes approches d'intégration qui existent pour fournir aux biologistes et médecins un accès unique aux informations situées dans des sources de données biomédicales distinctes. Enfin, nous introduisons la notion de mises en correspondance, nécessaires lors de la conception des systèmes d'intégration, et les différentes méthodes existantes pour les découvrir.

2.1 Contexte

2.1.1 Le domaine biomédical

2.1.1.1 Scénario d'interrogation de sources de données par les biologistes et médecins

En génomique fonctionnelle, les biologistes comme les médecins nécessitent d'accéder à des informations disponibles sur Internet. Ainsi, lors de l'analyse des résultats de puces à ADN, les biologistes interrogent de nombreuses sources de manière à récupérer l'ensemble des données dont ils ont besoin pour interpréter les résultats obtenus au sujet des gènes étudiés par cette technologie. Ces informations se trouvent dans des sources de données biomédicales qui sont à la fois hétérogènes, réparties, autonomes et potentiellement redondantes. En effet, il y a un recouvrement certain entre les informations existant dans les différentes sources. La difficulté du travail des biologistes pour rechercher ces informations liées à leurs résultats expérimentaux se situe donc à différents niveaux : l'**identification** des sources (ainsi que leur URL*) contenant l'information nécessaire, la **manière d'interroger** ces sources, la **navigation** entre elles, la **collecte** des données qui les intéressent et enfin la **fusion** et le **nettoyage** des informations obtenues.

Un exemple de requête est l'identification des maladies dans lesquelles sont impliqués des gènes explorés lors des approches transcriptomiques utilisant les puces à ADN ainsi que les interactions des protéines qui leur sont associées (Figure 2.1 page suivante). Pour cela, le biologiste va interroger des sources qu'il aura préalablement identifiées comme utiles et pertinentes vis à vis de sa recherche. Il devra ensuite naviguer au sein des sources interrogées de manière à collecter les informations qui l'intéressent. Plus précisément, cela consiste à récupérer dans certaines d'entre elles les maladies, les symptômes ou les syndromes dans lesquels les gènes étudiés sont impliqués (sources 1 et 5) et parallèlement à identifier les protéines associées aux produits de gènes étudiés dans une source 2 pour les fournir en entrée d'une autre source 3. Il pourra ainsi récupérer les interactions de protéines dans la source 3 mais aussi dans la source 4 directement au moyen des gènes déposés. Une fois cette collecte effectuée, le biologiste devra fusionner et nettoyer les résultats obtenus notamment dans les sources 1 et 5 en ce qui concerne les maladies ainsi que ceux fournis par les sources 3 et 4 au sujet des interactions. Cette unification est nécessaire car elle permet d'homogénéiser les données récupérées dans des sources différentes et d'y éliminer des éventuelles redondances.

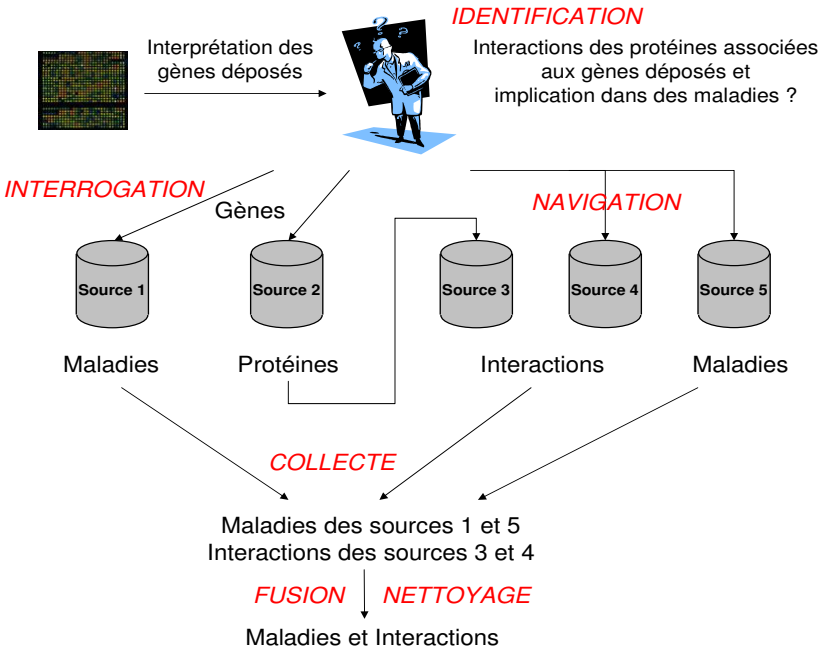


FIG. 2.1 – **Scénario d'interrogation de sources de données pour identifier les maladies dans lesquelles sont impliqués des gènes déposés sur une puce à ADN ainsi que les interactions des protéines qui leur sont associées.**

2.1.1.2 Caractéristiques des sources de données biomédicales

Les **caractéristiques** principales des sources de données biomédicales sont les suivantes [Hernandez 04] :

- les sources contiennent des **données de nature diverse**, dû au fait que le domaine biomédical interagit avec de nombreux sous-domaines ou domaines connexes, tels que l'anatomie, la pharmacologie ou encore la chimie. De plus, **les données que les sources hébergent diffèrent par leur format**. Par exemple, certaines stockent des données simples, comme GenBank qui fournit des informations textuelles sur les séquences alors que d'autres fournissent des données sous des formes plus complexes, telles que PDB qui contient des images décrivant la structure des protéines [Deshpande 05]. Une conséquence de la différence des formats est l'espace de stockage nécessaire qui est largement moindre dans le premier cas que dans le second ;
- les données sont **hétérogènes dans leur représentation**. Cela englobe des différences dans leur contenu, leur syntaxe et leur sémantique. Cet aspect, déjà abordé dans l'introduction, soulève différents types de problèmes comme la redondance d'informations d'une source à l'autre, et résultante de cela, la cohérence et la compatibilité de ces informations. En l'occurrence, si des données censées être semblables sont divergentes dans deux sources distinctes, on doit pouvoir identifier une incohérence dans au moins l'une des sources ;
- les sources étant **autonomes**, elles sont susceptibles de modifier leur contenu ou leur schéma à tout moment, voire même de supprimer des données sans en faire état. Par

exemple, quand une nouvelle publication apparaît sur un gène donné, il peut être nécessaire de compléter ou modifier les informations sur ce gène dans les sources qui contiennent au moins une entrée le concernant ;

- enfin, les **capacités d'interrogation offertes par les interfaces sont différentes** suivant les sources, ce qui complique la tâche de requêtes des biologistes et médecins. Ces derniers, qui ne sont pas nécessairement familiers avec chaque site Web hébergeant les sources, risquent de ne pas exploiter au mieux les fonctionnalités d'interrogation offertes et donc de passer à côté d'informations potentiellement pertinentes pour eux.

Ces différents points illustrent les difficultés auxquelles sont confrontés les biologistes et médecins quand ils cherchent à récupérer des informations dans les sources de données biomédicales accessibles sur Internet. L'intégration de ces sources doit donc permettre d'aider les biologistes et médecins à effectuer ces tâches et ainsi d'inférer de nouvelles connaissances.

2.1.1.3 Recherche d'informations des biologistes et médecins

Le travail de recherche d'informations et de mise en concurrence de résultats avec l'existant apparaît clairement comme une tâche très pénible, fastidieuse et source d'erreurs pour les biologistes et médecins. Il est ainsi nécessaire de faciliter ce travail en l'automatisant au maximum. Les différents points étant de pouvoir :

- guider les biologistes et médecins pour constituer leurs requêtes vis à vis des sources sans qu'ils aient besoin de se soucier de la manière dont elles sont implémentées et représentées ;
- proposer des chemins possibles entre les différentes sources au travers de liens (hypertextes en particulier) existant entre elles. Ces liens sont nommés références croisées (*Cross-references* en anglais) dans la littérature ;
- récupérer les informations pertinentes qui intéressent les biologistes et médecins (vis à vis de leur requête) ;
- agréger des portions d'informations qui se retrouvent réparties dans plusieurs sources ;
- analyser les résultats obtenus dans les différentes sources et les trier de manière à rendre aux biologistes et médecins une réponse à leur requête qui soit globale, homogène et non redondante. Plus précisément, ils devront pouvoir interroger de manière centralisée diverses sources et obtenir un résultat unique mutualisant les informations fournies par les sources sans reproduire leurs hétérogénéités (contenu, syntaxe et sémantique - cf 1 page 12).

Dans ce cadre, les technologies émergentes du Web sémantique, visant à rendre les informations disponibles sur Internet interprétables non seulement par les hommes mais également par les machines, semblent pouvoir répondre à ces attentes ou au moins contribuer partiellement à leur résolution. Les données présentes dans les sources doivent pouvoir être manipulées par des programmes de manière automatique afin de faciliter le travail de recherche d'informations des biologistes et médecins. Traiter cette tâche automatiquement plutôt que manuellement améliorerait les résultats, notamment en terme d'exhaustivité des données récoltées pouvant répondre à une requête dans un ensemble de sources. En effet, étant donné le nombre très important de sources biomédicales et le fait qu'il augmente chaque année un peu plus [Galperin 06], il est quasiment impossible de connaître l'ensemble des informations disponibles dans les sources. En plus de cela, les sources existantes évoluent très rapidement et il est très difficile pour les biologistes

et médecins d'avoir des connaissances constamment à jour même dans le cas des sources qu'ils utilisent fréquemment. Nous considérons les technologies du Web sémantique pour résoudre un certain nombre des problèmes identifiés lors de la collecte d'informations sur Internet dans le domaine biomédical.

2.1.2 Le Web sémantique

Selon Tim Berners-Lee, le Web sémantique est « une extension du Web actuel où l'information prend un sens bien précis permettant aux hommes et aux machines de travailler en coopération d'une meilleure façon » [Berners-Lee 01]. La plupart des informations se trouvant sur Internet sont effectivement lisibles par les hommes mais rarement (ou mal) interprétables par les machines. En effet, avec le moteur de recherche Google[1], par exemple, si on effectue une recherche sur le mot `Virus` sans spécifier explicitement que l'on est intéressé par les virus dans le domaine médical, parmi les 10 premiers résultats obtenus, seul le quatrième répond à notre attente (Figure 2.2 page suivante). De plus, les résultats retournés par Google sont généralement très nombreux, ici on obtient un total de 531 millions de pages Web, ce qui a logiquement pour effet de noyer les utilisateurs sous une masse d'informations beaucoup trop importante. Une solution pour obtenir des résultats plus pertinents et restreints est d'affiner sa requête en ajoutant le mot « médecine », mais on obtiendra malgré tout du bruit puisque des pages Web contenant notamment la phrase « on s'intéresse aux virus informatiques et non à ceux observés en médecine » feront partie des résultats proposés par Google.

Il est donc apparu nécessaire de représenter plus formellement le contenu des ressources du Web actuel au moyen de langages visant à ajouter de la sémantique à la description syntaxique des informations disponibles sur Internet. Cette dernière sera ainsi exploitable par les machines qui vont pouvoir automatiser le traitement de connaissances formalisées, et plus précisément le raisonnement sur celles-ci. En particulier, cet enrichissement a pour but de libérer les utilisateurs d'une grande partie de leur pénible travail de recherche d'informations et d'exploiter de grands volumes d'informations grâce à des systèmes gérant les connaissances d'un domaine.

Différents axes peuvent être dégagés dans le Web sémantique ainsi que des technologies fondamentales constituant celui-ci. Nous présentons les composants suivants : les langages, les méta-données et les ontologies. Nous verrons en quoi ceux-ci vont permettre de répondre en partie aux problèmes posés par les différents types d'hétérogénéité identifiés précédemment (cf 1 page 12).

2.1.2.1 Les langages

Une hiérarchie de langages constitue le pilier du Web sémantique. Ce sont en effet ces langages qui vont permettre de décrire le contenu des ressources Web, le rendant ainsi lisible et exploitable par les machines. Le modèle en couches (*Layer cake*) (Figure 2.3 page 21), proposé par Tim Berners-Lee et le « World Wide Web Consortium[2] » (W3C) plus généralement, illustre la structure des niveaux où se situent les différentes technologies du Web sémantique. Chaque niveau s'appuie sur les fonctionnalités de ceux qui sont en dessous de lui-même. Le W3C s'attache

[1] http ://www.google.fr
[2] http ://www.w3.org/

FIG. 2.2 – **Résultat d'une requête effectuée sur le mot Virus dans le moteur de recherche Google.**

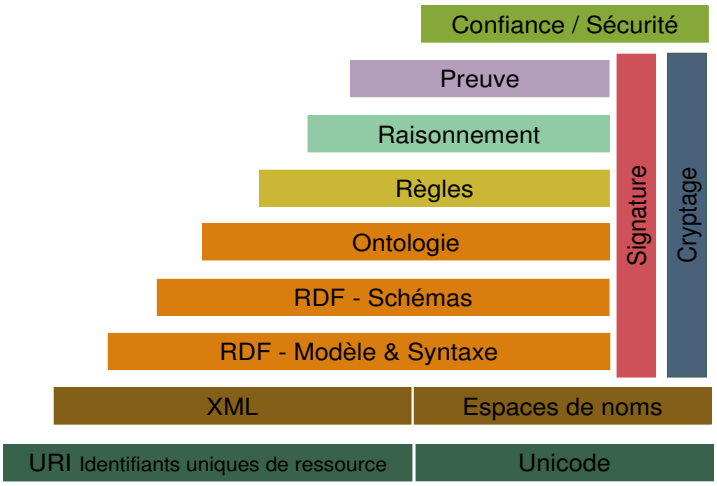


FIG. 2.3 – **Les technologies du Web sémantique organisées par niveaux - Vision de Tim Berners-Lee et du W3C (The Layer Cake).**

à développer des standards et des recommandations pour tous les niveaux, sachant que pour l'instant, les travaux les plus aboutis du Web sémantique traitent principalement les couches basses. Au sein de cette architecture pyramidale, on doit pouvoir identifier le langage le mieux adapté à l'application que l'on souhaite réaliser [Laublet 02]. Nous nous focalisons ici sur les cinq premières couches pour lesquelles le W3C a proposé des standards [Antoniou 04] :

- **Unicode - URI**[3] (Uniform Resource Identifier) : cette couche de base, dont la syntaxe respecte une norme d'Internet mise en place pour le Web, permet d'identifier une ressource avec certitude, de manière unique. L'URI est la technologie de base du Web car tous ses hyperliens sont exprimés sous forme d'URI ;
- **XML**[4] : ce format fournit une syntaxe pour structurer des documents mais n'impose pas de contraintes sémantiques sur le sens de ces documents. Il supporte donc l'interopérabilité syntaxique. Sa syntaxe est parfois utilisée par des langages de plus haut niveau, ce qui garantit leur échange à travers le Web. C'est ainsi l'infrastructure de base du Web sémantique ;
- **RDF***[5] (Resource Description Framework) [Lassila 98] : ce langage permet d'exprimer des contraintes sur les ressources ; c'est un modèle de données standard pour associer aux documents de la sémantique exploitable par les machines. Il permet de représenter toute sorte d'informations et en particulier les méta-données que nous définissons dans la partie qui suit (cf 2.1.2.2 page suivante) ;
- **RDF Schema**[6] [Brickley 00] : c'est un langage dont les caractéristiques permettent d'organiser les vocabulaires RDF en hiérarchies. On va ainsi pouvoir représenter un certain nombre de structures constituant les ontologies ;

[3]http ://www.w3.org/Addressing/
[4]http ://www.w3.org/XML/
[5]http ://www.w3.org/RDF/
[6]http ://www.w3.org/TR/rdf-schema/

– **Ontologies** : comme nous le verrons plus en détail par la suite (cf 2.1.2.3 page 24), elles définissent des vocabulaires et établissent l'usage que l'on peut faire des termes dans le contexte d'un domaine spécifique. De nombreux langages ont été développés pour représenter les ontologies, notamment DAML+OIL[7] (DARPA Agent Markup Language + Ontology Inference Layer) [Horrocks 02] et plus récemment OWL*[8] (Web Ontology Language) [Baader 03] qui a été inspiré de ce précédent langage mais offrant des fonctionnalités supplémentaires. C'est un langage plus riche que RDF Schema puisqu'il dispose d'une sémantique formelle propre et offre des fonctionnalités plus avancées, comme les relations entre concepts (telles que la disjonction) et les caractéristiques des relations (par exemple, la symétrie).

On peut constater que plus le niveau de représentation offert par les langages est haut, plus les technologies correspondantes ont des capacités avancées. Nous verrons par la suite que cela détermine également la puissance et l'automatisation des tâches que les systèmes pourront réaliser en exploitant l'information représentée par ces différents langages. Dans la partie sur les ontologies (cf 2.1.2.3 page 24), nous définirons les notions de règles et de raisonnement et nous aborderons ces points dans la discussion pour montrer en quoi ces technologies avancées peuvent également être utiles dans le cadre de l'intégration de sources de données hétérogènes (cf 7.4 page 178). Nous ne détaillons pas les niveaux supérieurs qui se situent encore au stade de développement.

2.1.2.2 Les méta-données

L'information se trouvant sur Internet est tout à fait satisfaisante pour les hommes mais le problème est qu'elle ne l'est pas pour les machines qui ne peuvent l'exploiter et l'interpréter que de manière très limitée. La solution est de remplacer le langage HTML par des langages plus appropriés pour représenter les ressources Web de manière plus structurée. De plus, les pages Web devraient contenir des informations supplémentaires concernant leur contenu. C'est ce type d'informations que Tim Berners-Lee a défini comme les **méta-données**, c'est-à-dire des « données sur les données »[9]. Plus précisément, ce sont des informations structurées et explicites permettant de décrire des documents. Dans le contexte du Web sémantique, elles constituent un module fondamental puisqu'elles sont la base pour décrire les ressources Web [Nilsson 02]. Les méta-données sont ainsi des marqueurs qui saisissent une partie du sens des données, participant donc à rendre les ressources auxquelles elles sont associées compréhensibles et exploitables par les machines. Même si l'intégration des méta-données aux contenus numériques n'est pas encore largement adoptée, des travaux et des volontés convergent dans ce sens. La principale est la norme Dublin Core[10] qui est une initiative dédiée à ces questions depuis 1995. Elle définit quinze éléments dont la sémantique a été établie par un consensus international de professionnels provenant de diverses disciplines telles que la bibliothéconomie, l'informatique, le balisage de

[7] http ://www.w3.org/TR/daml+oil-reference
[8] http ://www.w3.org/TR/owl-features/
[9] http ://www.w3.org/DesignIssues/Metadata.html
[10] http ://dublincore.org/

textes, la communauté muséologique et d'autres domaines connexes. Ces éléments sont répartis autour de trois domaines, qui permettent d'identifier et de décrire les ressources du Web :

- contenu : Titre, Sujet, Description, Source, Langage, Relation, Couverture ;
- propriété intellectuelle : Créateur, Éditeur, Contributeur, Droits ;
- matérialisation : Date, Type, Format, Identifiant.

Une contribution majeure des méta-données est qu'elles facilitent la recherche d'informations sur Internet. Des moteurs de recherche de référence, tels que CISMeF [Darmoni 00] pour des ressources numériques médicales françaises, les utilisent pour affiner leur mode de recherche. D'autre part, les méta-données sont particulièrement utiles pour gérer l'évolution des systèmes d'une manière flexible, ce qui est une caractéristique indispensable pour améliorer l'accès aux ressources Web [Busse 99], et en particulier dans le domaine biomédical dont les données évoluent très rapidement. En effet, le contenu des ressources change régulièrement et il faut pouvoir être sûr que cette information est à jour. Finalement, les méta-données garantissent en partie l'interopérabilité en assurant l'échange et le partage d'informations rendues lisibles et exploitables par les machines.

En terme d'intégration, elles sont une bonne solution à des stockages trop lourds d'informations [Kashyap 98]. Elles permettent d'abstraire et de capturer l'essentiel des informations se trouvant dans les sources réparties sur Internet, et ce de façon tout à fait indépendante des détails de représentation propres au contenu de chaque source. De plus, les descriptions des méta-données prennent généralement moins de place, au niveau du stockage, que les données elles-mêmes. Des exemples de méta-données sont le nom ou les dates de création et de dernière modification d'une source.

Plus spécifiquement dans le domaine biomédical, Markowitz et al. ont identifié les méta-données suivantes comme nécessaires pour décrire des bases de données, pouvant s'appliquer aux sources de données d'une manière plus globale [Markowitz 97] :

1. des informations générales incluant le nom de la source, son URL, le langage dans lequel la source est décrite, la manière dont elle est implémentée et des mots-clés permettant des recherches de haut niveau dans la source ;
2. le schéma définissant la structure de la source, voire des définitions associées aux différents éléments du schéma ;
3. des vues représentant des interprétations alternatives des sources en fonction des utilisateurs ;
4. les références croisées connues qui existent avec d'autres sources.

Cependant, il est important de souligner que ces méta-données sont très rarement fournies par les auteurs des sources. Elles ne sont souvent disponibles que partiellement et lorsque l'on trouve ces informations, le format dans lequel elles sont décrites est difficilement exploitable. De plus, mettre ce type de méta-données sous un même format est coûteux, tout comme maintenir ces dernières à jour peut être complexe et prendre beaucoup de temps. Il apparaît donc claire-

ment que des outils appropriés pour effectuer ces différentes tâches sont indispensables dans la perspective d'intégration de sources de données biomédicales.

En conclusion, il faut noter que même si les méta-données propose une première solution pour permettre aux machines d'exploiter plus efficacement les ressources Web, elles ne sont pas suffisantes. En effet, leur contribution principale se situe au niveau syntaxique puisqu'elles correspondent à une information descriptive sur les structures des ressources Web. Cela ne permet pas de résoudre les hétérogénéités de type sémantique, nécessitant d'apporter une signification aux termes utilisés au sein des ressources. Par exemple, il faut pouvoir utiliser un terme unique pour représenter la même information dans l'indexation de différents documents. C'est dans cette optique que les ontologies ont été développées.

2.1.2.3 Les ontologies

2.1.2.3.1 Définitions. La notion d'**ontologie** provient de la philosophie et a été utilisée en informatique.

En philosophie, l'ontologie est un domaine d'étude qui s'intéresse à la nature de ce qui est.

En informatique, une ontologie désigne un modèle d'un domaine ou un modèle des connaissances de ce domaine. Il peut donc y avoir plusieurs ontologies. Il n'existe pas de définition précise de ce qu'est une ontologie. La définition la plus répandue est celle donnée par Gruber [Gruber 93] : « *une ontologie est une spécification explicite d'une conceptualisation* ». Cette définition a ensuite été complétée pour rendre compte de l'interopérabilité sémantique, qui est une des raisons de l'étude des ontologies en informatique. La notion de *conceptualisation* se réfère à un domaine et plus exactement à la manière dont ce dernier va être décrit et la *spécification* se rapporte au formalisme qui sera utilisé pour réaliser cette description. En d'autres termes, l'ontologie permet de représenter un ensemble de connaissances de façon explicite, pour qu'elles soient ensuite compréhensibles par les machines.

De plus, alors que la validité de l'ontologie en philosophie est par nature absolue, celle d'une ontologie en informatique est relative et soumise aux choix de représentation. Ceci a mené Guarino à compléter la définition de Gruber pour faire d'une ontologie un accord sur une conceptualisation partagée et éventuellement partielle [Guarino 97b], [Borst 97]. Cette notion de partage correspond bien à l'objectif d'interopérabilité identifié précédemment.

Chandrasekaran a dégagé les éléments qui constituent une ontologie [Chandrasekaran 99]. Une ontologie est ainsi une théorie du contenu sur les sortes d'objets, les propriétés de ces objets et leurs relations dans un domaine spécifié de connaissance.

De manière plus pragmatique, une ontologie consiste en :

- des **concepts** (ou classes) correspondant chacun à un regroupement d'entités ayant des caractéristiques communes ;
- des **propriétés** (ou attributs) associées à ces concepts ;
- des **relations hiérarchiques** permettant aux enfants d'un concept d'hériter des propriétés du concept parent ;
- des **relations associatives** entre concepts (c'est à dire, autres que hiérarchiques [Zhang 04]) ;

- des **axiomes** (ou contraintes) qui ont pour but de définir, dans un langage logique, la description des concepts et des relations permettant de représenter leur sémantique. Par exemple, on pourra définir des restrictions sur la valeur des attributs ;
- éventuellement des **définitions** associées à ces concepts ;
- parfois des **instances** (ou individus) correspondant aux valeurs associées aux concepts.

Une **ontologie formelle** est une *ontologie ayant une représentation dont la sémantique est clairement définie et reposant sur des bases mathématiques logiques* [Guarino 97a], [Bachimont 00].

Les bases mathématiques, avec le fait que les connaissances sont explicites, permettent d'effectuer automatiquement des déductions logiques et donc de rendre les connaissances représentées dans l'ontologie utilisables par des programmes.

Le fait que la sémantique soit clairement définie assure la correspondance entre les éléments de l'ontologie et la réalité. Cela fait appel à des bases philosophiques explicites, comme la notion d'engagement ontologique (*Ontological Commitment* en anglais) [Gruber 93]. Cet engagement cherche à capter et à contraindre un ensemble de conceptualisations. Plus précisément, il offre un moyen de spécifier le sens d'un vocabulaire en contraignant l'ensemble de ses concepts au moyen d'informations explicites sur la nature intentionnelle des concepts et relations ainsi que sur la structure a priori des concepts, c'est-à-dire les relations existant entre eux [Guarino 94]. Cet aspect est nécessairement complémentaire du formalisme mathématique évoqué précédemment. En effet, la logique du premier ordre répond bien au besoin de déduction logique mais n'est pas un bon langage de représentation d'ontologie car il n'y a rien qui indique quels éléments de la réalité doivent être décrits par des prédicats unaires et lesquels doivent être décrits par des prédicats binaires, par exemple [Guarino 95].

Le respect du formalisme garantit qu'on va pouvoir faire du raisonnement formel, dont les conclusions sont cohérentes avec les faits représentés dans l'ontologie (mais ces faits peuvent eux-mêmes être contradictoires). Cela soulève donc la question de la construction d'ontologies sans contradictions internes. Des travaux ont été menés sur des méthodes de construction d'ontologies, notamment [Corcho 03] qui recense les différentes méthodologies existantes et souligne que, même si celles-ci constituent une base pour la modélisation, ces propositions ne sont ni unifiées ni tout à fait matures.

Enfin, en plus des capacités de raisonnement, l'aspect logique des ontologies formelles permet de vérifier qu'elles ne comportent pas de contradictions internes et l'engagement ontologique permet de garantir que le contenu (non contradictoire) d'une ontologie correspond bien à la réalité que l'on cherche à décrire.

Nous définissons quelques notions supplémentaires utiles dans la suite de ce manuscrit et directement liées aux ontologies.

La **relation de subsomption** organise les concepts et relations par niveau de généralité. On dira ainsi qu'un concept `C1` subsume `C2` si `C1` est plus général que `C2`, c'est-à-dire si l'ensemble des instances de `C2` est un sous-ensemble des instances de `C1`. On peut interpréter cette relation comme une spécialisation (toutes les instances de `C2` sont des instances de `C1`) et on dira aussi qu'elle est de type *`est-un`* (*`is-a`* en anglais).

La **relation de composition** établit une correspondance sémantique stable qui véhicule une notion de « connexion », de « faire partie ». On dira ainsi qu'un concept `C1` est composé des concepts `C2` et `C3` si l'ensemble des instances de `C1` contient l'union de l'ensemble des instances de `C2` avec l'ensemble des instances de `C3`. On dit aussi que cette relation est de type *`partie-tout`* (*`part-of`* en anglais).

2.1.2.3.2 Langages de représentation et raisonnement. Les langages qui peuvent être utilisés pour décrire les ontologies formelles sont divers. On citera parmi eux les **frames** ou les **logiques de description**.

Les **frames** sont présentées comme étant une structure de données capable de représenter des connaissances [Minsky 75]. Les concepts y sont représentés par les frames (dans le sens de quelque chose qui peut / doit être rempli) qui sont caractérisées par un certain nombre d'attributs (appelés aussi *slots*) contenant des informations sur leur contenu. Ces attributs peuvent être de plusieurs natures : valeur de l'attribut, ensemble de valeurs, restriction de valeurs, valeur par défaut, une propriété avec une autre frame, une combinaison des différents cas. Les frames se prêtent cependant mal au raisonnement automatique puisque les solutions développées sont souvent ad hoc.

Les **logiques de description** constituent un compromis entre les frames et la logique du premier ordre. Elles sont basées sur trois catégories d'entités, inspirées des frames [Napoli 97] : les individus, les relations (appelées propriétés) entre les individus et les classes qui sont définies comme des ensembles d'individus (leurs instances).

Par rapport aux frames, les logiques de description disposent d'une sémantique rigoureuse pour les individus, les relations et les classes [Baader 91]. Ces dernières sont définies comme des ensembles d'individus. Ceci permet alors d'appliquer le raisonnement ensembliste aux ontologies pour effectuer des inférences et de composer les classes en évitant ainsi l'explosion combinatoire des terminologies. Enfin, cela permet de réaliser une description intentionnelle du domaine, c'est-à-dire que l'on décrit les caractéristiques des classes et des individus et que l'on se base sur ces caractéristiques pour organiser les classes en taxonomies pour déterminer de quelle(s) classe(s) un individu est une instance.

Ces types de langages formels offrent des fonctionnalités avancées grâce à leur expressivité. L'une de ces principales capacités est le **raisonnement** que l'on peut faire grâce à des mécanismes d'**inférence** réalisés par des programmes nommés « classifieurs ». Ces raisonneurs automatiques peuvent déduire (ou inférer) des conclusions à partir d'une connaissance implicite donnée, afin de la rendre explicite. Les deux mécanismes d'inférence majeurs sont la classification de concepts et la classification d'instances.

La classification de concepts permet d'intégrer automatiquement un nouveau concept dans une ontologie. Pour cela, le classifieur exploite les propriétés associées au concept à ajouter afin de déterminer l'ensemble des relations de subsomption existant entre ce concept et tous les autres concepts de l'ontologie. Le raisonneur peut ensuite ajouter les relations nouvellement découvertes pour enfin placer automatiquement le nouveau concept au sein de la hiérarchie.

La classification d'instances permet de classer une instance de manière automatique sous un ou plusieurs concepts dont elle satisfait les propriétés.

Ces mécanismes permettent aussi de vérifier la validité d'informations à partir des axiomes, notamment la détection de contradictions entre deux faits concernant une instance. Les ontologies formelles permettent ainsi un traitement automatique de la sémantique de leur contenu par les machines, constituant un moyen efficace de faciliter la gestion des concepts, leur classification, la comparaison entre leurs propriétés ou simplement le parcours des ontologies pour en consulter le contenu.

2.1.2.3.3 Les différents types d'ontologies. On peut dégager quatre catégories principales d'ontologies selon leur couverture et les tâches pour lesquelles elles sont développées [Burgun 01b] :

- les **ontologies de haut niveau** où sont décrits des concepts liés à l'Espace ou au Temps s'appliquant à tous les domaines. Elles ne doivent pas se référer à des domaines en particulier et les concepts qu'on a besoin de décrire dans un domaine spécifique doivent pouvoir être reliés à une ontologie de haut niveau. Ces ontologies se veulent donc universelles et sont censées ne pas avoir été définies pour des tâches spécifiques. Des exemples d'ontologies de ce type sont SUMO[11] (Suggested Upper Merged Ontology) [Pease 02] et DOLCE[12] (Descriptive Ontology for Linguistic and Cognitive Engineering) [Masolo 02] ;
- les **ontologies générales** qui représentent des connaissances globales, indépendamment d'un domaine et d'une tâche, mais d'un niveau de précision moyen. Par exemple, l'ontologie OpenCyc[13] ;
- les **ontologies de domaine**, spécifiques à un domaine d'étude et indépendantes d'une tâche précise. Un exemple est FMA[14] (Foundational Model of Anatomy) dans le domaine de l'anatomie [Rosse 03] ;
- les **ontologies d'application** ou de tâches qui ont une portée restreinte et sont construites pour des objectifs spécifiques. On citera par exemple l'ontologie « The scheduling task ontology » [Rajpathak 01] qui vise à construire des applications pour gérer les emplois du temps.

2.1.2.3.4 Lien des ontologies avec les systèmes terminologiques. Les ontologies sont centrées sur la notion de concept. Celui-ci est caractérisé par un certain nombre de propriétés communes à plusieurs instances. Parallèlement, d'**autres systèmes de représentation**, dits terminologiques, sont focalisés sur les termes, c'est-à-dire l'aspect textuel de l'information. Par rapport aux concepts, ils correspondent par exemple à l'étiquette permettant de nommer un concept. Ces deux notions sont donc différentes mais pas indépendantes. Il est fréquent d'être confronté à des confusions entre ces systèmes terminologiques et les ontologies, et en particulier dans le domaine biomédical. Un large éventail de ces systèmes existent et diffèrent notamment selon leur portée (c'est-à-dire si elles sont génériques ou spécifiques d'un domaine),

[11] http ://www.ontologyportal.org/
[12] http ://www.loa-cnr.it/DOLCE.html
[13] http ://www.opencyc.org/
[14] http ://fma.biostr.washington.edu/

le type des relations sémantiques qui peuvent être représentées et leur degré de formalisation [Aussenac-Gilles 04]. Une proposition de typologie des systèmes terminologiques existants est donnée dans [de Keizer 00].

2.1.2.3.5 Les bio-ontologies. La confusion entre les ontologies et les systèmes terminologiques est très présente dans le domaine biomédical. Les besoins en représentation des connaissances y sont très importants, mais aussi en ce qui concerne la représentation de l'information textuelle. Nous présentons dans cette section certains des travaux qui se sont penchés sur ces aspects, certains étant à la frontière entre les ontologies et les systèmes terminologiques. Nous verrons que la notion d'ontologie dans le domaine biomédical est plus « légère » que celle présentée précédemment, qui correspond plus à celle utilisée en intelligence artificielle. Il apparaît en effet que, parallèlement au déploiement croissant des ontologies dans ce domaine, la définition réelle des ontologies et leurs modes d'utilisation n'ont pas été complètement intégrés [Soldatova 05], [Cannata 05]. Cela pose problème dans la mesure où les possibilités de partage de connaissances, leur ré-utilisation et l'inférence au sein de ses ontologies sont dès lors restreints, voire même inexistants. Pour faire référence à ce type d'ontologies ou systèmes de représentation dérivés, nous parlons de *bio-ontologies* et en présentons quelques unes pour illustrer ces propos.

Le consortium **Gene Ontology®** (GO) [Ashburner 00], [Consortium 06] a pour but de créer un vocabulaire contrôlé, structuré et commun afin de décrire les rôles des gènes et produits de gènes dans n'importe quel organisme. Elle est organisée suivant trois axes distincts : les *processus biologiques* décrivent les différents rôles d'un produit de gène, les *composants cellulaires* indiquent l'endroit au sein de la cellule où les produits de gènes sont actifs et les *fonctions moléculaires* précisent les activités biochimiques des produits de gènes. Ces trois catégories, définies de manière indépendante dans GO, présentent un niveau de détail très fin et sont organisées sous forme de hiérarchies de plus de 20 000 nœuds. Ces derniers, définis en langage naturel (par exemple, `ferrous iron binding`), sont des concepts reliés entre eux par des relations de types *est-un* et *partie-tout*. Au sein d'une hiérarchie, un nœud peut avoir plusieurs parents et enfants (on parle d'*héritage multiple*) et les relations ne forment pas de cycles (nous détaillerons cet aspect par la suite - cf 4.2.4.1 page 111). Ces hiérarchies peuvent donc être représentées comme des graphes orientés sans cycle, ou DAG (*Directed Acyclic Graph* en anglais). GO, malgré son nom, n'est pas une ontologie au sens propre. En effet, elle n'utilise pas de langage de représentation et son format qui est basé sur les DAGs n'offre pas la possibilité de spécifier des propriétés ou des définitions sur les concepts qui soient interprétables par les machines. D'autre part, les concepts organisés suivant les trois axes ne sont reliés par aucune relation d'une hiérarchie à l'autre alors qu'ils le devraient. Ainsi, comme les processus biologiques agrègent des fonctions moléculaires, une relation de type *partie-tout* devraient inter-connecter les fonctions et processus [Kumar 03].

Il faut cependant noter qu'étant donné son succès et sa large utilisation dans la communauté bioinformatique, de nombreux travaux sont menés pour pallier les problèmes posés par GO. Le projet GONG (Gene Ontology Next Generation) notamment vise à définir GO en logique de description (initialement en DAML+OIL puis en OWL-DL) pour résoudre certaines de ces limites [Wroe 03]. En particulier, cette traduction permet de détecter des incohérences au sein des hiérarchies et d'inférer de nouvelles relations de subsomption entre les termes GO. Dans

ce cadre, des systèmes terminologiques existants comme MeSH[15] sont utilisés afin d'y récupérer des informations supplémentaires et d'associer des définitions formelles aux termes GO. D'autres efforts de formalisation ont été proposés dans [Smith 04]. Smith et al. soulignent principalement le besoin de définir des relations ontologiques autres que *est-un* et *partie-tout* et indiquent que l'utilisation d'un outil de visualisation comme DAG-Edit[16] peut permettre aux utilisateurs d'effectuer cette tâche simplement au travers de l'interface. De plus, les auteurs définissent des critères que doivent vérifier les définitions associées aux concepts, telles que l'intelligibilité imposant qu'une définition utilise des termes plus simples que celui qu'elle cherche à expliquer.

L'**Unified Medical Language System®** (UMLS®) [Lindberg 93] est un système intégrant plus de 100 vocabulaires biomédicaux. L'un de ses composants principaux est le Réseau Sémantique (Semantic Network) qui est un réseau restreint de 135 types sémantiques reliés entre eux par environ 50 relations de type hiérarchique (*est-un*) mais aussi associatives (*affecte*, *produit*, etc). Le graphe correspondant est constitué de deux nœuds racines ; Event, un vaste type regroupant les activités, états ainsi que les processus biomédicaux, et Entity qui contient les entités physiques et conceptuelles. Bien que largement utilisé dans la communauté médicale, et de plus en plus dans le domaine biomédical, ce réseau sémantique ne constitue pas une ontologie. Tout d'abord, il n'utilise aucun langage de représentation, ce qui constitue sa principale limite. De plus, des confusions conceptuelles au sein des types sémantiques ont été identifiées par certains auteurs. Par exemple, Biologic Function est défini comme sous-classe de Natural Phenomenon or Process alors qu'ils ne correspondent pas à un même concept abstrait. En effet, une fonction biologique existe en soit et indépendamment du temps, tandis qu'un processus ou phénomène naturel apparaît à un moment donné dans certaines conditions. Biologic Function devrait plutôt, de ce point de vue, appartenir à l'arbre ayant pour racine Entity [Kumar 03] tout en ayant une relation de type *partie-tout* avec le type sémantique Natural Phenomenon or Process pour les raisons exposées dans le paragraphe précédent.

MGED (Microarray Gene Expression Data) Ontology (ou MO) [Whetzel 06] est une des premières tentatives pour formaliser la description des données d'expression de puces à ADN en biologie. Elle consiste en deux parties : une ontologie stable utilisée pour la production et une ontologie extensible. Les éléments, tous définis en langage naturel, sont de trois catégories : les *classes* correspondant aux différents types d'informations qu'il est nécessaire de représenter, les *propriétés* (ou relations) et les *instances* contenant les valeurs réelles. La structure de MO est, comme GO, organisée sous forme d'un graphe orienté sans cycle. MO contient ainsi 229 classes, 110 propriétés et 658 instances et des synonymes sont intégrés au sein de la définition des concepts, ce qui les rend difficiles à exploiter directement. Des versions aux formats DAML+OIL et OWL sont disponibles. Mais MO est une ontologie incomplète et présente des problèmes dans son organisation [Soldatova 05]. Par exemple, des incohérences existent dans les noms attribués aux classes, où certaines, telles que « Individu », sont nommées de la même façon que certains types d'objets composant MO. De plus, certains noms sont difficilement compréhensibles par les biologistes, comme « Package » qui est informatif pour des bioinformaticiens ou informaticiens

[15] Medical Subject Headings, http ://www.nlm.nih.gov/mesh/meshhome.html
[16] http ://www.godatabase.org/dev/java/dagedit/docs/index.html

uniquement. Les définitions associées aux classes sont, elles aussi, parfois incorrectes à cause de confusions entre concepts abstraits et processus physiques ou bien d'imprécisions ne permettant pas une interprétation directe. Enfin, le processus de raisonnement est limité par le fait que la distinction entre les classes et instances n'est pas claire ou parce que les relations ne sont pas bien définies au travers de propriétés trop nombreuses.

Le projet **Open Biomedical Ontologies**[17] (OBO) est un effort pour créer des vocabulaires contrôlés qui puissent être utilisés conjointement dans les domaines biologique et médical. La librairie OBO, comprenant environ 40 bio-ontologies définies dans le langage du même nom, est à la base d'une expérience collaborative de développeurs d'ontologies qui souhaitent adopter des principes spécifiant les bonnes pratiques du développement d'ontologies. L'objectif est une meilleure interopérabilité des ontologies au sein d'OBO et la volonté d'améliorer la qualité et la rigueur dans les ontologies, et ce de manière à répondre aux besoins croissants d'intégration de données dans le domaine biomédical. Des critères nécessaires au développement d'une bio-ontologie ont été définis dans le cadre d'OBO Foundry[18], comme utiliser un même langage formel, décrire et spécifier clairement son contenu ou encore associer des définitions textuelles aux concepts. Bien qu'OBO s'efforce à mieux définir et formuler les termes, certains auteurs soulignent que des aspects élémentaires constituant les ontologies sont négligés, en particulier concernant les relations associatives [Smith 05] ou l'ambiguïté des relations de type *partie-tout* [Schulz 06].

D'autres bio-ontologies ont été développées pour des objectifs et applications spécifiques, en particulier pour la conception de systèmes d'intégration. Ces dernières, bien que réalisées pour un objectif précis, peuvent malgré tout être génériques (par exemple, TAMBIS Ontology - cf 2.2.3.3.4 page 56). Nous ne les détaillons pas ici mais elles seront présentées par la suite en même temps que les systèmes pour lesquels elles ont été créées. Malgré leurs limites, les bio-ontologies existent et beaucoup de travaux tentent de pallier leurs défauts, notamment OBO Foundry. De plus, de nombreuses applications nécessitent leur création et amélioration. [Schulze-Kremer 02] et [Bard 04] ont recensé quelques unes des applications du domaine biomédical où l'utilisation d'une ontologie est capitale :

- les différences terminologiques et sémantiques sont fréquentes et nécessitent de bien définir les termes propres à chaque source de données pour regrouper les synonymes, même s'ils ne sont pas homonymes, sous un même concept ;
- l'exploitation des larges ensembles de données biomédicales, ainsi que leur annotation et leur interprétation ;
- le besoin de regrouper et fusionner des domaines intéragissant avec la biologie qui ont été historiquement développés indépendamment et qui nécessitent désormais d'être étroitement liés à la biologie. C'est le cas notamment de la médecine mais aussi de la chimie et de la pharmacologie.

[17] http ://obo.sourceforge.net/
[18] http ://obofoundry.org/

2.1.2.3.6 Conclusion. Les ontologies permettent donc de représenter des connaissances de manière à les rendre interprétables par les machines. Les ontologies formelles ont une granularité d'expressivité et de description de l'information très fine et offrent des capacités de raisonnement. Les machines peuvent ainsi non plus uniquement lire les connaissances mais aussi les interpréter et les exploiter automatiquement. Cependant, les bio-ontologies existantes n'ayant pas été développées en accord avec les principes définis en intelligence artificielle, elles ne permettent pas en l'état de réaliser des mécanismes d'inférence et donc de raisonner sur leur contenu. Parallèlement, le domaine biomédical est très demandeur des technologies du Web sémantique et le fait qu'il dispose malgré tout de connaissances formalisées et de cas d'application réels en font un excellent domaine d'étude.

Le problème est donc à la fois d'augmenter la qualité des bio-ontologies et de les combiner entre elles. L'objectif initial du Web sémantique concernant les ontologies était que les données accessibles sur Internet fassent référence aux éléments d'un nombre restreint d'ontologies de haut niveau, complexes et cohérentes afin de mutualiser l'ensemble des informations et ce, dans une représentation globale commune. Il apparaît cependant peu réaliste d'espérer pouvoir trouver des consensus suffisamment larges pour contenter l'ensemble des attentes des utilisateurs du Web. De plus, compte tenu des diversités très importantes existant entre des domaines éloignés et les objectifs bien distincts qui peuvent exister même dans un domaine de travail commun, cela semble irréalisable. Ainsi, il est plus raisonnable d'imaginer un Web sémantique constitué d'un grand nombre de petites ontologies bien définies interagissant entre elles, qui soient moins complexes et donc plus faciles à mettre en œuvre, à entretenir et à exploiter [Hendler 01] et partageant cependant des connaissances communes décrites dans une même ontologie de haut niveau. Pour cela, et plus particulièrement dans le domaine biomédical qui nécessite des efforts conséquents en terme de conception des bio-ontologies, une méthodologie est nécessaire. Une suggestion a été introduite dans [Soldatova 05] ; l'idée est de bien différencier les concepts décrivant des connaissances spécifiques du domaine de ceux qui définissent des notions génériques. Ces derniers pourront être représentés au moyen d'ontologies de haut niveau pré-existantes garantissant que la connaissance de type général est située à un endroit précis, mis à jour correctement pour en assurer la cohérence. La division des connaissances à des niveaux différents (plus précisément, les connaissances d'ordre général versus les connaissances spécifiques d'un domaine) permettra d'améliorer la compréhension et la fonctionnalité des systèmes bioinformatiques.

2.1.2.4 Conclusions

Les limites du Web actuel concernent de nombreuses tâches qui restent manuelles et par conséquent très fastidieuses. En particulier, l'accès aux données présentes dans des sources distribuées est possible mais cet accès est encore réalisé manuellement et le travail consistant à combiner les informations présentes dans les différentes sources requiert également l'intervention humaine. Ceci est lié au fait que l'information disponible sur Internet l'est dans un format lisible et interprétable par les hommes mais pas par les machines, ce qui ne permet pas à ces dernières d'aider les hommes en automatisant certaines tâches qu'ils doivent réaliser à la main, telles que la collecte de données ainsi que leur unification. Au travers des différents composants fondamentaux constituant le Web sémantique et les technologies existantes (ou en cours) développées pour

permettre sa mise en place, il apparaît que de nombreuses applications sont et devraient être facilitées par cette nouvelle vision, plus avancée, du Web. En particulier et c'est la clé de cette problématique, ce Web doté de capacités accrues cherche à rendre le contenu des informations se trouvant sur Internet compréhensibles et donc exploitables par les machines. Pour cela, des données de plus haut niveau, les méta-données, doivent être ajoutées pour permettre l'interprétation des informations qu'elles décrivent. De plus, ces informations doivent être représentées de manière plus formelle, au travers des ontologies, pour augmenter les capacités des machines. Les ontologies permettront de raisonner, et donc de découvrir de nouvelles connaissances déduites des informations existantes. Enfin, ces différents composants seront représentés dans le langage du Web sémantique approprié (ou un équivalent qui offrirait les mêmes fonctionnalités).

Dans le domaine biomédical plus spécifiquement, [Buttler 02] a recensé un certain nombre de capacités dont le Web sémantique dispose pour faciliter l'interrogation de multiples sources et donc alléger la lourde charge que représente ce travail pour les biologistes et médecins. Les méta-données peuvent jouer un rôle important notamment pour décrire des informations concernant la confiance que l'on peut accorder aux sources biomédicales, en fonction de la manière dont elles intègrent leurs données (par exemple, les données d'une source entièrement gérée manuellement peuvent être de meilleure qualité que celles d'une source qui est remplie automatiquement sans validation par des experts). Cette information est particulièrement intéressante puisqu'elle peut orienter le choix d'une source à utiliser au sein d'un système plutôt qu'une autre. Les ontologies, enfin, peuvent en particulier permettre l'évaluation du sens des termes utilisés dans les schémas, les méta-données ainsi que les données elles-mêmes des sources. Il sera ainsi possible, au travers des connaissances représentées au sein des ontologies, d'organiser ces termes en fonction de relations sémantiques. Cet aspect est fondamental pour identifier les éléments pertinents présents dans les sources et les relations existant entre ces éléments de manière à mettre en correspondance des fragments d'informations dans un même cadre de référence, pour finalement fusionner correctement les données issues de sources distinctes. Il apparaît donc clairement que les différentes technologies du Web sémantique peuvent apporter des solutions intéressantes dans le cadre de l'intégration de sources de données hétérogènes et distribuées en général, et dans le domaine biomédical en particulier.

2.2 Approches d'intégration

La problématique d'intégration est un point clé du Web sémantique puisqu'elle a pour but de permettre une recherche d'informations plus simple et automatisée au maximum grâce à des programmes capables d'exploiter et donc d'interpréter les informations se trouvant sur Internet. Ces programmes participent à diverses tâches [Laublet 02] :

- découvrir les sources pertinentes par rapport à une requête posée en utilisant les informations dont on dispose concernant chaque source ;
- accéder à ces sources pertinentes de façon centralisée, plutôt que de laisser les utilisateurs parcourir une multitude de sources dont ils ignorent la représentation, et donc parfois la manière dont il est possible de récupérer leur contenu ;

– agréger automatiquement les résultats obtenus dans les sources afin de pouvoir fournir aux utilisateurs une réponse globale et exhaustive ;
– intégrer des fragments d'informations répartis pour générer une réponse homogène.

Nous verrons que les composants du Web sémantique présentés dans la section précédente sont les acteurs principaux dans la mise en œuvre de systèmes offrant ces différentes fonctionnalités et pouvant ainsi faire face à la distribution, l'autonomie et l'hétérogénéité des sources de données.

Dans le domaine biomédical, différents types de systèmes visant à intégrer des sources de données existent mais, comme pour les ontologies, il est fréquent d'être confronté à des confusions sur la notion de systèmes d'intégration. Les premiers travaux réalisés dans ce sens ont consisté à développer des bases de données intégrant des informations provenant de sources de données distantes (par exemple, GeneCards [Rebhan 98]). D'autres se sont basés sur les liens hypertextes permettant de naviguer entre diverses sources de données pertinentes pour les utilisateurs (par exemple, le portail Entrez [Schuler 96]). Cependant, ce type d'« intégration » ne peut pas être réellement considérée comme telle étant donné que ces systèmes soit fournissent aux utilisateurs des fonctionnalités limitées en terme de requêtes à réaliser sur un ensemble de sources de données, soit ne permettent pas de mutualiser des informations réparties sur des serveurs locaux. Les approches d'intégration développées en informatique ont rapidement été exploitées dans le domaine biomédical, offrant clairement des capacités plus puissantes que les premières propositions et étant particulièrement bien adaptées aux besoins spécifiques de ce domaine complexe.

Nous introduisons tout d'abord quelques notions concernant les sources biomédicales et soulignons les enjeux sur lesquels nous nous focalisons dans ce travail. Nous présentons ensuite les approches que nous avons qualifiées de « simples » puis, de manière plus approfondie, les approches classiques pour la conception de systèmes d'intégration au sens informatique.

2.2.1 Notions et enjeux autour des sources de données biomédicales

2.2.1.1 Notions concernant les sources de données biomédicales

Il est nécessaire de définir certaines notions utilisées lorsque l'on parle des sources de données biomédicales et plus exactement des éléments les caractérisant. Pour cela, nous empruntons des notions issues du modèle entité - relation [Chen 76] et des bases de données relationnelles [Codd 70]. Les notions principales nécessaires pour la suite sont les suivantes :

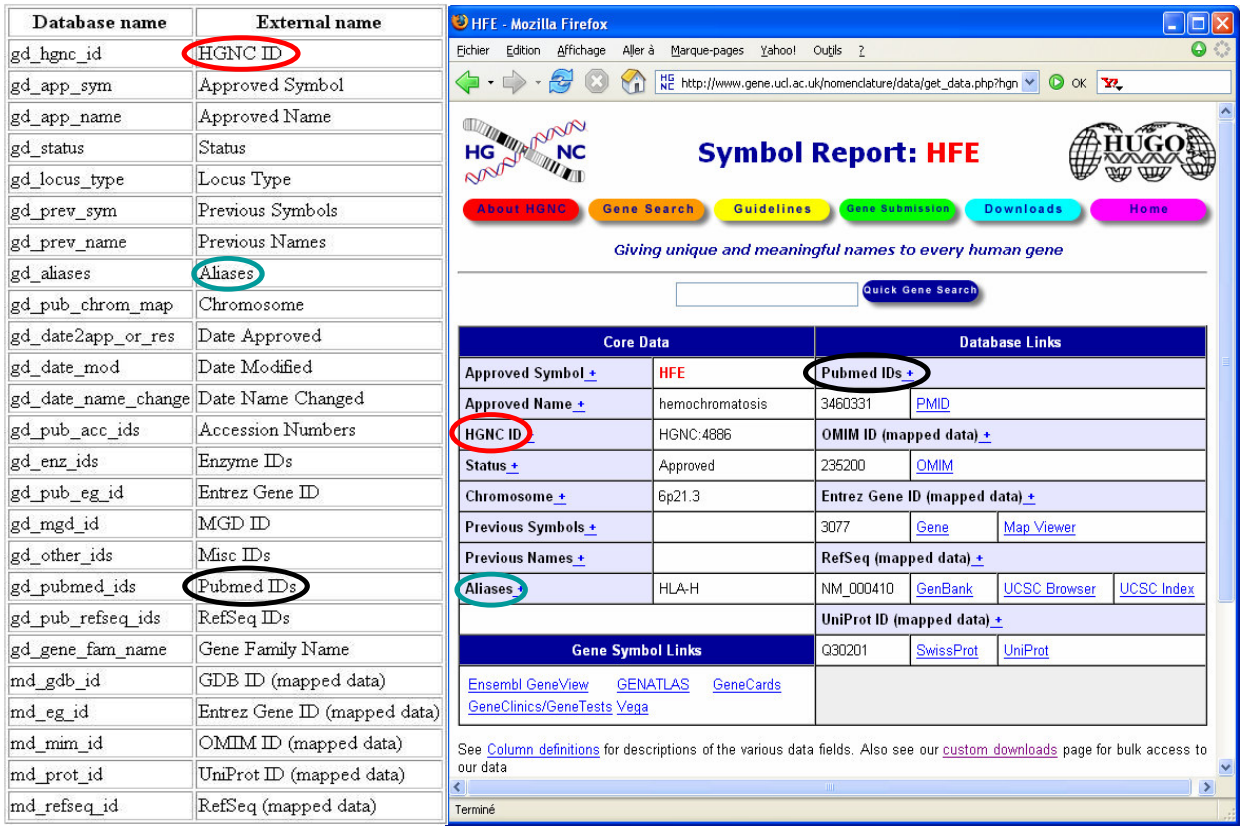
– le **schéma** d'une source est la modélisation de ses données permettant de décrire les connaissances relatives aux entités et relations les représentant ;
– une **entité** est un objet concret ou abstrait, ayant une existence propre, c'est-à-dire pouvant être décrit ou manipulé sans qu'il soit nécessaire de connaître d'autres objets. Chaque entité est décrite par un ensemble d'attributs. Des exemples sont les entités biologiques de base, telles que `Protéine`, `Gène` et `Maladie` ;
– un **attribut** est une propriété d'une entité. L'existence de l'attribut est donc tributaire de celle de l'entité. Des exemples sont « Nom » ou « Symbole », permettant ainsi de déterminer qu'un `Gène` a un « nom » et un « symbole ». À ces attributs sont associées des **valeurs**. Ainsi, on aura respectivement pour les attributs pré-cités des valeurs telles que

« aminolevulinate, delta-, synthase 1 », « hepcidin antimicrobial peptide », « transferrin receptor » et « ALAS1 », « HFE », « TFRC » ;
- une **relation** permet de lier deux entités. Des exemples de relations sont *est-impliqué-dans* liant l'entité `Gène` à `Maladie` ou *est-produit-par* entre `Protéine` et `Gène` ;
- les **références croisées** sont des liens présents dans une source vers d'autres sources de données. Ces références sont définies explicitement au travers d'un lien hypertexte contenant l'URL ou parfois uniquement l'identifiant de l'élément correspondant au sein de la source référencée.

Les schémas des sources de données biomédicales sont rarement disponibles tels quels sur leur site Web ou, quand ils le sont, leur format est assez complexe et n'est surtout pas exploitable par les machines. C'est notamment le cas de la source InterPro [Mulder 05] (source intégrée de familles de protéines, domaines et sites fonctionnels) dont le schéma est accessible sur Internet à l'adresse http ://www.ebi.ac.uk/interpro/interpro_schema_diagram.pdf. Par contre, il est parfois possible de reconstituer ces schémas en récupérant localement les données présentes dans les sources et ainsi identifier les entités, relations et attributs disponibles dans la source en question.

À titre d'exemple, nous présentons ici (Figure 2.4 page ci-contre) les informations que nous avons pu recueillir au sujet du schéma de la source HGNC [Eyre 06]. Cette dernière, développée par le HGNC (HUGO Gene Nomenclature Committee), fournit une nomenclature des gènes humains. L'entité biologique centrale de cette source est donc le `Gène`. Le schéma n'est pas accessible sur le site Web mais le descriptif des attributs présents dans la source est donné à l'adresse http ://www.gene.ucl.ac.uk/cgi-bin/nomenclature/gdlw.pl. Cette liste recense les attributs du schéma de la source HGNC ainsi que le **nom externe** associé à chaque attribut. Ce nom externe, qui est celui utilisé sur la page Web descriptive d'un gène, est plus explicite que le nom de l'attribut tel qu'il est dans le schéma. En effet, on interprète aisément que l'attribut « Entrez Gene ID » donne l'identifiant du gène, sur lequel on a interrogé HGNC, dans la source Entrez Gene [Maglott 05], ce qui est plus difficilement compréhensible en considérant le nom de l'attribut dans le schéma, à savoir « gd_pub_eg_id ». Par contre, il reste des ambiguïtés parmi les noms externes. Par exemple, l'attribut « Chromosome » décrit la localisation chromosomique du gène recherché, information que nous, en tant qu'hommes, savons interpréter grâce à la valeur associée à cet attribut (« 6p21.3 » sur la figure). Cependant, si nous disposions uniquement du nom de l'attribut, nous aurions très probablement penser trouver une valeur autre au travers de l'attribut « Chromosome » (le numéro du chromosome sur lequel se situe le gène par exemple).

L'illustration présentée sur la source HGNC nous permet d'introduire une dernière notion importante. Certains attributs sont suffisamment explicites pour être compréhensibles et interprétables au moyen de leur nom externe (« Entrez Gene ID » ou « Status »). Par contre, d'autres (« Chromosome » ou « Approved Symbol ») restent ambigus ou difficilement interprétables et il est nécessaire de regarder les valeurs associées à ces attributs pour les préciser. Pour résoudre ce type de problèmes, certains systèmes que nous présentons par la suite proposent des solutions. Dans ce cadre, les **approches mises en œuvre se situent au niveau *schéma* si elles utilisent des informations concernant uniquement les attributs** (et références croisées) et



Database name	External name
gd_hgnc_id	HGNC ID
gd_app_sym	Approved Symbol
gd_app_name	Approved Name
gd_status	Status
gd_locus_type	Locus Type
gd_prev_sym	Previous Symbols
gd_prev_name	Previous Names
gd_aliases	Aliases
gd_pub_chrom_map	Chromosome
gd_date2app_or_res	Date Approved
gd_date_mod	Date Modified
gd_date_name_change	Date Name Changed
gd_pub_acc_ids	Accession Numbers
gd_enz_ids	Enzyme IDs
gd_pub_eg_id	Entrez Gene ID
gd_mgd_id	MGD ID
gd_other_ids	Misc IDs
gd_pubmed_ids	Pubmed IDs
gd_pub_refseq_ids	RefSeq IDs
gd_gene_fam_name	Gene Family Name
md_gdb_id	GDB ID (mapped data)
md_eg_id	Entrez Gene ID (mapped data)
md_mim_id	OMIM ID (mapped data)
md_prot_id	UniProt ID (mapped data)
md_refseq_id	RefSeq (mapped data)

FIG. 2.4 – **Liste des noms externes utilisés par la source de données HGNC et extrait d'une page Web décrivant le gène HFE.** À gauche, les différents attributs constituant la source sont listés ainsi que les noms externes de ces attributs, utilisés sur la page Web résultat du descriptif d'un gène présent dans HGNC. À droite, la page Web décrivant les données fournies par HGNC pour le gène ayant pour symbole « HFE »illustre les noms externes des attributs de la source et la valeur associée à chacun. Des exemples de correspondances entre le descriptif général des noms externes et leur utilisation dans une page Web spécifique sont entourés par des ovales.

au **niveau *instances*[19] si elles exploitent les valeurs associées aux attributs**, c'est-à-dire les données elles-mêmes.

2.2.1.2 Enjeux pour l'intégration des sources de données biomédicales

Parmi les différents **enjeux** de l'intégration, nous souhaitons focaliser plus particulièrement cet état de l'art sur les points suivants :

- l'**extraction d'informations à partir des sources de données** de manière à obtenir une description précise de leur contenu, pour en déduire des éléments décrivant les sources, c'est-à-dire le schéma (entités, attributs et relations éventuellement), les références croisées, et des méta-données d'ordre général sur la source (son nom, son URL d'interrogation, etc). Nous verrons par la suite que cette extraction est généralement faite manuellement, soit en exploitant le schéma décrivant la source quand il est disponible, soit en identifiant, dans la source directement, les informations permettant de définir le type de données ainsi que les éléments représentés dans la source, et ce par une intervention humaine. Ceci est possible soit en utilisant des descriptions textuelles, comme celles fournies par HGNC (Figure 2.4 page précédente - partie gauche), soit en analysant directement les données de différentes pages, ce qui peut être très fastidieux. Pouvoir extraire cette description des sources de manière automatisée facilite la conception des systèmes d'intégration et leur maintenance. En effet, le processus d'extraction pourra être exécuté périodiquement de manière à identifier d'éventuelles modifications des sources intégrées et ainsi mettre le schéma de ces dernières à jour beaucoup plus facilement que si la veille stratégique devait être faite manuellement ;
- la **définition d'un *schéma global*** dans lequel représenter l'ensemble des éléments des schémas de sources de données à intégrer. Les schémas des sources sont définis comme des *schémas locaux* en opposition à la notion de schéma global qui vise à les unifier. Nous souhaitons analyser les efforts effectués dans les différents systèmes existants pour définir un schéma global, et notamment d'éventuelles automatisations des diverses étapes y menant ;
- l'**existence de méthodes situées au niveau *instances*** pour compléter les informations présentes au niveau *schéma*. Pour réaliser une intégration au moyen d'approches avancées (que nous présentons par la suite - cf 2.2.3 page 39), il faut exploiter le niveau *schéma* pour pouvoir identifier les informations indispensables à une description unifiée mais aussi celles qui sont nécessaires pour pouvoir accéder aux sources. Mais les données peuvent aider à préciser certaines informations, comme nous l'avons illustré ci-dessus (cf 2.2.1.1 page 34), raison pour laquelle nous souhaitons également considérer les techniques éventuelles développées au niveau *instances* ;
- la **maintenance du système d'intégration** est un enjeu déterminant dans le domaine biomédical où de nombreuses données apparaissent continuellement et donc où les sources les fournissant sont en perpétuelle évolution. Dans ce cadre, des outils doivent être mis en

[19] Notons au passage que nous choisissons d'utiliser le terme *schéma* au singulier puisque les méthodes exploitent les informations se trouvant dans un schéma alors que le terme *instances* est au pluriel car cela correspond aux différentes instances du schéma. Cet aspect sera plus particulièrement détaillé dans la section 2.3 page 71 où différentes approches des deux types pré-cités sont présentées

œuvre pour faciliter la mise à jour des différents composants du système d'intégration, en particulier dans le cas des deux premiers points.

Dans les deux parties suivantes, nous présentons les différentes approches d'intégration existantes et les systèmes les implémentant. Étant donné le nombre de systèmes d'intégration déjà développés, nous avons détaillons uniquement certains d'entre eux qui nous paraissaient plus particulièrement intéressants. L'inventaire proposé n'est donc pas exhaustif.

2.2.2 Approches simples

Deux types d'approches d'intégration offrent des fonctionnalités restreintes : les bases de données intégrées et les portails. Leurs caractéristiques sont les suivantes. Les **bases de données intégrées** sont des systèmes créés spécifiquement pour regrouper des données contenues dans d'autres sources de données. Leurs caractéristiques sont les mêmes que les bases de données, c'est-à-dire la structuration et l'organisation de grandes quantités d'informations afin d'en faciliter l'exploitation. Leur spécificité est qu'elles contiennent des informations qu'elles ont récupérées dans d'autres sources de données. Le but est de récolter et de stocker les données considérées comme pertinentes pour les concepteurs dans un base de données intégrées unique. L'avantage est qu'ainsi, les utilisateurs ont un accès centralisé aux informations qui sont pourtant distribuées. Le problème est que le choix des données à récolter dans les sources de données externes est faite de manière arbitraire par les concepteurs. De plus, la mise à jour d'un tel système peut nécessiter de changer entièrement le schéma de la base de données intégrées.

Le **portail**, au sens informatique du terme, est un type de site Web qui propose une indexation thématique des ressources s'appliquant à un domaine précis, sans proposer nécessairement d'informations par lui-même[20]. Ainsi, c'est une porte d'entrée unique sur un large panel de ressources et de services centrés sur un domaine ou une communauté particulière. Il est alors possible pour les utilisateurs d'interroger un portail central afin de retrouver en une requête unique toutes les données se trouvant dans les sources gérées par le portail. Le problème est que celui-ci va indiquer aux utilisateurs quelles informations sont accessibles dans ces différentes sources, sans le guider vers une source plutôt que vers une autre. Ce sont les utilisateurs du portail qui doivent choisir quelle(s) source(s) leur semble(nt) pertinente(s) et aller naviguer parmi elle(s), parfois sans aucune idée préalable du type d'informations s'y trouvant. De plus, si une requête implique plusieurs sources alors il n'est pas possible de récupérer un résultat global directement ; les utilisateurs doivent se charger de récupérer certaines informations dans une source pour aller ensuite récupérer le reste dans une autre.

GeneCards [Rebhan 98], qui est une encyclopédie sur les gènes humains, est un exemple de base de données intégrées. Elle rassemble sous une forme intuitive des informations sur les gènes, les protéines, les séquences et les pathologies ainsi que les interactions entre ces différentes entités. Le principe consiste à entrer un symbole de gène ou des mot-clés pour obtenir la page Web décrivant le gène correspondant ou une liste de pages descriptives de gènes dans lesquelles les mot-clés ont été trouvés. Les informations sont issues des bases de données les plus couram-

[20] http ://fr.wikipedia.org/wiki/Portail_web

ment utilisées sur leurs sujets, entre autres Swiss-Prot et GDB. De ces sources sont sélectionnées uniquement les données pouvant être utiles, selon les concepteurs, pour offrir une vue d'ensemble des connaissances existantes du moment. Depuis 2002, les diverses informations sont stockées sous forme de fichiers plats au format XML [Safran 02]. Bien que plus facilement exploitable que des fichiers textes, ce format est limité pour faciliter la conception de requêtes. Malgré le développement d'un module permettant d'étendre les requêtes des utilisateurs (par recherche en texte libre de chaque terme de la requête si aucun résultat n'est trouvé avec un nom ou symbole de gène) et quelques fonctionnalités supplémentaires (correcteur orthographique et exploitation des stratégies d'interrogation effectuées par les utilisateurs), les requêtes possibles restent assez simples.

MADSENSE [Teusan 03] est une autre base de données intégrées visant à regrouper des informations biologiques dans un système unique compréhensible et exploitable par l'homme. MADSENSE contient des données issues de bases publiques, notamment PubMed [Wheeler 06] et KEGG. L'interface Web fournie par cette base intégrée a pour but d'aider les utilisateurs dans leurs travaux d'interprétation et de validation des données obtenues au cours des expérimentations sur les puces à ADN. Il est possible de chercher des informations à partir d'un (ou des) symbole(s) de gène(s) ou d'identifiants GenBank. Trois modules ont été développés : le premier fournit la carte d'identité d'un gène donné permettant de visualiser les gènes liés à celui-ci en exploitant des données biologiques et bibliographiques, un module offre la possibilité de regrouper des gènes ayant des profils d'expression proches en fonction de données présentes dans Gene Ontology. Enfin, un autre module recense les publications les plus pertinentes concernant les interactions identifiées entre deux gènes. Là encore, malgré des fonctionnalités intéressantes proposées dans MADSENSE, les capacités de requêtes fournies aux utilisateurs sont limitées.

Entrez[21] [Schuler 96] est un portail qui fournit une interface d'interrogation des sources de données du *National Center for Biotechnology Information* (NCBI) incluant notamment des séquences nucléotidiques et protéiques, des structures tri-dimensionnelles de protéines et des données bibliographiques issues de PubMed. C'est à la fois une base de données et un système de recherche qui présente une vue intégrée des données biomédicales et de leurs inter-relations. Entrez permet des recherches en texte libre en utilisant des requêtes booléennes simples dans environ 30 bases de données. Il est qualifié de portail puisqu'il permet uniquement de chercher dans ces différentes bases de données et d'indiquer le nombre d'enregistrements trouvés dans chacune. Ce sont finalement les utilisateurs qui se chargent de choisir les bases de données susceptibles de les intéresser et ensuite du travail de navigation et de collecte des informations pertinentes.

Le portail offre des fonctionnalités d'intégration limitées puisqu'il se contente d'indiquer aux utilisateurs les sources qui fournissent un résultat pour la requête qu'ils ont posée sans les guider plus en fonction de critères de préférence, par exemple. Il ne permet pas non plus de récupérer des informations impliquant le parcours de plusieurs sources. Les bases de données intégrées présentent un inconvénient important : le choix des informations à récupérer dans les sources est fait par avance par les concepteurs. Ce sont eux qui décident si telle source ou même si telle

[21] http ://www.ncbi.nlm.nih.gov/gquery/gquery.fcgi

information se trouvant dans une source sont pertinentes pour les utilisateurs et les intègrent ou pas au système en faisant parallèlement et manuellement les adaptations nécessaires au sein du système. Cela pose problème puisque d'un utilisateur à l'autre, les besoins sont différents et le fait de choisir *a priori* ce qui va avoir de l'intérêt dans le système est trop rigide. De plus, le schéma de la base de données intégrées ou du portail, qui est créé manuellement, ne vise pas ici à unifier l'ensemble des schémas des sources utilisées mais à représenter uniquement les éléments qui sont utiles au système. Il est effectivement défini uniquement pour être capable de stocker ou répertorier les informations accessibles dans le système. L'évolution de ce type de systèmes d'intégration pose problème car ils ne sont pas flexibles. L'ajout d'une source peut effectivement impliquer la re-définition entière du schéma représentant les données dans le système. En conclusion, ces différents systèmes sont développés dans des buts assez précis et de manière non extensible et offrent des capacités de requêtes limitées. Ils ne sont donc pas, à notre sens, adaptés pour offrir aux biologistes et médecins les fonctionnalités suffisantes pour les aider efficacement dans leur recherche d'informations.

2.2.3 Approches avancées

La communauté informatique a initialement défini les systèmes permettant de réaliser l'intégration de sources de données comme des « systèmes d'information fédérée » [Busse 99]. Une différence majeure entre les différents types de ces systèmes existants concerne le stockage des données présentes dans les sources : on distingue l'approche **matérialisée** et l'approche **virtuelle (ou non matérialisée)**. La première consiste à stocker tout ou une partie des données se trouvant dans les sources à intégrer tandis que l'approche virtuelle vise à intégrer uniquement des informations permettant de décrire les sources, leur contenu réel restant à sa place. D'autres caractéristiques différencient les systèmes d'intégration existants. Dans le domaine biomédical, les trois architectures principales sont : **les entrepôts de données**, **les systèmes d'intégration navigationnelle** et **les systèmes basés médiateurs**.

1. Les systèmes dits entrepôts de données (*Datawarehouses* en anglais).

   Dans ces systèmes, l'intégration est matérialisée et il existe un schéma global dans lequel les données des différentes sources (ayant leur propre modèle) doivent être représentées. Les données des sources sont donc entièrement ou en partie stockées dans l'entrepôt et les utilisateurs peuvent soumettre des requêtes directement sur ces données locales au lieu d'interroger chaque source indépendamment. Ici, l'intégration a lieu en amont, au moment de l'ajout des données à l'entrepôt.

2. Les systèmes d'intégration navigationnelle.

   Ils permettent une intégration virtuelle et n'offrent pas de schéma global mais uniquement un langage de requêtes unique pour accéder à différentes sources distribuées. Cela permet de laisser les sources autonomes mais parallèlement, cela suppose que les utilisateurs se chargent de choisir les sources pertinentes vis à vis de leurs besoins, mais aussi qu'ils fassent la fusion des résultats obtenus. Dans ce cas, l'intégration est partielle et a lieu en aval lors de l'expression de la requête et du traitement des résultats.

3. Les systèmes basés médiateurs.

Ils réalisent une intégration virtuelle au moyen du *médiateur* qui joue le rôle de schéma global des différentes sources. Le médiateur interagit avec celles-ci au travers d'*adaptateurs* (*Wrappers* en anglais) qui se chargent d'interroger les sources et de rendre le résultat au médiateur qui s'occupe de fusionner les différentes réponses de manière homogène et globale. Ici, l'intégration a lieu en amont au moment de la création des adaptateurs.

Nous rentrons ici dans les détails de ces différentes architectures en précisant les avantages et inconvénients des trois approches et nous recensons un certain nombre de travaux ayant mis en œuvre ce type de systèmes. Par souci de simplification, nous considérons uniquement les systèmes d'intégration centralisés alors qu'il existe aussi des **approches décentralisées** [Hacid 04]. Dans ce cas, certains composants normalement uniques dans les systèmes, comme le médiateur, sont multiples. Cela permet une meilleure flexibilité des systèmes mais complique en même temps leur mise en œuvre. D'autre part, les méthodes nécessaires pour développer des systèmes basés sur une approche décentralisée étant pour la plupart semblables à celles utilisées pour les systèmes centralisés (avec cependant un niveau de complexité supplémentaire), il nous a semblé suffisant de nous limiter aux approches centralisées.

2.2.3.1 Entrepôt de données

2.2.3.1.1 Architecture des entrepôts de données. Cette approche permet de regrouper des données extraites de multiples sources dans un entrepôt centralisé. Cela nécessite de convertir le schéma de chaque source dans un schéma unique et global avant de pouvoir stocker leurs données physiquement dans l'entrepôt. Ce schéma commun doit ainsi prendre en compte l'ensemble des spécificités propres aux différents schémas des sources, de manière à garantir la représentation, et l'intégration dans un deuxième temps, des données à stocker dans l'entrepôt. Les systèmes d'intégration basés sur les entrepôts de données comportent donc les éléments suivants (Figure 2.5 page 42) [Widom 95] :

- les **pseudo-adaptateurs** sont connectés à chaque source et ont pour but de transformer les informations décrites au sein des schémas locaux dans les termes du schéma global utilisé par l'entrepôt. Ces composants ne font pas systématiquement partie des entrepôts, mais cette phase de transformation étant indispensable, elle est implémentée dans chaque entrepôt même si c'est sous une autre forme. De manière facultative, se trouvent au même niveau des moniteurs qui, eux, ne jouent aucun rôle dans la conception initiale du système mais visent à gérer la maintenance de celui-ci. Ils ont la charge de détecter automatiquement les modifications survenues dans les sources de données et de les propager dans l'entrepôt. Un composant pseudo-adaptateur/moniteur doit être associé à chaque source dont une partie ou l'ensemble des données sont à intégrer dans l'entrepôt puisque ces éléments sont spécifiques du type de la source dont ils gèrent la sélection, l'extraction et l'évolution des données ;
- l'**intégrateur** réalise l'installation des informations au sein de l'entrepôt. Les données doivent être mises en forme et unifiées afin d'en assurer la cohérence, ce qui nécessite de les normaliser et de bénéficier d'un référentiel unique et cohérent ainsi que de bonnes

règles de gestion. Cela implique de filtrer, résumer et fusionner des données provenant des diverses sources ;

– l'**entrepôt** lui-même stocke des données intégrées et historisées. En effet, les données existant déjà dans l'entrepôt ne sont pas supprimées et lorsque cela est nécessaire, de nouvelles données plus récentes et/ou modifiées sont intégrées de manière incrémentale. Un référentiel de temps doit donc être mis en place afin de pouvoir identifier chaque donnée dans le temps. Son modèle est défini au moyen du schéma global qui regroupe l'ensemble des entités représentant le type de données se trouvant au sein de l'entrepôt. C'est l'unique composant avec lequel interagissent les utilisateurs puisque l'interrogation du système consiste ensuite à poser les requêtes directement sur les données de l'entrepôt et non pas dans les sources elles-mêmes. Les entrepôts reposent sur le système OLAP (On Line Analytical Processing) qui travaille en lecture seule et dont les programmes permettent de consulter d'importantes quantités de données pour procéder à des analyses. Des outils d'analyse ou de manipulation des données intégrées au sein de l'entrepôt sont généralement disponibles de manière à offrir des fonctionnalités diverses aux utilisateurs. L'approche entrepôt est notamment bien adaptée aux méthodes de fouille de données (*Data mining* en anglais) [Palpanas 00]. Il s'agit d'un ensemble de techniques qui permettent d'extraire des modèles d'une source de données historisées afin de décrire le comportement actuel et de prédire les comportements futurs. La fouille de données se base sur une ensemble de techniques d'extraction de connaissances. Des exemples de techniques mises en œuvre sont les statistiques et la découverte de règles.

2.2.3.1.2 Avantages des entrepôts de données. Cette approche présente de nombreux avantages, en particulier dans le cas où les biologistes et médecins souhaitent utiliser un système d'intégration leur permettant de modifier les données s'y trouvant.
En effet, comme les données sont physiquement présentes dans l'entrepôt, il est possible de les **traiter localement**. Un premier avantage dans ce cadre se situe au niveau du filtrage : les concepteurs de l'entrepôt peuvent sélectionner uniquement les données qui leur semblent pertinentes, mais aussi, les vérifier et les modifier si cela est nécessaire. Ainsi, des validations et corrections peuvent être mises en place au moment de l'intégration, mais aussi la suppression de redondances éventuellement identifiées entre des sources distinctes. De plus, les données peuvent être sécurisées, ce qui est particulièrement intéressant dans le domaine biomédical. Les biologistes veulent garder leurs données privées tant qu'elles ne sont pas publiées. Les médecins doivent garantir une protection sur les données se trouvant dans l'entrepôt si elles concernent des patients. Un autre bénéfice au stockage local des données est qu'il est possible pour les utilisateurs de faire des annotations sur celles-ci, chacun pouvant y ajouter des éléments supplémentaires par rapport à ce qui a déjà été proposé. Un suivi des mises à jour effectuées sur les données depuis le début de leur intégration dans l'entrepôt peut également être réalisé, cette historisation pouvant être utile aux biologistes et médecins pour l'exploitation et l'interprétation des données.
Parallèlement, les systèmes basés entrepôt ont l'avantage d'**éviter les problèmes de connexion** au réseau, d'accessibilité des sources et de lenteur de réponses de leur serveur. Au moment de l'interrogation, le système ne dépend pas des sources directement et donc de leur disponibilité sur le réseau à ce moment précis. De plus, les tâches de mise en correspondance et d'intégration,

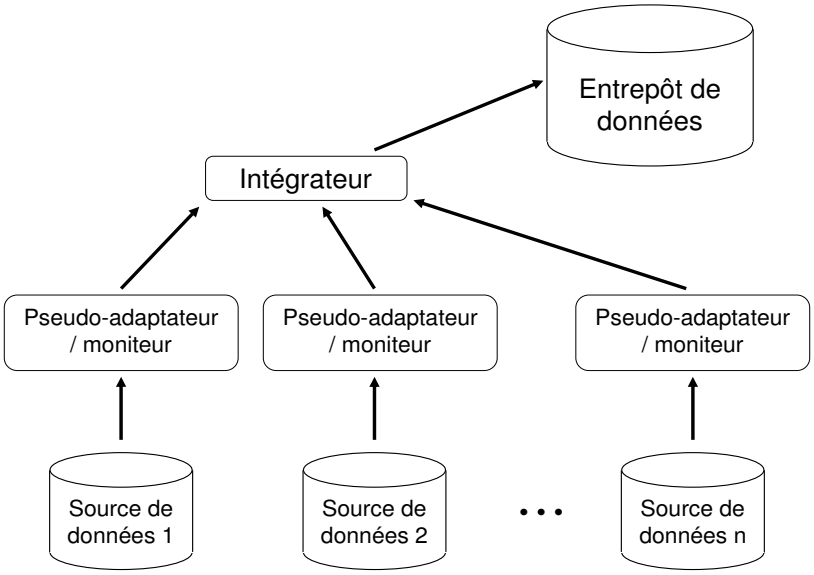


FIG. 2.5 – **Architecture d'un système basé sur un entrepôt de données.** Les flèches représente le chemin suivi par les données à intégrer à l'entrepôt. Les données sont d'abord récupérées dans les sources au travers de leur pseudo-adaptateur (ou moniteur au cours de la mise à jour du système) qui les transmet à l'intégrateur. Celui-ci effectue les traitements (fusion, nettoyage, etc) nécessaires sur les données avant de les intégrer à l'entrepôt.

qui peuvent avoir des répercussions sur le temps de réponse du système, ont déjà été effectuées lors de l'intégration des sources. L'entrepôt permet donc de répondre rapidement aux requêtes des utilisateurs en exploitant exclusivement les données qu'il contient. Cela garantit ainsi une meilleure fiabilité du système.
Enfin, l'approche entrepôt **simplifie l'optimisation de requêtes** puisque cette dernière peut être faite localement sur les données intégrées dans l'entrepôt, en se basant par exemple sur des informations indexées, une fois intégrées au sein du système [Davidson 95].

2.2.3.1.3 Inconvénients des entrepôts de données. La première difficulté rencontrée avec l'approche entrepôt de données concerne **l'implémentation du système**. En effet, en plus de devoir connaître les sources à intégrer pour identifier quelles informations sont pertinentes dans chacune d'elles (qui est cependant un problème commun aux différents types de systèmes d'intégration), il faut ensuite réaliser l'extraction des données, ce qui requiert un système d'interrogation ou de rapatriement des données efficace dans les sources d'origine. La définition du schéma global permettant de décrire l'ensemble des données à ajouter dans l'entrepôt est complexe puisqu'il faut qu'il soit suffisamment expressif pour couvrir les différentes spécificités des sources. Une fois ce schéma défini, le problème réside dans la mise en forme des données des sources en fonction de ce schéma global de manière à ce qu'elles soient intégrées correctement à l'entrepôt. Cette partie est particulièrement délicate puisque des transformations, parfois complexes, des éléments constituant le schéma des sources d'origine ainsi que des

données correspondantes sont requises pour qu'ils puissent être intégrés au sein de l'entrepôt. Ces différentes étapes indispensables à la conception de l'entrepôt sont lourdes et pénibles à réaliser, c'est pour cela que le besoin d'automatiser au maximum des processus pouvant faciliter ces tâches a émergé. Cela permettra ainsi de réaliser certaines tâches de manière systématique, notamment le remplissage de l'entrepôt, tandis que d'autres ne pourront être que partiellement automatisées, comme, par exemple, les vérifications de cohérence de données ou leur nettoyage, où les experts gérant l'entrepôt ont un rôle indispensable de validation [Schonbach 00].
Un inconvénient majeur de ce type de système est **sa maintenance**. Tout d'abord, il n'y a aucune garantie que l'entrepôt fournisse des informations à jour. En effet, les sources peuvent modifier leur contenu sans le notifier et dès ce moment là, l'entrepôt devient obsolète puisqu'il contient des données qui ont éventuellement changé. Ces problèmes d'évolution sont d'autant plus compliqués si les données internes à l'entrepôt ont été modifiées, corrigées et/ou annotées. En fait, ces modifications étant propres à l'entrepôt, les sources n'en bénéficient pas. Ainsi, dans le cas où des données de l'entrepôt nécessitent d'être mises à jour parce qu'elles ont été modifiées dans leur source d'origine, les problèmes qui se posent sont les suivants : va-t-on pouvoir trouver la correspondance entre la donnée modifiée dans l'entrepôt et celle qui doit désormais être prise en compte ? Comment répercuter les modifications internes à l'entrepôt sur la donnée qui a été mise à jour ?
Enfin, l'autre inconvénient important des entrepôts de données est celui du **stockage des données**. Dans le domaine biomédical en particulier, cet aspect est déterminant puisque la quantité d'informations existantes est énorme et ne cesse de croître [Galperin 06]. Stocker les données dans un système global fait logiquement émerger des limites à cette approche. Il devient indispensable de faire des choix pour réduire ou résumer les données au maximum au moment de l'intégration, ce qui peut supposer une perte d'informations pertinentes. De plus, les systèmes qui gardent un historique des données devenues obsolètes sont confrontés à deux problèmes : cette information additionnelle occupe une place conséquente et précieuse au sein de l'entrepôt mais aussi cela nécessite une tâche supplémentaire dans la maintenance de l'entrepôt : purger celui-ci des données obsolètes qui peuvent être finalement erronées ou simplement inutiles.

2.2.3.1.4 Entrepôts de données existants. Nous présentons ici quelques systèmes basés sur l'approche entrepôt de données et qui offrent des fonctionnalités avancées cherchant à répondre aux inconvénients de cette approche tout en bénéficiant de ses avantages. Nous introduisons tout d'abord un entrepôt de type général, c'est-à-dire qui ne s'applique pas à un domaine en particulier. Ensuite, nous présentons deux systèmes traitant des données du domaine biomédical pour finir avec un système spécifique à une application particulière du domaine biomédical, à savoir l'analyse du transcriptome.

Le système **Xylème** [Xyleme 01] est un entrepôt permettant de stocker des données XML de n'importe quel domaine et présentes sur le Web. Il n'est pas spécifique au domaine biomédical mais étant donné les nombreuses fonctionnalités qu'il offre, cela nous a semblé intéressant de l'introduire ici. Il est constitué de cinq modules dont le premier permet de rechercher, d'acquérir, et d'intégrer les données à l'entrepôt. La recherche et l'acquisition sont réalisées au moyen de robots d'indexation et de recherche, nommés habituellement *crawlers*, qui récupèrent automatiquement

des documents XML sur le Web. Le second module, dit « sémantique », fournit le schéma global du système, une DTD* (Document Type Definition) qui correspond à la fusion des DTDs associées aux documents XML à intégrer. Des méthodes automatiques et semi-automatiques ont été développées pour effectuer la mise en correspondance entre le schéma global (DTD globale) et les schémas locaux (DTDs propres à chaque document XML à intégrer). On y trouve des approches syntaxiques (inclusion d'un terme dans un autre), sémantiques (utilisation d'une terminologie existante pour obtenir des connaissances supplémentaires) et structurelles (exploitation du contexte d'une propriété, au travers de la hiérarchie). Deux autres modules visent à stocker et indexer les documents XML de l'entrepôt ainsi qu'à traiter les requêtes. Enfin, un module de maintenance est implémenté de manière à pouvoir faire des modifications dans le schéma global en fonction des changements apparaissant dans les documents XML. Des mises à jour du type renommage ou suppression d'attributs au sein des sources sont identifiées de manière automatique.

Cet entrepôt est particulièrement intéressant en ce qui concerne les aspects d'implémentation et de maintenance, dans la mesure où la plupart des tâches sont en partie, voire complètement, automatisées. Cependant, le format XML pour les données de l'entrepôt est limitatif car les tâches de conception des composants d'un tel système, notamment du schéma global, sont moins délicates que pour des cas plus généraux où l'on souhaite intégrer des données moins structurées. De plus, les notions d'annotation, de nettoyage et de correction des données ne sont pas abordées probablement parce que Xylème est un système générique. Il en résulte que les enjeux de maintenance des données elles-mêmes intégrées à l'entrepôt ne sont pas considérés dans Xylème ; des solutions au niveau *schéma* (renommage, suppression d'attributs, etc) sont proposées mais pas au niveau *instances*. En conclusion, les spécificités du domaine biomédical nécessitent d'être traitées à part et bien que ce système aborde un certain nombre d'aspects clés des entrepôts de données, il n'est pas adapté aux besoins précis des biologistes et médecins.

L'entrepôt de données **GUS** [Davidson 01], ou Genomics Unified Schema, est défini comme une plateforme de bases de données génomiques, incluant notamment GenBank, Swiss-Prot et les termes GO. Il s'agit d'un système massif intégrant des données sur les séquences nucléiques et protéiques mais aussi concernant la fonction, la régulation, l'expression et les interactions de gènes ou protéines. Des algorithmes développés localement ou obtenus à partir de l'existant, dont le nombre avoisine les 170, permettent de nombreuses fonctionnalités offertes sous forme d'outils aux utilisateurs. Il est par exemple possible, au travers d'une interface, d'annoter et/ou d'interroger les données au sein de l'entrepôt mais aussi de leur appliquer des méthodes de fouille de données ou d'analyse.

Au niveau conception, des algorithmes facilitant l'intégration ainsi que le nettoyage et la correction des données sont disponibles, permettant ainsi d'effectuer des traitements (semi-automatiques) au niveau *instances*. En terme d'implémentation, le schéma global de GUS, de type relationnel, est très large puisqu'il intègre environ 300 tables, certaines étant spécifiques alors que d'autres sont obtenues directement à partir des schémas de certaines sources, telles que Swiss-Prot. Ce schéma a donc été en partie créé manuellement en fonction des besoins de l'entrepôt. A priori, aucune méthode automatique exploitant le niveau *schéma* n'est proposée dans le but de mettre en correspondance le schéma global et les schémas des sources qui n'ont

pu être intégrés tels quels. Par contre, la maintenance dans GUS est un aspect important. En effet, un historique des données modifiées est complété dès que nécessaire. La nouvelle information remplace ainsi l'ancienne dans la table utilisée pour les requêtes. De plus, un outil permet d'assister les concepteurs du système dans la gestion des modifications de format des schémas des sources, telles que l'ajout ou le renommage d'un attribut.

Biozon [Birkland 06] est un entrepôt de données qui intègre des séquences nucléiques et protéiques, des informations concernant la structure des protéines ou leurs interactions, des voies métaboliques et des données d'expression. Ces données sont récoltées dans plus de 20 bases de données différentes, telles que Swiss-Prot et KEGG, et enrichies de données dérivées qui sont calculées au sein même de Biozon. Celles-ci sont obtenues à partir des données des sources auxquelles sont appliquées des méthodes variées développées dans l'entrepôt, par exemple pour détecter des similarités entre des séquences ou des structures de protéines. Le schéma global est composé de deux éléments : la hiérarchie de classes décrivant les objets biologiques fondamentaux (`Séquence`, `Tissu`, etc) et la hiérarchie des relations existant entre ces objets (*`encode`*, *`décrit`*, etc). Pour représenter les données, un graphe où chaque nœud et chaque arc sont respectivement des instances d'une classe et d'une relation est défini. Les nœuds représentent des objets physiques, tels qu'une séquence protéique spécifique, auxquels sont associés un ou plusieurs descripteurs contenant des informations interprétables par les hommes, comme des annotations ou des résultats expérimentaux. Les objets physiques doivent être non redondants et jouent ainsi le rôle d'identifiants. Les relations associant les objets physiques issus de sources différentes et qui correspondent au même objet sont ainsi explicites et non ambiguës, contrairement aux références croisées. Enfin, Biozon offre des fonctionnalités avancées pour l'analyse des données, au travers d'outils intégrés, mais aussi pour la recherche d'informations au sein du système. La représentation sous forme de graphe donne à chaque objet biologique un réel contexte. Cela permet de proposer aux utilisateurs non seulement les informations sur lesquelles leurs requêtes portent explicitement, mais aussi d'autres données qui y sont rattachées et pouvant potentiellement avoir de l'intérêt.
L'analyse du schéma des sources ainsi que la mise en correspondance des attributs choisis dans les schémas locaux avec les classes du schéma global sont faites manuellement (niveau *schéma*). Cependant, l'intégration des données dans le schéma global, c'est-à-dire au niveau des objets et des descripteurs du graphe, se fait de manière semi-automatique, en prenant soin d'éviter les redondances pour garantir l'unicité des objets biologiques. Parallèlement, un traitement effectué sur les données extraites des sources permet d'obtenir les données dérivées (utilisation d'informations situées au niveau *instances*). En terme de maintenance des données, Biozon permet l'ajout, la modification et la suppression de nœuds et arcs au sein des graphes instanciés tout en assurant la cohérence du système résultant de ces mises à jour et en les répercutant sur les données dérivées. Malgré tout, l'ajout d'un nouvel élément dans les hiérarchies de classes et de relations n'est a priori pas pris en charge dans ce système, ce qui ne permet pas de gérer la maintenance du schéma global.

Le système **GEDAW** [Guérin 05] est un entrepôt de données dédié à l'analyse du transcriptome. Il contient des données de différents types : des données expérimentales issues des

expériences de puces à ADN et extraites d'une base de données relationnelle développée en local, des données génomiques obtenues à partir de la source GenBank ainsi que des annotations biologiques et médicales issues du système BioMeKE [Marquet 05], intégrant les sources GO et UMLS. Le schéma global est orienté objet. La ré-écriture des fichiers XML fournis par GenBank est réalisée en associant les éléments de la DTD propre à GenBank avec les entités du schéma global. Le principe est le même pour les fiches XML créées par BioMeKE. Lors de l'intégration des données des sources à l'entrepôt, des méthodes de réconciliation sémantique (exploitant notamment BLAST[22] - Basic Local Alignment Search Tool - pour les séquences ou BioMeKE pour les gènes semblables mais de noms différents) ont été implémentées pour unifier et/ou filtrer les données. Les utilisateurs peuvent ensuite interroger GEDAW de manière à accéder aux données et annotations concernant un gène, un ARN messager ou une protéine fournis en entrée ou bien pour rechercher des groupes de gènes partageant un même profil d'expression.

La définition du schéma global ainsi que la mise en correspondance DTDs-schéma global (donc située au niveau *schéma*), qui est effectuée au travers de règles syntaxiques, ont été définies manuellement. En revanche, le chargement de l'entrepôt et les techniques développées pour résoudre l'hétérogénéité sémantique des données extraites des sources (exploitation au niveau *instances*) sont réalisés automatiquement. En ce qui concerne la maintenance, il n'y a apparemment pas de méthodes proposées pour faciliter l'évolution de ce système.

De nombreux systèmes suivant l'approche entrepôt existent et nous n'en donnons pas une liste exhaustive. Nous cherchons à identifier les familles principales et à présenter ceux qui abordent au mieux les enjeux qui nous intéressent. L'outil **BioWarehouse** [Lee 06] présente également des aspects intéressants puisqu'il permet de créer des entrepôts bioinformatiques. Un entrepôt développé par les auteurs avec cet outil est aussi disponible sur Internet et il est possible de le récupérer en local afin de l'adapter aux besoins spécifiques de son équipe. Les concepteurs se sont attachés à définir un schéma relationnel simple visant à représenter, au sein d'une table commune, des données traitant d'une même entité biologique même si elles sont issues de sources différentes (évitant ainsi la multiplication des tables dans le schéma global). L'intégration des données dans l'entrepôt se fait grâce à des programmes (*loaders*) développés spécifiquement pour chaque source. Par ailleurs, des tables du schéma global incluent des informations utiles pour gérer la maintenance du système : l'une d'entre elles contient des informations concernant les dates d'entrée et de dernière modification des données et d'autres permettent de stocker des versions différentes d'une même source. Cependant, bien qu'étant un outil de création d'entrepôts, aucune fonctionnalité n'est proposée pour exploiter ces informations de manière à faire évoluer le système automatiquement. La conception du schéma global a été faite manuellement. De plus, même si des classes Java sont disponibles pour faciliter la création d'autres programmes *loaders*, leur conception reste en partie manuelle. Cela pose problème puisque ce sont les administrateurs qui doivent convertir (syntaxiquement et sémantiquement) les informations du schéma de la source au format du schéma de l'entrepôt, laissant donc à la charge des concepteurs les tâches les plus délicates.

BioMart (initialement EnsMart) [Kasprzyk 04] est un entrepôt générique permettant d'interroger de larges sources de données biologiques. Dans le schéma global, les tables centrales sont

[22] http ://www.ncbi.nlm.nih.gov/Education/BLASTinfo/information3.html

les entités biologiques principales telles que le gène, la protéine tandis que les tables satellites (rattachées par des relations de type 1-n ou m-n) décrivent les sources contenant des données concernant ces entités biologiques principales (schéma dit en « étoile inversée »). Il est possible d'interroger BioMart au travers de trois interfaces différentes : une classique où les utilisateurs naviguent dans plusieurs pages qui les aident à spécifier leurs requêtes, y appliquer des filtres au besoin et préciser le format de sortie dans lequel ils souhaitent obtenir les résultats. Une interface graphique est aussi disponible où les éléments (attributs et filtres choisis par les utilisateurs) de la requête sont présentés sous la forme d'un arbre. La dernière interface permet de poser une ou plusieurs requêtes en ligne de commandes (en *batch*). Le schéma global du système a été créé manuellement mais a l'avantage d'être modulaire. Cela permet d'ajouter ou de modifier des informations seulement dans une table centrale ou dans un satellite sans affecter le reste du schéma. Cependant, il n'y a pour l'instant pas d'outils mis en œuvre pour assister ces tâches. Les informations permettant de représenter les sources ont été récupérées manuellement mais l'intégration des données est faite de manière automatique. Par contre, pour ajouter des données issues d'une nouvelle source dans l'entrepôt, il faut créer manuellement le programme associé.

Parmi les systèmes présentés, on constate que les entrepôts de données implémentent des méthodes automatiques pour intégrer les données des sources locales mais ils identifient les informations qui les intéressent au sein des schémas de manière manuelle. Contrairement aux approches virtuelles que nous allons présenter dans les deux parties suivantes, les systèmes d'intégration matérialisée proposent généralement des méthodes avancées, et souvent automatiques, au niveau *instances* de manière à unifier et nettoyer les données qu'ils intègrent. Cela paraît logique puisque de tels systèmes stockent les données des sources localement et ont tout intérêt à les intégrer de la meilleure manière. En revanche, il est rare d'avoir des approches automatisées pour récupérer les informations situées au niveau *schéma* (attributs, entités et méta-données), pourtant utiles pour la conception du schéma global et sa mise en correspondance avec les schémas locaux. De plus, les données des sources sont soit de type structuré (base de données), soit semi-structuré (XML), ce qui rend certaines tâches de l'intégration plus simples à réaliser. Dans le domaine biomédical, la conception du schéma global est faite soit de manière complètement manuelle (Biozon et GEDAW), soit partiellement grâce à l'intégration directe et systématique des schémas de certaines sources dont la modélisation est la même que celui du schéma global (par exemple, le schéma de Swiss-Prot est ajouté tel quel au schéma global de GUS puisqu'il est aussi de type relationnel). La mise en correspondance du schéma global avec les schémas locaux est dans la plupart des cas entièrement manuelle. D'autre part, certains systèmes abordent les problèmes de maintenance principalement au niveau des données intégrées mais pas du tout pour gérer l'évolution des composants du système lui-même (Biozon). Enfin, Xylème traite de manière plus efficace les aspects d'automatisation de certaines tâches mais ne considère pas des problèmes spécifiques du domaine biomédical, comme l'annotation des données et par conséquent la maintenance de ce type d'informations au sein du système.

2.2.3.2 Approche d'intégration navigationnelle

2.2.3.2.1 Architecture des systèmes d'intégration basée navigationnelle. Cette approche est non matérialisée et est aussi appelée **approche à base de chemins** [Cohen-Boulakia 05a].

Elle ne vise donc pas à rapatrier localement les données des sources distantes. Le nombre de sources nécessitant une intervention manuelle de l'utilisateur pour relier deux (ou plusieurs) informations a entraîné l'émergence de ce type de systèmes. Ceci peut arriver au sein d'une même source et, dans ce cas, on parle de lien interne (par exemple dans le cas où un descriptif plus complet de certaines données est accessible en suivant ce lien), mais surtout pour passer de la source interrogée à une autre source (référence croisée). Dans ce cadre, la notion de chemin est définie par une suite de références croisées ou liens internes entre des sources de données. On parle de source(s) d'entrée (ou d'origine) pour désigner la (les) source(s) d'où les chemins peuvent débuter pour répondre à la requête des utilisateurs et de source(s) cible(s) (ou finales) pour définir les sources d'arrivée du chemin et qui doivent logiquement permettre de fournir les résultats escomptés. L'approche d'intégration basée sur la navigation consiste à transformer la requête en chemins qui peuvent chacun mener à des résultats en exploitant les références croisées définies entre sources et les liens internes. Ces systèmes éliminent souvent le modèle originel des données et appliquent à la place un modèle où les sources sont définies en tant qu'un ensemble de pages Web avec leurs interconnexions et points d'entrée. Les composants d'un tel système sont les suivants (Figure 2.6 page ci-contre) :

- des **méta-données représentant les sources**, notamment le type de données que l'on peut y trouver ainsi que les références croisées ;
- le **générateur de chemins**. Les chemins existant entre ces sources peuvent être de deux types : directs ou indirects. Il peut en effet exister des chemins directs entre deux sources mais aussi des chemins passant par d'autres sources, dites intermédiaires, avant de parvenir à la (aux) source(s) cible(s). A partir de points d'entrée proposés par le système, les utilisateurs suivent des liens hypertextes explicites pour naviguer des sources d'origine vers d'autres sources jusqu'à la (aux) source(s) cible(s) dans laquelle (lesquelles) ils doivent collecter les informations qui les intéressent.

2.2.3.2.2 Avantages des systèmes d'intégration basée navigationnelle. Cette approche a comme principal intérêt d'être **intuitive** pour les utilisateurs. Elle se base sur la structure du Web, c'est pour cela que c'est le type de système qui a eu initialement le plus de succès. En effet, les utilisateurs sont habitués à cette organisation et à naviguer en son sein, ils savent donc intuitivement utiliser les systèmes d'intégration basée sur la navigation.
Un avantage évident se situe au niveau de la **conception du système qui est relativement simple** puisqu'il n'y a pas de schéma global à définir et donc pas de correspondance à établir avec les schémas des sources. Ce type d'architecture garantit également aux utilisateurs qu'ils **accèdent aux informations en temps réel** puisque les sources sont interrogées dynamiquement en fonction des besoins.
De plus, [Friedman 99] a souligné que cette approche est parfois plus efficace que les autres approches d'intégration pour retrouver certaines informations. Il est effectivement fréquent que l'on ait besoin de suivre des liens internes présents dans une source menant à des données plus détaillées que celles décrivant la fiche générale d'une entité donnée. Traverser les différents liens hypertextes présents dans les sources intégrées **permet d'accéder à des informations plus profondes au sein des sites Web**, c'est-à-dire qui ne sont pas disponibles dans les premières

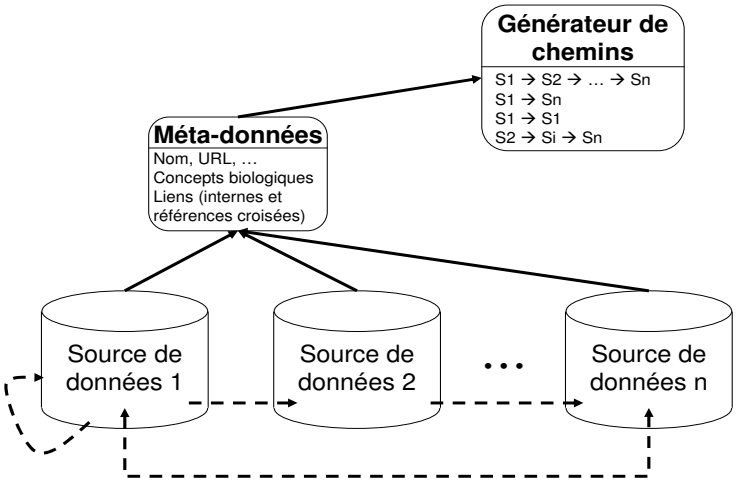


FIG. 2.6 – **Composants d'un système d'intégration basée sur la navigation entre sources de données.** Les flèches pleines correspondent le passage des informations concernant les sources vers le système pour que celui-ci puisse générer les chemins. Les flèches en pointillés correspondent aux références croisées et liens internes existant dans une source pour naviguer vers une autre ou au sein d'elle-même. Un ensemble de flèches pointillés constituent un chemin d'une source d'entrée vers une source cible.

pages obtenues lorsque l'on interroge les sources.
Enfin, la ré-écriture des requêtes sous forme de chemins est **intéressante en terme d'optimisation**. Il est possible de prédire le « coût » que peut avoir une requête en fonction du chemin que l'on empruntera, et ce en fonction de certains critères comme le degré de confiance que l'on peut attribuer à une source ou encore la quantité de résultats que peut fournir une source cible. Cela peut guider les utilisateurs pour choisir un chemin plutôt qu'un autre, notamment s'ils veulent traverser une source intermédiaire donnée plutôt qu'une autre ou bien s'ils préfèrent qu'une source donnée serve de cible parce qu'elle rend plus de résultats que d'autres.

2.2.3.2.3 Inconvénients des systèmes d'intégration basée navigationnelle. Le fait que plusieurs chemins existent pour passer d'une source à une autre pose des problèmes puisque cela nécessite de **déterminer quel chemin est le meilleur** en fonction de critères définis par les concepteurs (par rapport aux besoins des utilisateurs). Il est possible de présenter uniquement aux utilisateurs l'ensemble des chemins possibles entre deux sources mais cela risque de le désarmer face à des résultats trop nombreux et pas forcément simples à comprendre pour eux. Il faut donc offrir aux utilisateurs un ordre de préférence (défini par défaut ou à demander aux utilisateurs) parmi les chemins possibles, de manière à déterminer le meilleur à emprunter pour obtenir des résultats satisfaisants. Cela impose évidemment de préciser les critères de satisfaction choisis pour effectuer ce classement.
Un autre inconvénient de la navigation est lié aux problèmes d'**incompatibilités ou d'ambiguïtés** au niveau des noms utilisés d'une source à une autre. Par exemple, les noms de gènes ne

sont pas forcément identiques d'une espèce à l'autre, ce qui rend difficile la comparaison de résultats obtenus entre des espèces différentes lorsque la correspondance entre les différentes sources n'existe pas. Dans ce type de cas, il n'est pas possible d'identifier aisément des liens explicites entre deux sources pour combler l'absence de référence croisée.
Cette approche **ne permet pas d'assister suffisamment le processus de requêtes**. En effet, le principe consistant à proposer un certain nombre de chemins possibles permettant de répondre à une requête rend les utilisateurs responsables de la tâche d'intégration. Ce sont eux qui se chargent de décider de la manière dont il faut exploiter les connections existant entre les données d'une source à l'autre mais aussi d'en interpréter le sens [Stein 03]. Cela revient à rendre plus complexe le travail de recherche d'informations pour les utilisateurs, effet contraire de celui escompté par l'intégration.
Enfin, des inconvénients existent également au niveau de la **maintenance** d'un système basé sur la navigation entre sources. Une référence croisée suppose que la source cible du lien soit toujours valide, mais aussi accessible et pertinente, ce qui n'est pas forcément le cas si cette dernière a été modifiée. Un tel système implique donc de surveiller l'ensemble des sources qui s'inter-référencent pour voir si leur contenu n'a pas été mis à jour ainsi que de vérifier que les références croisées sont toujours valides et peuvent être interprétées comme avant. De plus, l'ajout d'une source nécessite d'identifier les connexions entre ses entrées et celles des sources déjà intégrées au système, tâche assez complexe. En conclusion, l'approche navigationnelle n'est pas très adaptée à l'extension et au changement très fréquent des sources de données.

2.2.3.2.4 Systèmes d'intégration basée navigationnelle existants. **SRS** (Sequence Retrieval System) [Etzold 93] est un système d'intégration navigationnelle pour la biologie moléculaire. C'est à la fois un système de recherche d'informations basée sur des mots-clés et qui intègre plus de 140 sources de données, telles que Swiss-Prot et PubMed, mais aussi un serveur d'applications et d'outils pour l'analyse de données. SRS consiste en un ensemble d'informations concernant les sources de données : leur nom, leur URL mais aussi les attributs (au sens base de données) pour lesquels il existe des données dans la source et enfin les références croisées entre sources. Un composant du système se charge de parcourir et traiter les fichiers plats contenant du texte structuré ou les bases de données pour identifier les attributs et repérer les liens qui existent entre les entités de différentes sources. Deux types de liens entre sources sont ainsi identifiés : les hypertextes, c'est-à-dire les références croisées, et les indexés qui sont bidirectionnels et créés par SRS lorsque deux attributs issus de deux sources différentes sont similaires. Le système rend donc explicites des liens existant originellement entre différentes sources de manière implicite. L'interface d'interrogation est simple à utiliser : les utilisateurs se chargent de sélectionner les sources qu'ils souhaitent parcourir pour répondre à leurs requêtes qui sont limitées à des mots-clés ou des séquences nucléiques ainsi que des opérateurs booléens. Le résultat est fourni sous forme d'un ensemble d'attributs présents dans les sources et qui correspondent aux mots-clés formant la requête, les utilisateurs doivent ensuite se charger de parcourir les sources en fonction des informations qu'ils veulent trouver. En exploitant les liens indexés bidirectionnels, SRS propose également des liens entre des sources qui ne sont pourtant pas des références croisées directes, rendant ainsi les données résultats plus pertinentes et complètes puisqu'elles sont présentées dans le contexte d'autres données. Par exemple, le système permet de statuer

qu'une source S1 a un attribut tel que `Publication` qui peut être relié à un autre attribut de nom `Citation` dans une source S2.
Les informations décrivant les sources (c'est à dire concernant leur schéma) sont récupérées de manière manuelle. Par contre, l'identification des liens indexés permettant d'enrichir les seules références croisées existant physiquement entre les sources est faite semi-automatiquement grâce à un parseur* et donc de manière syntaxique au niveau *schéma*. En terme de maintenance, SRS est mis à jour quotidiennement de manière automatique en utilisant un programme qui vérifie si de nouvelles entrées sont apparues au sein des sources externes. Cependant, à notre connaissance, il n'y a pas de facilités pour ajouter une nouvelle source au sein du système ainsi que pour gérer d'éventuelles modifications des schémas des sources, touchant directement leurs attributs et donc les liens indexés bidirectionnels.

BioNavigation [Lacroix 05] était originellement une méthode de parcours, au travers de chemins, des références croisées existant entre les sources biomédicales, mais c'est désormais un système à part entière intégrant notamment OMIM et PubMed. Il permet aux utilisateurs de visualiser les sources intégrées au système et de naviguer au sein de celles-ci pour exprimer leurs requêtes et surtout pour les aider à déterminer quelles sources sauront au mieux y répondre. Deux modèles de représentation sont utilisés, une ontologie au niveau logique pour définir les aspects conceptuels et un graphe au niveau physique. L'ontologie permet de représenter les concepts scientifiques biomédicaux mais aussi les relations existant entre eux. Le graphe contient les sources de données ainsi que les applications et les références croisées existantes. Les deux niveaux sont reliés au travers des entités des sources qui sont associées aux concepts et des références croisées qui sont mises en correspondance avec les relations. BioNavigation propose de classer des chemins suivant trois critères :

- le nombre de paires d'enregistrements qui sont interconnectés par une relation existant entre deux sources données,
- le nombre d'enregistrements que contient la source cible,
- le coût d'évaluation, qui considère le traitement local nécessaire pour répondre à une requête mais aussi les temps d'accès aux différentes sources.

Les utilisateurs posent leurs requêtes sous forme d'une expression régulière, constituée de concepts et relations de l'ontologie, à partir de laquelle les chemins sont générés (en exploitant les éléments associés dans le graphe). L'interface fournit la liste des chemins pouvant répondre aux requêtes classés selon les critères choisis par les utilisateurs et visualisables sous forme de graphes.
Le schéma global à deux niveaux a été effectué manuellement tout comme le recueil d'informations au sujet des sources. Il n'y a donc aucune méthode proposée pour exploiter les informations situées aux niveaux *schéma* et *instances*. En revanche, la génération des chemins existant entre les sources intégrées est faite de manière automatique. D'autre part, BioNavigation n'aborde pas le thème de la maintenance des éléments constituant le système. Ajouter une source, par exemple, implique de nombreuses mises à jour au niveau physique puisqu'il est nécessaire d'identifier l'ensemble des références croisées d'une nouvelle source avec les autres. Par contre, cet ajout est facilité par le type de représentation sur deux niveaux puisque seul le graphe nécessite d'être modifié. Aucun changement n'est à effectuer dans l'ontologie à partir du moment où les concepts

et relations de la nouvelle source y sont déjà présents.

BioGuide[23] [Cohen Boulakia 05b] est un système basé sur l'approche navigationnelle dont l'objectif est de guider l'utilisateur dans le choix des sources et outils pertinents pour répondre à sa requête. Un premier système, DSS (Data Source Selection), a été développé par les mêmes membres du LRI[24] dans le cadre du projet européen HKIS [Cohen Boulakia 04] pour guider les utilisateurs de cette plate-forme dans la sélection de sources. Ce système avait été conçu en fonction des besoins des utilisateurs HKIS, des sources qu'ils utilisent, de leurs préférences et de leur propre stratégie d'interrogation. L'objectif de BioGuide a alors été de proposer un cadre entièrement générique, adaptable à différents types d'utilisateurs, ayant leurs propres entités biologiques d'intérêt, sources de données, préférences et stratégies d'interrogation. Pour ce faire, une étude des besoins a été menée à l'aide d'un questionnaire envoyé à une vingtaine de biologistes et bioinformaticiens. Il a permis d'identifier que ces derniers effectuent des requêtes impliquant des entités biologiques et des relations qui les lient plutôt que des noms de sources ou d'outils spécifiques, tout en voulant connaître l'origine des résultats. En réponse à ces besoins, BioGuide offre un support dans le processus d'interrogation en proposant une représentation sous forme de deux graphes : 1) un graphe du domaine biologique (entités biologiques et relations entre elles) qui joue le rôle de schéma global et 2) le graphe du réseau formé par les outils et les références croisées présents entre les sources. L'interface d'interrogation assiste les utilisateurs dans la conception de leurs requêtes. Ils peuvent interagir avec ces graphes et également les modifier s'ils le souhaitent. Le processus d'interrogation est le suivant : les utilisateurs se chargent d'indiquer les entités biologiques (et éventuellement les relations qu'elles partagent) au sein du graphe des entités. Ensuite, ils peuvent indiquer leur stratégie fondée sur certains critères : l'ordre dans lequel ils souhaitent parcourir les entités et l'exploitation qu'ils veulent faire des éventuelles entités reliées à celles de leurs requêtes. Enfin, les utilisateurs peuvent renseigner leurs préférences et indiquer un dernier critère de stratégie déterminant s'ils veulent utiliser une même source de données plusieurs fois (des préférences et une stratégie par défaut sont proposées). À partir de ces informations, le système génère automatiquement les chemins de sources ordonnés suivant les critères de préférences et de stratégie.
BioGuide peut être utilisé comme interface de requêtes à divers systèmes d'intégration. Par exemple, BioGuideSRS est un guide à la construction de requêtes pour le système SRS, ce qui permet aux utilisateurs d'exploiter l'interface, plus conviviale, de BioGuide. En revanche, les enjeux qui nous intéressent ne sont quasiment pas traités dans ce système. En effet, le schéma global (graphe des entités), le graphe des sources, la correspondance entre ces graphes et l'ensemble des méta-données ont été déterminés en fonction des réponses obtenues à l'étude des besoins, et donc de manière manuelle. Les aspects de maintenance du système sont peu abordés bien que chaque utilisateur ait la possibilité de modifier graphiquement les graphes et les préférences et d'enregistrer sa propre configuration du système sous forme d'un fichier XML qu'il pourra charger ultérieurement.

[23] http ://bioguide-project.net/
[24] Laboratoire de Recherche en Informatique http ://www.lri.fr/

Les systèmes d'intégration navigationnelle se focalisent sur la génération de chemins au travers de sources de données à partir des références croisées, voire d'autres liens supplémentaires identifiés entre elles. Les utilisateurs se chargent ensuite de parcourir ces chemins qui, en fonction des systèmes, sont entièrement construits (ils sont de la forme source d'origine - source(s) intermédiaire(s) - source cible) ou uniquement partiellement lorsque seules les sources d'entrée et cible sont fournies. Parmi les trois systèmes présentés, BioNavigation et BioGuide ont complété leur architecture d'une représentation des entités biologiques et leurs relations au niveau conceptuel qu'ils ont réalisée en fonction de leurs besoins. Cela permet de rattacher les méta-données des sources à un schéma global et de générer ainsi des chemins permettant de répondre aux requêtes des utilisateurs de manière automatique. SRS qui ne comporte que des méta-données et hyperliens entre sources ne peut ainsi offrir de fonctionnalités aussi puissantes que BioNavigation et BioGuide. Il se limite à proposer les sources d'entrée pouvant répondre aux requêtes des utilisateurs qui doivent eux-même parcourir les sources de données pour arriver à une réponse satisfaisante. Les enjeux étudiés dans cette thèse ne sont quasiment pas traités dans ce type de systèmes : les informations récupérées au sujet des sources, la conception du schéma global (quand il existe) et la maintenance des composants du système ne sont pas considérés ou uniquement réalisés de manière manuelle. En particulier, aucun des systèmes présentés n'exploitent les données des sources (niveau *instances*), leur utilisation n'apparaît qu'au moment de rendre les résultats à une requête. Le niveau *schéma* est exploité puisqu'il est absolument indispensable d'identifier des informations au sujet des schémas locaux ainsi que de les organiser dans un schéma global, si cela était nécessaire. Cependant, aucune proposition n'est faite pour automatiser, au moins partiellement, ces aspects pourtant cruciaux, notamment en terme de conception et de maintenance. Le point essentiel de ce type de systèmes se focalise sur la génération automatique de chemins entre sources pour répondre à des requêtes, avec parfois des méthodes avancées pour prendre en compte les critères et stratégies de recherche des utilisateurs.

2.2.3.3 Système de médiation

2.2.3.3.1 Architecture des systèmes de médiation. Cette approche [Wiederhold 92], également non matérialisée, consiste à laisser les données dans les sources et fournir un schéma global donnant une vue réconciliée, intégrée et virtuelle sur ces sources. Les utilisateurs posent leurs requêtes dans les termes du schéma global qui les redéfinit dans les termes des schémas des sources pertinentes. Il est donc nécessaire dans ce type de systèmes de définir une mise en correspondance entre les schémas locaux et le schéma global. Cet aspect est un point clé dans un tel système puisqu'il influe sur la reformulation de la requête. Les différents éléments constituant l'architecture des systèmes de médiation sont les suivants (Figure 2.7 page suivante) :

- les **schémas des sources** regroupent un certain nombre d'informations nécessaires pour décrire les sources, c'est-à-dire des méta-données comme leurs nom et URL d'interrogation mais aussi leurs entités et attributs auxquels des données sont associées. Des références croisées (voire même des références d'autres types) vers d'autres sources sont parfois présentes ;
- le **médiateur ou schéma global** joue le rôle d'interface entre les utilisateurs et les sources. Il regroupe l'ensemble des prédicats modélisant le domaine d'application du système de médiation et fournit donc un vocabulaire structuré servant de support à l'expression de

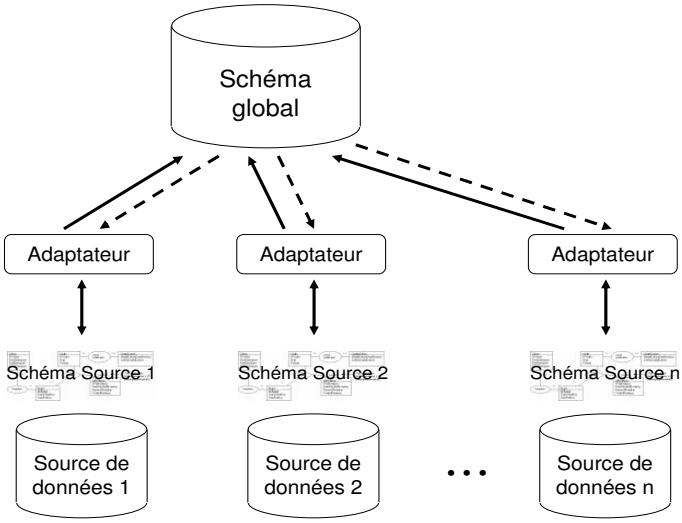


FIG. 2.7 – **Composants d'un système de médiation.** Les flèches pointillées représentent la transmission des requêtes aux adaptateurs. Les flèches unidirectionnelles pleines correspondent aux résultats récupérés dans les sources au travers de leur adaptateur qui les fournit au schéma global. Les flèches bidirectionnelles indiquent l'interaction existant entre les sources locales et les adaptateurs : l'envoi d'une requête dans le langage de la source associée et la récupération du résultat dans la source.

requêtes [Rousset 02]. Ainsi, les utilisateurs posent leurs requêtes dans les termes du médiateur qui les reformulent avant de les transmettre aux adaptateurs des sources identifiées comme pertinentes. Finalement, c'est à travers de ce médiateur qu'est réalisée la fusion des résultats obtenus à partir de chacune des sources avant de rendre une réponse homogène et globale aux utilisateurs ;

– les **adaptateurs** associés à chaque source permettent d'interroger la source qu'ils gèrent dans son langage spécifique afin d'y récupérer les données pertinentes. Ils fournissent en retour au médiateur les résultats obtenus dans les sources.

Comme présenté en détail dans [Lenzerini 02], modéliser la correspondance entre les schémas locaux et le schéma global est un point crucial. C'est ce qui permet de traiter une requête posée sur le schéma global par reformulation en un ensemble de requêtes dans les sources. La mise en correspondance vise à établir la connexion entre les éléments du schéma global et ceux des schémas locaux. Deux approches ont été proposées dans ce cadre : *Global-As-View* (GAV) où le schéma global est défini dans les termes des schémas des sources et *Local-As-View* (LAV) où le schéma global se veut indépendant des sources, et dans ce cas, les relations de correspondance existant entre le schéma global et les schémas des sources sont établies en définissant chaque source comme des vues du schéma global.

– L'approche GAV consiste à définir le schéma global en fonction des schémas des sources, donc dans les termes propres à ces dernières. Cela implique donc que les sources soient

connues et suffisamment stables. Cette approche favorise le traitement des requêtes puisque la ré-écriture des requêtes posées sur le schéma global se fait directement au travers des mises en correspondance préalablement définies et exprimant les éléments du schéma global dans les termes des schémas locaux. En revanche, l'expansion d'un système basé GAV pose des problèmes puisque l'ajout d'une nouvelle source peut avoir un impact sur la définition de certains éléments du schéma global.

– L'approche LAV est inverse ; ce sont les schémas des sources qui sont définis au moyen de termes du schéma global. Une bonne organisation et une stabilité du schéma global est nécessaire, les systèmes d'intégration utilisant une ontologie comme schéma global sont ainsi bien adaptés à LAV. Contrairement à l'approche GAV, le traitement des requêtes ici est complexe. En effet, les seules connaissances dont on dispose sur les données dans le schéma global sont uniquement accessibles au travers des vues représentant les sources, fournissant des informations partielles sur les données. Par contre, l'évolution de tels systèmes est particulièrement favorisée par l'approche LAV puisqu'ajouter une nouvelle source au système nécessite simplement de compléter les relations de correspondance de manière à représenter le nouveau schéma avec des termes du schéma global, sans aucune autre modification. La suppression se fait d'une manière aussi simple ; le schéma global n'est pas modifié, seules certaines relations sont supprimées, n'affectant pas le reste du système.

Une approche hybride a été proposée dans [Friedman 99]. *Global-Local-As-View* (GLAV) consiste à combiner la méthode de définition des relations de correspondance entre le schéma global et les schémas locaux de chacune des approches GAV et LAV pour en retirer une meilleure expressivité. Nous n'entrons pas plus dans les détails de cette approche.

2.2.3.3.2 Avantages des systèmes de médiation. Tout d'abord, la médiation a des avantages communs à l'approche navigationnelle puisqu'elles sont toutes deux non matérialisées. L'**autonomie locale est préservée**, ce qui évite de perturber les sources, chacune ayant un contrôle sur ses propres données et garantissant aux utilisateurs que les informations auxquelles ils accèdent sont purement issues de ces sources. De plus, ce type d'approche permet une **interrogation en temps réel des sources**, ce qui assure un accès à des données à jour. Finalement, en terme de maintenance, le fait de ne pas intégrer les données au niveau du système permet de **s'affranchir des problèmes de mises à jour et de synchronisation des données**.
Un avantage intéressant est celui de la **transparence** offerte aux utilisateurs lorsqu'ils effectuent leurs requêtes, ceux-ci ne sont en effet pas conscients de la distribution et de l'hétérogénéité des sources vu que l'on donne l'illusion aux utilisateurs d'interroger un système global et homogène. Ils n'ont ainsi pas à se charger de déterminer quelles sources ils vont devoir traverser ou interroger pour obtenir un résultat pertinent, c'est le système qui s'en occupe.
La définition d'un schéma global sous forme d'ontologie paraît particulièrement **bien adaptée pour l'intégration d'un ensemble de sources de données d'un même domaine d'application**. Celle-ci peut permettre d'assister les utilisateurs dans la conception de leurs requêtes [Stevens 00] mais surtout de faciliter la mise en correspondance entre le schéma global et les schémas locaux en exploitant les propriétés définies dans l'ontologie [Mork 05]. Cela est particu-

lièrement utile pour résoudre l'hétérogénéité sémantique existant entre les sources puisque, par exemple, des termes synonymes sont regroupés dans des concepts communs au sein de l'ontologie.
Enfin, cette approche est **efficace pour gérer l'évolution du système** si l'on s'astreint à utiliser la méthode LAV. Comme nous l'avons souligné ci-dessus, des ajouts et suppressions se font relativement simplement au niveau du schéma global. Les utilisateurs ont ainsi la possibilité de consulter des nouvelles connaissances quand de nouvelles sources intéressantes sont intégrées.

2.2.3.3.3 Inconvénients des systèmes de médiation. Des inconvénients communs aux différentes approches non matérialisées existent. Par exemple, **la fiabilité** du système n'est pas garantie à cause de l'interrogation en temps réel des sources. Si le serveur de celles-ci est momentanément indisponible, cela se reporte sur le système qui ne pourra pas récupérer des informations potentiellement pertinentes sur ce serveur. De plus, un trafic chargé sur le réseau peut ralentir un tel système.
La question de la **conception du système est particulièrement délicate**. Tout d'abord, quand les schémas décrivant les sources à intégrer sont inexploitables ou indisponibles, il est nécessaire de modéliser les données existant dans les sources (sous forme de schémas locaux), ce qui est une tâche difficile. Ensuite, la définition d'un schéma global permettant l'intégration de l'ensemble des informations nécessaires issues des schémas des sources est complexe. Enfin, la mise en correspondance entre ces différents éléments pose également des difficultés car elle nécessite généralement une intervention manuelle des concepteurs du système.
D'autre part, **l'évaluation des requêtes** (à moins de choisir une approche GAV) ainsi que **leur optimisation sont complexes** à réaliser, et ce d'autant plus si le nombre de sources est important. Comme détaillé précédemment, il n'est pas simple d'effectuer la traduction des requêtes posées sur le schéma global dans les termes des schémas locaux. Cette traduction peut être facilitée si les relations de mise en correspondance ont été bien établies et validées par un expert.
Enfin, l'évolution du système est plus simple à gérer avec l'approche LAV. Cependant, les **aspects de maintenance** représentent un travail plus conséquent. Par exemple, lors de l'intégration d'une nouvelle source, il est nécessaire de définir un adaptateur pour cette dernière puisque l'écriture de celui-ci est spécifique à chaque type de source [Vargas Solar 02].

2.2.3.3.4 Systèmes de médiation existants. Le projet **TAMBIS** (Transparent Access to Multiple Bioinformatics Information Sources) [Stevens 00] est un système de médiation développé dans le domaine biomédical dont l'objectif est de fournir aux utilisateurs une transparence entière des sources. Pour cela, l'ontologie TAO [Baker 99] a été définie dans la logique de description GRAIL[25] (GALEN Representation And Integration Language) et ses composants sont les suivants :

- des concepts primitifs représentent des entités biologiques telles que les protéines et acides nucléiques ainsi que leur localisation cellulaire, leurs fonctions et processus biologiques, leurs motifs et structures ;

[25] http ://www.openclinical.org/dld_galenGrail.html

- des rôles correspondant à des relations binaires entre concepts ;
- des concepts composites résultant de la composition d'un rôle et d'un concept (primitif ou composite) réalisée au moyen l'unique constructeur proposé par GRAIL : *WHICH* ;
- des relations de type *est-un* pour organiser les concepts sous la forme d'une hiérarchie.

L'ontologie, constituée d'environ 1800 concepts, est exprimée à l'aide d'une logique de description simple permettant de contrôler la création de nouveaux concepts et les classer au bon endroit suivant leurs descriptions. Le prototype du système intègre 15 sources, dont Swiss-Prot. Les utilisateurs posent leurs requêtes en explorant le modèle : ils choisissent les concepts qui les intéressent et peuvent en créer de nouveaux en combinant ces derniers avec des rôles. Pour chaque élément sélectionné, le système propose aux utilisateurs les éléments reliés à celui d'intérêt, permettant ainsi de spécialiser et d'enrichir la requête initiale. La sortie de cette phase est un plan de requêtes écrit dans le langage CPL (Collection Programming Language) [Buneman 95], base du système **BioKleisli** [Davidson 97] pour accéder à des sources biomédicales. Ce langage de requêtes permet de modéliser les différents types d'entrée fournis par les sources biomédicales. Au travers de pilotes (*drivers* en anglais), CPL traduit ainsi la requête initiale en sous-requêtes envoyées aux sources de données dans leur propre format puis récupère les résultats qu'il reconvertit dans son modèle.
Le schéma global a été créé à partir d'une ontologie déjà existante et complétée par des concepts identifiés dans les sources pour représenter l'ensemble des informations présentes dans celles-ci. Elle a donc été constituée semi-automatiquement. Les schémas locaux sont ceux définis dans BioKleisli au moyen du langage CPL. Les mises en correspondance des éléments de l'ontologie avec les adaptateurs se font au niveau *schéma* au travers d'une base de connaissance, appelée SSM (Sources and Services Model). Celle-ci permet d'identifier automatiquement les options disponibles pour évaluer la requête faite sur l'ontologie de manière à générer le programme CPL qui pourra répondre à la requête. Cette base de connaissances a cependant été générée manuellement. De plus, les programmes CPL permettent l'accès à de nombreuses sources biomédicales, mais ils sont pré-définis, ce qui impose au système d'intégrer des sources qui sont gérées par CPL et le rendant ainsi dépendant des évolutions prises en charge par CPL. Il n'y a aucune méthode se situant au niveau *instances* qui propose d'exploiter les données pour faciliter certains aspects de conception ou d'évolution du système. Enfin, bien que la représentation du schéma global en logique de description puisse être favorable à l'extension du système, aucune fonctionnalité de maintenance n'a été proposée pour gérer l'évolution de TAMBIS.

Le système **BACIIS** (Biological And Chemical Information Integration System) [Ben Miled 05] est basé sur l'approche médiateur et intègre sept sources biomédicales, comme OMIM et GenBank. Son architecture comporte les composants suivants :

- une ontologie, BAO (BACIIS Ontology), qui a le rôle de schéma global. Elle est indépendante du schéma des sources locales et consiste en trois classes : Objet, Relation et Propriété. Son modèle est tri-dimensionnel avec des structures hiérarchiques pour les Objets et Propriétés et un réseau de Relations, faisant l'association entre les concepts des deux autres classes (relation *a-pour-propriété*) et d'une même classe (relations *est-un-sous-ensemble-de* pour organiser les concepts en hiérarchies et autres pour relier les Objets, telles que *encode*). BAO est implémentée avec une logique de description étendue

avec un langage de « frames » (PowerLoom[26]). Elle permet de résoudre les hétérogénéités syntaxiques et sémantiques en regroupant des termes synonymes dans un unique concept. Ainsi, par exemple, si deux attributs issus de sources différentes sont associés au même concept dans BAO alors le concept commun sert de lien, on dit qu'il y a une correspondance sémantique entre les deux sources [Mahoui 05] ;

- un médiateur qui se charge du traitement des requêtes. Les utilisateurs construisent ces dernières à partir des concepts de l'ontologie et le médiateur les décompose en sous-requêtes. Il détermine ensuite quelles sources peuvent répondre en exploitant leurs schémas ; si l'entrée et la sortie de la sous-requête sont identiques aux entrée / sortie de la source, cette source est sélectionnée et le médiateur lance les adaptateurs associés. Il collecte ensuite les données récupérées par les différents adaptateurs, regroupe celles qui sont associées à une même classe (au travers de l'ontologie) et organise les résultats sous forme d'un graphe qui relie les concepts correspondant à la requête posée, formant ainsi l'ensemble des chemins différents permettant de mener à une réponse ;
- le schéma des sources consiste en des méta-données d'ordre général (e.g. nom de la source), les types de données acceptées en entrée décrits avec des concepts de BAO ainsi que les URLs et attributs associés pour interroger la source. Les types de données fournis en sortie ont les mêmes caractéristiques que ceux d'entrée mais à la place des URLs et des attributs, il y a des règles permettant d'extraire les données associées aux attributs présents dans les pages retournées par le site Web de la source ;
- les adaptateurs récupèrent les données dans les sources en exploitant les URLs d'interrogation des sources et les règles d'extraction définies dans les schémas associés. Il n'y a donc pas besoin d'un adaptateur par source mais plutôt par type de format de sortie offert par les sources (HTML, XML, texte) pour parcourir les pages résultats et y récupérer les informations voulues.

Le schéma global de BACIIS a été développé manuellement mais il est malgré tout extensible et flexible. En effet, BAO est indépendante des schémas des sources et évolue uniquement quand le domaine biologique subit des modifications importantes (comme la découverte d'un nouveau concept tel que `Pathway`, ce qui arrive très rarement), plutôt qu'à chaque mise à jour des sources. De plus, son modèle tri-dimensionnel facilite la prise en charge des modifications puisque, par exemple, un changement d'une Propriété n'affecte que cette hiérarchie mais pas celle des Objets. L'approche adoptée pour les adaptateurs évite un changement systématique d'un adaptateur si une source est modifiée. Il est, en effet, uniquement nécessaire de définir un nouvel adaptateur si un format de sortie nouveau est utilisé par une source déjà existante ou nouvellement intégrée. Cependant, les méta-données sur les sources (et donc ce qui sert de schéma local) n'ont pas été récupérées automatiquement. Enfin, les règles d'extraction des données des sources initialement générées manuellement sont désormais induites semi-automatiquement [Ben-Miled 04]. Un algorithme a été développé de manière à parcourir un certain nombre de pages Web fournies par les sources afin d'identifier des attributs correspondant à des concepts de BAO et créer les règles d'extraction à partir de ces attributs. Cette méthode, située au niveau *instances*, illustre l'intérêt d'exploiter les données mêmes des sources pour faciliter la définition de leur schéma.

[26] http ://www.isi.edu/isd/LOOM/PowerLoom/

Le système **ONTOFUSION** [Alonso-Calvo 06] est un système de médiation intégrant des sources de données biomédicales de différents types : publiques, telles que PubMed et PDB (Protein Data Bank) [Kouranov 06], mais aussi des bases privées et des bio-ontologies disponibles en ligne mais qui sont installées localement, comme l'UMLS et GO. Ses composants sont les suivants :

- quatre fichiers XML par source de données publique décrivent l'information nécessaire pour la représenter et l'interroger. Le premier contient le schéma physique de la source, le second décrit la manière dont il faut interroger la source (URL, paramètres, etc), les deux autres décrivent la structure des pages résultats obtenus à partir du site Web de la source. Il faut noter que les sources de données privées ne nécessitent pas ces fichiers puisque ONTOFUSION récupère directement leur schéma et les interroge au moyen de langages simples tels que SQL ;
- des ontologies permettant chacune de représenter l'ensemble des concepts d'un même domaine (déjà existantes ou nécessitant d'être créées) dont le possible recouvrement est résolu manuellement ;
- des schémas virtuels, construits à partir des schémas physiques, sont en fait des ontologies décrivant la structure de l'information contenue dans les sources à un niveau conceptuel. Ceux-ci sont construits par les administrateurs du système qui disposent d'une interface les aidant à réaliser cette tâche. Ils doivent mettre en correspondance les éléments du schéma physique de chaque source avec les concepts de l'ontologie de domaine appropriée ;
- des schémas virtuels unifiés fusionnent les schémas virtuels associés à une même ontologie de domaine. Tous les schémas virtuels sont définis au moyen du langage DAML+OIL ;
- un module médiateur se charge d'interroger et d'accéder aux différents schémas virtuels unifiés au travers d'adaptateurs. Ces derniers exploitent les fichiers XML associés à chaque source. Les résultats collectés par les adaptateurs sont d'abord fusionnés puis retournés comme des instances de chaque ontologie de domaine impliquée dans la requête ;
- l'interface utilisateur a été créée sous la forme d'un éditeur d'ontologies où les utilisateurs peuvent explorer et naviguer au travers des schémas virtuels des sources, et même choisir les éléments précis auxquels ils souhaitent accéder (grâce au fichier XML contenant le schéma physique de la source). Ensuite, l'interface génère une requête dans le langage RDQL[27] (RDF Data Query Language) et l'envoie aux adaptateurs appropriés.

Ce système offre des fonctionnalités intéressantes, notamment parce qu'il utilise des technologies du Web sémantique pour représenter les sources sous forme d'ontologies. De plus, des interfaces ont été développées pour assister les différentes tâches laissées aux administrateurs. Cependant, ONTOFUSION ne traite quasiment aucune des tâches qui nous intéressent plus particulièrement de manière automatique. Tout d'abord, il n'y a pas de réel schéma global, ce sont les ontologies de domaine qui jouent ce rôle et celles-ci sont censées déjà exister ou être définies par un administrateur. De plus, l'ensemble des informations récupérées au sujet des sources (contenu des quatre fichiers XML) sont collectées manuellement. Ce sont aussi les administrateurs qui doivent se charger de mettre en correspondance les schémas virtuels avec les ontologies de domaine. Seule la phase d'unification des schémas est effectuée automatiquement, grâce à un algorithme qui regroupe des concepts identiques ou liés hiérarchiquement tout en unifiant également l'ensemble

[27] http ://www.w3.org/Submission/RDQL/

des attributs de ces concepts issus de différents schémas virtuels. Cette méthode se situe au niveau *schéma* mais rien n'est proposé pour exploiter le niveau *instances*. D'autre part, RDQL est un langage de requêtes pour RDF qui ne tient pas compte de la sémantique de RDF Schema et ne permet donc pas d'exploiter la richesse de DAML+OIL dans lequel sont décrits les schémas virtuels. Enfin, pour ce qui est de la maintenance, seule une interface est fournie aux administrateurs pour effectuer l'ajout d'une nouvelle source mais ce sont eux qui restent responsables des lourdes tâches pré-citées.

SEMEDA (SEmantic MEtaDAtabase) [Köhler 02] est un système de médiation intégrant six sources de données dont Swiss-Prot et KEGG. Différents composants constituent SEMEDA :

- une ontologie définie dans un langage proche de RDF joue le rôle de schéma global. Elle contient un ensemble de concepts reliés par des relations binaires. Chaque concept est une entité bien définie avec un sens unique, des propriétés (telles que son nom), une description et un identifiant. Chaque relation est définie par sa sémantique (*est-un*, *partie-tout*, etc) et ses propriétés algébriques (transitivité, symétrie, etc). Cette ontologie de haut niveau est spécifique pour l'intégration et sert à définir les sources au niveau *schéma* au moyen des concepts racines suivants : nom, identifiant, description et propriété. Les attributs des sources sont reliés à un ou plusieurs concepts de cette ontologie de haut niveau, chaque concept peut donc représenter des attributs de sources distinctes ;
- une base de données relationnelle permet de stocker localement les méta-données des sources (URL, nom, etc). S'y trouvent également des bio-ontologies et vocabulaires contrôlés très larges déjà existants (par exemple, GO) qui servent à unifier les données présentes dans les diverses sources. L'idée est d'étudier les valeurs associées aux attributs afin de préciser ces derniers qui sont souvent peu explicites dans les sources. Cela permet de relier des attributs de noms différents dans des sources distinctes, et d'unifier leurs valeurs. Ces informations donnent des définitions sémantiques concernant les sources ;
- des interfaces utilisateurs dont les fonctionnalités dépendent des droits accordés à chacun. L'interface administrateur permet d'importer des méta-données d'une source, générer leurs schémas et ajouter des nouveaux concepts à l'ontologie de haut niveau. L'interface utilisateur offre la possibilité de parcourir et d'afficher les informations sur les méta-données et définitions sémantiques des sources. À partir des concepts choisis dans l'ontologie, les utilisateurs sont guidés vers les attributs pertinents des diverses sources pour construire leur requête. Un formulaire est alors généré automatiquement pour interroger la source dont l'attribut a été choisi par les utilisateurs. C'est le nom du concept présent dans l'ontologie (associé à l'attribut) qui est présenté aux utilisateurs car il est souvent plus explicite que celui de la source. Le type de valeurs qui correspond à ce concept est également précisé quand il est disponible. De plus, si un concept de haut niveau est impliqué dans une requête et que ce concept a plusieurs concepts plus spécifiques qui sont associés à des attributs dans des sources, l'interface propose tous ces concepts (plus spécifiques) aux utilisateurs, permettant ainsi d'étendre leur requête. C'est un outil indépendant, BioDataServer [Freier 02], qui se charge de décomposer les requêtes en sous-requêtes qu'il pose aux différentes sources afin d'y récupérer automatiquement les données pertinentes. Il joue ainsi à la fois le rôle du médiateur et des adaptateurs.

Le schéma global du système a été créé manuellement. Ce sont les fournisseurs des sources de données qui doivent se charger de définir leurs méta-données et donc de mettre en correspondance les bons concepts dans l'ontologie de haut niveau avec leurs attributs. Cet aspect est trop contraignant car il ne semble pas logique de laisser cette lourde tâche aux fournisseurs de sources, pour qui l'intérêt d'intégrer leur source à SEMEDA n'est pas forcément évident. L'approche, située au niveau *instances*, permettant de préciser les attributs des sources en identifiant le type de leurs valeurs est particulièrement intéressante mais n'a pas été implémentée, même de manière manuelle. C'est aussi manuellement qu'un type peut être défini, au niveau *schéma* cette fois, pour les attributs des sources, au travers des bio-ontologies et vocabulaires contrôlés intégrés à SEMEDA. En terme d'évolution du système, les fournisseurs peuvent ajouter, éditer et supprimer des relations et concepts mais ces actions ne sont répercutées que si les administrateurs les valident. Ainsi donc, même si des interfaces facilitent ces tâches, l'intervention humaine reste indispensable.

Nous ne donnons pas une liste exhaustive de l'ensemble des systèmes de médiation existants. Nous nous attachons à identifier les familles principales, à souligner leurs spécificités et à en citer les principaux représentants. Le système **INDUS** [Reinoso-Castillo 03], qui intègre de l'information distribuée et hétérogène sémantiquement pour l'acquisition de connaissances et leur intégration, aborde la question des mises en correspondance sémantiques entre des sources de données hétérogènes. Il considère notamment la possibilité de les résoudre au niveau *schéma* au travers des attributs des sources mais également, et c'est cela qui est intéressant, au niveau *instances* grâce aux valeurs associées aux attributs. Malheureusement, comme dans SEMEDA, ce sont les utilisateurs qui ont la charge de définir ces correspondances ainsi que d'identifier les attributs au sein des sources. **BioKleisli** [Davidson 97], cité précédemment dans la description de TAMBIS [Stevens 00], est quant à lui un système de médiation de bas niveau puisqu'il ne dispose pas d'un réel schéma global. Il n'y a donc pas de facilités offertes aux utilisateurs pour interroger le système simplement. Il faut en effet qu'ils aient une bonne connaissance de CPL (langage re-définissant la requête principale en sous-requêtes adaptées aux sources pouvant y répondre) et aussi des structures et schémas des sources intégrées.

Parmi les systèmes présentés, il n'y a pas vraiment de systèmes basés GAV ou LAV dans le domaine biomédical (tout du moins, aucun ne se situe explicitement par rapport à l'une ou l'autre des approches), certainement parce que la nécessité d'intégration a été antérieure à leur définition. En revanche, de nombreuses fonctionnalités ont été implémentées. On distingue ainsi des méthodes cherchant à faciliter le traitement des requêtes par l'utilisation d'ontologies formelles dans TAMBIS et BACIIS. Certains systèmes visent à automatiser une partie de la conception du système, soit en définissant préalablement des règles manuellement et qui sont ensuite appliquées de manière automatique (BACIIS), soit en proposant des interfaces visant à assister les experts dans leur travail d'unification et de mise en correspondance d'informations issues des sources avec les éléments du schéma global (ONTOFUSION et SEMEDA). En terme de maintenance, principalement deux cas de figure existent. D'une part, le système est défini de la manière la plus flexible possible, ce qui permet de minimiser les traitements, lors de modifications, ajouts ou suppressions de sources ou d'attributs, mais qui sont malgré tout à effectuer

manuellement (BACIIS et BioMediator). D'autre part, des interfaces sont proposées aux administrateurs du système, ayant cependant encore la charge de tâches manuelles beaucoup trop lourdes (ONTOFUSION et SEMEDA). Ces efforts ne sont donc pas suffisants.

Cependant, ce sont les systèmes de médiation qui abordent le mieux les différents points de conception et d'évolution qui nous intéressent. En effet, les systèmes existants proposent tout d'abord un schéma global dont la représentation est relativement bien faite, la plus avancée étant celle qui est réalisée de manière formelle. Parallèlement, ces systèmes ayant pour principe de laisser les sources autonomes, ils cherchent à disposer d'une description de celles-ci afin de pouvoir mettre en correspondance les éléments constituant ce schéma avec ceux du schéma global. Enfin, ces systèmes sont flexibles puisque l'ajout ou la modification d'une source ne nécessite généralement que de décrire (ou mettre à jour) le schéma de cette source pour ensuite définir les correspondances permettant de représenter les éléments de ce schéma local dans les termes du schéma global. L'approche d'intégration basée médiateur est celle qui a été la plus souvent mise en œuvre dans le domaine biomédical ces dernières années, sans doute grâce au nombre très important de systèmes suivant cette approche dans le domaine informatique. Ces derniers, souvent construits selon des méthodes avancées, sont de bons exemples quant aux possibilités offertes par cette approche et qui paraissent particulièrement bien adaptées aux besoins identifiés dans notre domaine. Picsel par exemple est un système suivant cette approche qui a été testé avec succès pour le domaine du tourisme [Rousset 02]. Cependant, le domaine biomédical est particulièrement spécifique, notamment par la variété de ses données (il faut, par exemple, pouvoir manipuler aussi bien des séquences que des structures 3D de protéines), l'hétérogénéité multiple des sources de données et même leurs capacités différentes d'interrogation. Ces caractéristiques nécessitent de mettre en œuvre un système pouvant y répondre efficacement.

2.2.3.4 Systèmes hybrides

D'autres types de systèmes se situant à l'interface entre les approches pré-citées existent. Ces travaux, dits hybrides, espèrent ainsi bénéficier des avantages de chacune et s'affranchir de leurs inconvénients en même temps. Cela peut malgré tout soulever d'autres problèmes, notamment en terme de conception. En effet, les éléments constituant les systèmes hybrides sont plus nombreux, compliquant ainsi encore plus leur création mais aussi la gestion de leur évolution.

Pour illustrer ce type d'approche, nous présentons ici **BioMediator** (originellement BioSeek [Mork 01]) qui est un système d'intégration de données hybride médiateur-navigationnel développé pour le domaine de la biologie moléculaire. Il regroupe sept sources, telles que OMIM et GO et repose sur une base de connaissances (Source Knowledge Base - SKB) représentée en frames au moyen de l'éditeur Protégé[28] et qui contient :

– le schéma médiateur où sont modélisées uniquement les entités biologiques (ex. : *Genes*, *Proteins*) partagées par les sources intégrées. Elles sont organisées sous la forme d'une hiérarchie de classes et d'une hiérarchie de propriétés définissant les relations entre les entités (*associated-with*, *codes-for*) ;

[28] http ://protege.stanford.edu/

– l'ontologie du système qui regroupe des méta-données concernant l'ensemble des sources intégrées, comme les éléments du schéma médiateur contenus dans celles-ci, les références croisées existant entre les sources mais aussi les annotations décrivant comment ces références sont établies et maintenues (manuellement ou générées automatiquement). Elle contient également les règles de correspondance entre le schéma médiateur et les schémas locaux [Shaker 02]. On distingue deux types de règles. Les règles de *transmission* (Forward Mapping Rules ou FMR) qui consistent à ré-écrire la requête sous forme d'URL envoyée à une source de données avec les paramètres et entités appropriés. Les autres règles sont dites *inverses* (Reverse Mapping Rules ou RMR). Elles permettent d'identifier de nouvelles entités présentes dans les sources et de les ajouter au schéma médiateur, ou simplement d'associer les entités des sources avec celles qui sont déjà présentes dans le schéma médiateur. Elles se chargent aussi d'établir des relations entre ces entités ainsi que de récupérer les données leur correspondant.

L'autre composant de BioMediator est le processeur de requêtes qui fournit une API* (*Application Program Interface*) pour gérer et lancer les requêtes posées en utilisant les éléments du schéma médiateur. Plus précisément, il est constitué des éléments suivants :

– le méta-adaptateur qui traduit les requêtes dans les termes des sources en utilisant les règles de mise en correspondance de type *FMR* présentes dans la SKB et les transmet aux adaptateurs. Dans l'autre sens, il re-traduit les résultats fournis par les adaptateurs dans les termes du schéma médiateur en exploitant les règles *RMR* ;
– les adaptateurs (un par source) posent les requêtes en temps réel dans les termes des sources et traduisent en XML le résultat fourni par les sources pour le transmettre au méta-adaptateur ;
– le résultat est de la forme suivante : les données retrouvées sont organisées en un graphe (réseau sémantique) dont les nœuds correspondent à des instances des entités définies dans la hiérarchie de classes. Chaque flèche est une instance d'un élément de la hiérarchie de propriétés. De plus, la requête des utilisateurs peut être étendue automatiquement au moyen d'entités voisines de celle(s) d'intérêt et qui sont reliées à cette (ces) dernière(s) par des propriétés. Enfin, les méta-données associées aux relations existant entre les sources permettent d'appliquer des critères de préférence que les utilisateurs peuvent préciser quand ils interrogent le système. Les résultats vérifiant au mieux ces critères sont retrouvés automatiquement.

BioMediator est un système hybride dans le sens où ses composants sont ceux d'un système de médiation (médiateur, méta-adaptateurs et adaptateurs). Il offre cependant la possibilité de définir des critères de préférence pour personnaliser les résultats et présente ces derniers sous forme de graphes « instances » du schéma médiateur. C'est sur ce point qu'il s'inscrit dans l'approche navigationnelle.

Le schéma médiateur a été initialement conçu manuellement mais il peut être étendu au travers des règles *RMR* comme décrit précédemment. Les méta-données concernant les sources ont été recueillies manuellement. La mise en correspondance entre les schémas des sources et le schéma médiateur est établie automatiquement en utilisant les règles de correspondance appliquées par le méta-adaptateur. Par contre, ces règles ont été préalablement définies par un expert. Aucun changement n'est nécessaire sur le méta-adaptateur quand l'ontologie et son

schéma médiateur changent ou quand la sortie d'une source est modifiée car il suffit de modifier les règles. En revanche, les adaptateurs ainsi que les informations présentes dans l'ontologie au sujet des sources (notamment lors de l'intégration d'une nouvelle source, les références croisées de chaque source déjà intégrée peuvent nécessiter d'être modifiées) doivent être mis à jour et ce, de manière manuelle. On peut donc en déduire que l'ajout de nouvelles sources en particulier est relativement difficile à effectuer.

2.2.3.5 Conclusion - Tableau récapitulatif

Le type d'approche d'intégration à suivre n'est pas le même en fonction des objectifs pour lesquels les utilisateurs souhaitent disposer d'un système d'intégration. En effet, les utilisateurs jouent un rôle important dans le choix de l'approche à utiliser, et plus précisément leurs connaissances et le type d'informations auxquelles ils espèrent accéder grâce au système. Les biologistes et les médecins, en particulier, ne doivent pas connaître les détails concernant, par exemple, l'implémentation des sources de données intégrées ou encore leur localisation physique [Karp 96], cela noierait les données qui sont réellement pertinentes pour eux au milieu d'informations sans intérêt par rapport à leurs besoins. Un certain niveau de transparence est donc requis, de manière à bien différencier ce à quoi les utilisateurs n'ont pas besoin d'accéder du minimum qu'ils doivent connaître des sources, notamment l'origine des données résultats et la qualité des données qui sont des aspects très importants pour eux. Ainsi, pour des utilisateurs non experts en représentation des connaissances et plus généralement en informatique, des systèmes tels que BioKleisli où ce sont les utilisateurs qui doivent sélectionner les sources appropriées en fonction de leurs contenu et schéma afin d'obtenir les résultats à leur question, ne sont pas adaptés. Par contre, il est indispensable qu'un système d'intégration soit flexible pour que les utilisateurs puissent être libres de choisir quelles sources ils souhaitent utiliser mais sans qu'ils aient à connaître et / ou à gérer des détails informatiques ou des choix de représentation faits par les concepteurs au sujet des sources.

Les limites des systèmes d'intégration actuels sont les suivantes. Notre premier constat est que les différentes étapes de conception des systèmes d'intégration existants nécessitent encore un lourd travail manuel et **il apparaît indispensable que certaines phases soient automatisées**, ou au minimum semi-automatisées [Karp 96]. L'existence d'outils ou d'algorithmes facilitant ce type de tâches permettrait non seulement d'assister une création souvent complexe des systèmes d'intégration et dans un second temps, leur maintenance. Nous reprenons les différents enjeux sur lesquels nous avons focalisé notre étude des différents systèmes d'intégration que nous avons détaillés dans ce chapitre : l'acquisition des schémas locaux, la définition d'un schéma global, l'utilisation de méthodes situées au niveau *instances* pour compléter celles qui exploitent généralement le niveau *schéma* afin d'identifier les correspondances entre les schémas locaux et le schéma global et enfin la gestion de l'évolution du système.

D'abord, il est nécessaire de disposer d'une description suffisamment détaillée de chaque source de données, afin de créer le schéma local correspondant s'il n'est pas accessible. Sur ce point, seul BACIIS [Ben Miled 05] propose des solutions pour automatiser cette acquisition en partie (TAMBIS [Stevens 00] dispose des schémas locaux de manière automatique car ils étaient

déjà décrits en CPL ; ce ne sont pas les concepteurs de ce système qui se sont chargés de leur création). Il faut de plus noter que les concepteurs n'ont pas développé cette méthode au moment de la construction du système mais dans un deuxième temps, lorsqu'ils ont voulu faciliter sa gestion. Cela confirme donc bien notre hypothèse comme quoi le fait de pouvoir **automatiser la création de schémas locaux sera utile non seulement à la conception de notre système mais aussi à sa maintenance**.

La création du schéma global est la tâche qui est souvent la mieux abordée dans les différents systèmes. En effet, l'importance de ce composant a été largement soulignée et démontrée et les concepteurs de systèmes d'intégration ont généralement effectué cette étape correctement. Ceci dit, **on peut déplorer que peu d'entre eux aient tenté de créer une partie de leur schéma global automatiquement** car c'est une tâche particulièrement lourde et pénible. Cela peut s'expliquer par le fait que les concepteurs préfèrent décrire eux-mêmes le schéma global afin d'être certains qu'il soit bien formé et permette de représenter l'ensemble des informations nécessaires concernant les sources à intégrer. Pourtant, dans Xylème [Xyleme 01], le schéma global est construit semi-automatiquement à partir des schémas locaux. Cependant, les schémas des sources étant décrits en XML, cette tâche est plus simple dans ce système. En revanche, TAO [Baker 99] qui est l'ontologie servant de schéma global au système TAMBIS a été créée semi-automatiquement à partir d'une ontologie pré-existante et complétée en fonction des besoins du système. Cela constitue une solution intéressante dans la perspective d'automatiser cette phase. En effet, la ré-utilisation d'une ontologie ou d'un système terminologique suffisamment bien formé peut être un bonne base pour la création automatisée du schéma global. Les connaissances déjà existantes (et donc validées) peuvent être utilisées telles quelles et elles pourront être complétées si elles ne suffisent pas pour représenter les éléments présents dans les schémas locaux.

Concernant les méthodes proposées pour **exploiter les informations situées au niveau *instances***, les objectifs pour lesquels elles sont utilisées diffèrent d'une approche à l'autre. En effet, les entrepôts ayant comme spécificité de suivre une approche matérialisée, il est fréquent que ces systèmes mettent en œuvre des techniques au niveau *instances*. Dans ce cas, le but recherché est de ne pas se limiter à rapatrier les données mais aussi d'améliorer le système en les modifiant, annotant, corrigeant de manière à les personnaliser. Le cas de Xylème est à part car les auteurs, n'ayant pas développé un système spécifique au domaine biomédical, n'ont pas focalisé leurs efforts sur l'exploitation des données, mais plutôt sur la conception. C'est ce dernier point que les entrepôts ont tendance à négliger ; aucun système existant n'a développé de techniques basées sur les données pour faciliter la création d'entrepôts. Ainsi, leur utilisation du niveau *instances* n'est en réalité pas mise en œuvre pour compléter le niveau *schéma*, ils se contentent de traiter les données. Les systèmes suivant l'approche navigationnelle n'ont, à notre connaissance, pas développé de méthodes utilisant des informations situées au niveau *instances*. En revanche, parmi les systèmes de médiation existants, SEMEDA [Köhler 02] et INDUS [Reinoso-Castillo 03] ont introduit l'intérêt que pourrait avoir l'extraction d'informations à partir des données associées aux éléments des schémas des sources, mais rien n'a été implémenté. En revanche, BACIIS propose une méthode semi-automatique visant à extraire les attributs des sources de données qui se trouvent dans l'ontologie BAO jouant le rôle de schéma global. Cette

approche est intéressante dans la mesure où elle cherche à faciliter la gestion de l'évolution des sources de données intégrées au système et plus spécifiquement de leur schéma. Cependant, BACIIS ne tire pas profit des données associées aux attributs qu'il extraits et n'exploite donc pas tout l'intérêt que peut apporter le niveau *instances*. En effet, l'utilisation des valeurs permettraient par exemple de typer les attributs, ce qui enrichirait le schéma local de la source dont ils sont issus. L'exploitation au niveau *instances* dans le but de gérer la conception de systèmes d'intégration ainsi que leur maintenance est donc, pour le moment, relativement peu développée.

Enfin, les **efforts réalisés pour gérer la maintenance des systèmes** sont inégaux suivant l'approche d'intégration mise en œuvre. Ce sont les systèmes d'intégration basée entrepôt qui sont les plus avancés dans ce cadre. Cela est du au fait que ce type de systèmes nécessite encore plus que les autres d'être mis à jour régulièrement. En effet, comme ils intègrent les données issues des sources localement, il est nécessaire de vérifier très souvent s'il y a eu des modifications dans les sources d'origine pour les répercuter au plus vite sur les données présentes dans l'entrepôt. Les autres approches, qui sont virtuelles, n'ont pas ce problème vu que l'interrogation des sources est faite dynamiquement au moment où les utilisateurs effectuent leurs requêtes. Pourtant, il reste d'autres tâches à gérer pour maintenir le système à jour. Ce ne sont pas seulement les données qui changent mais aussi les caractéristiques des sources contenant ces données, modifications qui se répercutent indirectement sur le schéma global. En l'occurrence, leur schéma ou encore leur URL d'interrogation doivent être vérifiés régulièrement. Dans les systèmes dont l'intégration n'est pas basée sur la médiation, des efforts ont été faits pour gérer ces aspects de manière automatisée, soit totalement dans Xylème et SRS [Etzold 93], soit partiellement dans GUS [Davidson 01], Biozon [Birkland 06] et BioGuide [Cohen Boulakia 05b]. Par aillaurs, certains systèmes de médiation proposent des interfaces pour faciliter ces tâches aux concepteurs (ONTOFUSION [Alonso-Calvo 06] et SEMEDA). D'autres ont implémenté un schéma global qui est flexible et simplifie donc ces tâches (BACIIS et BioMediator [Mork 01]). Le problème est que, dans ces deux derniers cas de figure, le travail reste entièrement manuel, ce qui n'est pas satisfaisant.

Avant de conclure, nous rappelons que, par souci de simplification, seuls les systèmes d'intégration centralisés ont été abordés. Ceci étant, il existe des **approches décentralisées**, notamment des entrepôts de données présentant plusieurs « intégrateurs » mais aussi des systèmes de médiation où il existe plusieurs médiateurs, chacun représentant un ensemble de sources ayant des caractéristiques proches. On parle de technologie « pair à pair » (ou Peer-to-Peer ou P2P en anglais) [Halevy 03] et celle-ci offre des fonctionnalités intéressantes dans le cadre du domaine biomédical. En particulier, il serait utile de pouvoir définir un schéma médiateur par sous-domaine au lieu d'en développer un seul pour représenter l'ensemble des informations du domaine plus général [Louie 06]. En effet, il peut être difficile de regrouper les différents types de données dans un schéma unique et c'est d'autant plus complexe que les sources biomédicales sont hétérogènes à divers niveaux. D'autre part, l'entretien d'un schéma global unique et complexe peut se révéler de plus en plus délicat au fur et à mesure que de nouvelles sources sont ajoutées au système. Ainsi, disposer de plusieurs schémas simples (et spécifiques à un sous-domaine ou à un type de données) permettrait de gérer et de compléter uniquement celui qui va servir à

représenter les éléments des sources à intégrer.

Cependant, la recherche d'informations dans des systèmes suivant cette approche devient plus complexe et nécessite des langages de requêtes plus riches. De plus, un niveau supplémentaire est nécessaire pour mutualiser les informations issues de ces schémas distribués, ce qui constitue une tâche additionnelle assez lourde en terme d'implémentation. Dans ce cadre, l'approche décentralisée peut tirer profit des diverses approches centralisées existantes, ainsi que des algorithmes, outils et méthodes utilisés pour les mettre en œuvre [Hacid 04]. En particulier, les travaux menés sur la mise en correspondance de schémas, que nous présentons dans la section suivante, et proposant une automatisation partielle pour des domaines d'application particuliers pourront guider ces travaux. Il n'y a actuellement pas de système suivant une approche décentralisée développé dans le domaine biomédical mais ce pourrait être une solution adaptée aux besoins de ce domaine [Louie 06]. Nous n'entrons pas plus dans les détails de ce type de systèmes, considérant que les approches plus classiques nécessitent déjà d'être plus abouties pour pouvoir passer à un niveau supérieur de complexité en terme d'intégration.

Le récapitulatif des systèmes d'intégration avec les différentes caractéristiques les concernant sont donnés dans le tableau 2.1 page suivante. Pour chacun, il indique l'approche d'intégration suivie, le format et le moyen de conception des schémas locaux et du schéma global. La manière de réaliser les mises en correspondance entre schémas locaux et schéma global est caractérisée par son automatisation. Enfin, les méthodes déployées au niveau *schéma* puis *instances* sont brièvement détaillées avant de préciser les techniques proposées par chaque système pour gérer son évolution.

TAB. 2.1 – Tableau récapitulatif des caractéristiques des systèmes d'intégration présentés

Nom du système	**Approche d'intégration**	**Schémas locaux**	**Schéma global**	**Mise en correspondance**	**Méthodes au niveau *schéma***	**Méthodes au niveau *instances***	**Aspect maintenance**
Xylème	Entrepôt	XML / manuel	DTD / semi-automatique	Semi-automatique	Fusion des DTDs associées aux schémas XML locaux / automatique	-	Renommage ou suppression d'attributs du schéma global / automatique
GUS	Entrepôt	Relationnel / manuel	Relationnel / manuelle + exploitation de schémas existants	Manuelle	Association des éléments des attributs des schémas locaux au sein du schéma global / manuel	Intégration, nettoyage et correction des données	Historique des données modifiées, outil facilitant ajout et suppression d'attributs / semi-automatique
Gedaw	Entrepôt	DTD / manuel	Objet / manuel	Manuelle	Association des éléments des DTDs avec les entités du schéma global / manuel	Unification et filtrage des données / automatique	-
Biozon	Entrepôt	Existant / manuel	Hiérarchie de classes + graphe / manuel	Manuelle	Association des éléments des attributs des schémas locaux au sein du schéma global / manuel	Intégration des données + données dérivées pour éliminer la redondance / automatique	Ajout, modification et suppression de nœuds dans le graphe / semi-automatique

Nom du système	**Approche d'intégration**	**Schémas locaux**	**Schéma global**	**Mise en correspondance**	**Méthodes au niveau *schéma***	**Méthodes au niveau *instances***	**Aspect maintenance**
SRS	Navigationnelle	Méta-données / manuel	-	-	Indexation des attributs des sources pour faciliter l'interrogation du système / automatique	-	Mise à jour quotidienne si modification dans les sources locales / automatique
BioNavigation	Navigationnelle	Méta-données / manuel	Ontologie + graphe / manuel	Manuelle	Représentation des méta-données au sein du schéma global / manuel	-	-
BioGuide	Navigationnelle	Méta-données / manuel	Réseau sémantique / manuel	Manuelle	Représentation des méta-données au sein du schéma global / manuel	-	Interface pour modifier les graphes
TAMBIS	Médiateur	CPL (par BioKleisli) / semi-automatique	Ontologie en GRAIL (DL) / semi-automatique	Manuelle	Association des éléments des attributs des schémas locaux au sein du schéma global / manuel	-	-
BACIIS	Médiateur	Méta-données / manuel	Ontologie en PowerLoom / manuel	Semi-automatique	Résolution des hétérogénéités syntaxiques et sémantiques d'attributs différents en les représentant par un concept unique dans l'ontologie / automatique	Extraction des attributs (certaines méta-données) des sources / semi-automatique	Système flexible mais nécessité de faire les mises à jour (rarement nécessaires) manuellement

Nom du système	**Approche d'intégration**	**Schémas locaux**	**Schéma global**	**Mise en correspondance**	**Méthodes au niveau *schéma***	**Méthodes au niveau *instances***	**Aspect maintenance**
ONTOFUSION	Médiateur	4 fichiers XML / manuel	Ontologie(s) / manuel	Manuelle	Unification des schémas virtuels en un unique donc regroupement des concepts identiques ou liés hiérarchiquement dans un même concept / automatique	-	Interface pour faciliter mais travail entièrement manuel
SEMEDA	Médiateur	Méta-données / manuel	Ontologie / manuel	Manuelle	Unification et précision des attributs des sources grâce à des ontologies et vocabulaires contrôlés intégrés au système / manuel	Proposition d'exploiter les valeurs des attributs pour en préciser le type mais non implémenté / -	Interfaces disponibles pour l'ajout et la suppression d'attributs des sources mais à réaliser manuellement
BioMediator	Médiateur + navigationnelle	Méta-données regroupées dans une ontologie / manuel	Hiérarchies de classes et de propriétés / manuel mais extensible automatiquement	Règles définies pour chaque source / manuel	Association des méta-données avec le schéma médiateur en exploitant les règles de mises en correspondance / automatique	-	Système flexible mais nécessité de faire les mises à jour (rarement nécessaires) manuellement

2.3 Problématique de mise en correspondance de schémas

Les travaux réalisant une intégration de sources de données avancée nécessitent de mutualiser des informations issues de sources distribuées et hétérogènes. En pratique, cela consiste à mettre en correspondance des éléments issus des schémas des sources considérées avec des éléments d'un (ou plusieurs) schéma global. La problématique de mise en correspondance de schémas n'est ni nouvelle ni spécifique du problème qui nous intéresse. Elle se pose en effet dans de nombreuses applications touchant à des domaines divers de l'informatique, comme la recherche documentaire ou encore dans le traitement des requêtes à travers des entrepôts de données. La mise en œuvre de telles applications nécessite également de considérer des informations issues de sources et/ou d'ontologies distantes pour ensuite les représenter de manière globale. L'opération de mise en correspondance de schémas peut être définie ainsi : elle prend en entrée deux schémas (représentant chacun une source) ou ontologies, chacun consistant en un ensemble d'éléments (classes, attributs, relations, etc), et détermine les relations (équivalence, subsomption, etc) existant entre ces éléments. Shvaiko et al. [Shvaiko 05] ont souligné qu'il est possible de considérer ensemble les approches développées pour réaliser la mise en correspondance d'ontologies avec celles pour effectuer la mise en correspondance de schémas, les unes pouvant bénéficier des autres. Par la suite, on utilisera cependant le terme « schéma » car dans le cas des sources de données biomédicales, on ne dispose pas d'ontologie associée pour les représenter.

Dans cette section, nous présentons tout d'abord des définitions et caractéristiques de l'opération de mise en correspondance permettant de classifier les différentes méthodes existantes. Nous introduisons ensuite ces méthodes, en distinguant celles qui exploitent le niveau *schéma* de celles situées au niveau *instances*. Nous considérons finalement des travaux qui ont implémenté des approches situées au niveau *instances*, c'est-à-dire exploitant les données présentes dans les sources avec comme objectif d'obtenir des informations utiles pour la mise en correspondance de schémas de sources.

2.3.1 Définitions et caractéristiques de l'opération de mise en correspondance de schémas

2.3.1.1 Définitions

L'opération de mise en correspondance de schémas consiste donc à analyser deux schémas fournis en entrée afin d'identifier des correspondances entre des éléments de ces deux schémas [Rahm 01]. **Un élément de l'opération de mise en correspondance de schémas** est défini par un « 5-uplet » $<id,e,e',n,R>$ où [Shvaiko 05] :

- id est l'identifiant de l'élément de mise en correspondance ;
- e et e' sont les éléments respectifs de chaque schéma que l'on souhaite mettre en correspondance. Cela peut inclure non seulement les attributs des sources mais aussi les relations existant entre eux, ou bien les propriétés les précisant ;
- n est la mesure de confiance permettant d'évaluer la correspondance entre e et e' ;
- R est une relation existant entre les deux éléments (équivalence, plus - ou moins - général, disjonction, recouvrement).

On dira qu'un **alignement** est l'ensemble des éléments obtenus lors de l'opération de mise en correspondance.

2.3.1.2 Caractéristiques

L'opération de mise en correspondance de schémas est caractérisée par le niveau auquel elle se situe (*schéma* ou *instances*), le type et la granularité des éléments qu'elle prend en entrée, les informations exploitées pour mettre en œuvre ce processus et enfin, la cardinalité et la graduation de la mise en correspondance. Ces critères sont définis dans la suite de cette section.

Tout d'abord, l'opération de mise en correspondance **s'effectue soit au niveau *schéma*, soit au niveau *instances***. Comme indiqué précédemment, dans le premier cas, seules les informations existant dans le schéma sont utilisées pour effectuer l'opération. Par contre, pour les approches situées au niveau *instances*, ce sont les données elles-mêmes qui sont exploitées pour inférer des informations nécessaires à une mise en correspondance des éléments des schémas des sources associées. L'exemple de l'attribut « Approved Symbol » introduit en début de la section précédente (cf 2.2.1.1 page 34) illustre l'intérêt d'utiliser les informations situées au niveau *instances*. En effet, cet attribut présent dans la source HGNC manque de précision et l'analyse des valeurs associées à celui-ci peut permettre de le désambiguïser en identifiant le contexte dans lequel cet attribut s'exprime. On espère ainsi pouvoir déterminer que cet attribut a pour valeurs des symboles de gènes et devrait donc être mis en correspondance avec un attribut ayant par exemple comme nom « Gene Symbol » et non pas « Protein Symbol ». Les valeurs utilisées correspondent bien aux données contenues dans la source HGNC mais leur exploitation a pour but d'apporter des informations au niveau *schéma* afin de déterminer quel autre attribut peut être mis en correspondance avec « Approved Symbol ».

Parmi les approches existantes, il faut distinguer celles qui tiennent compte de la granularité des entrées, des autres. En terme de **granularité**, une différence existe entre des méthodes portant uniquement sur les éléments des schémas par rapport à celles qui exploitent la structure des schémas. Plus précisément, au niveau **élément**, l'opération de mise en correspondance consiste à considérer les éléments seuls, ignorant leurs relations avec d'autres éléments. Au niveau **structure**, on détermine les mises en correspondance possibles en analysant la manière dont les éléments des schémas apparaissent ensemble dans une structure. Dans ce deuxième cas, on pourra obtenir, par exemple, une correspondance entre un élément d'un schéma et deux éléments plus spécifiques présents dans un autre schéma (Figure 2.8 page ci-contre). Ici, la granularité des informations se trouvant dans les deux schémas est différente, raison pour laquelle il est intéressant de considérer plus globalement la structure que les éléments uniquement.

Les approches dépendent également du **type d'information traitée en entrée** se trouvant dans les sources plutôt que de leur implémentation (base de données relationnelle, XML ou base objet). Les algorithmes implémentés considèrent les éléments soit d'un point de vue terminologique et dans ce cas, ce sont des méthodes traitant les chaînes de caractères qui sont proposées, soit structurellement en exploitant la structure interne des entités (les types des attributs ainsi que les relations existant entre eux), soit de manière sémantique où le contexte des éléments

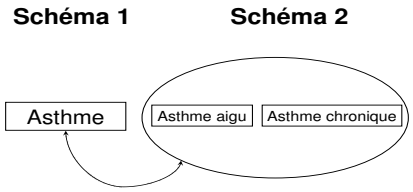


FIG. 2.8 – **Exemple de mise en correspondance au niveau *structure* entre des éléments de schémas de granularité différente.**

du schéma est considéré. Ces algorithmes sont plus détaillés dans la section suivante (cf 2.3.2 page 74).

Un autre paramètre d'entrée est généralement pris en compte, c'est le caractère **exact ou approximatif de l'opération**. Dans le premier cas, une solution absolue est calculée pour un problème tandis que dans l'autre, on cherche soit à identifier un plus grand nombre de correspondances, soit à disposer d'une indication pour effectuer des traitements supplémentaires dans un deuxième temps. Plus précisément, il n'est parfois pas possible d'obtenir des correspondances exactes et pour limiter le silence, on choisit de rechercher également des correspondances approximatives. Dans ce cas, l'algorithme permet de calculer plus de correspondances candidates au détriment de la précision, mais peut malgré tout constituer une bonne base à laquelle appliquer ensuite d'autres techniques plus poussées. Si on reprend l'exemple ci-dessus de l'attribut « Approved Symbol », c'est grâce à des correspondances approximatives qu'il pourrait être associé à « Gene Symbol » et « Protein Symbol ». Cela est insuffisant car ambigu mais l'exploitation des valeurs respectives de ces attributs permettra de valider ou non les correspondances approximatives identifiées préalablement.

Enfin, les entrées de l'opération de mise en correspondance sont parfois complétées par l'**exploitation d'informations auxiliaires**. En effet, tandis que certaines approches réalisent l'interprétation des entrées d'un schéma en ne considérant que sa propre structure, d'autres exploitent des informations externes présentes dans des ressources auxiliaires d'un domaine et d'une connaissance commune. Cela permet par exemple de déterminer que l'élément E1 d'un schéma peut être mis en correspondance avec l'élément E2 d'un autre schéma car une ressource terminologique externe contient un concept dont le terme préféré est E1 et l'un des termes synonymes est E2.

La dernière caractéristique introduite ici concerne les **sorties** fournies par l'opération de mise en correspondance. Les résultats sont de différentes formes et n'ont notamment pas la même **cardinalité**. On obtient en effet des correspondances directes où un élément du schéma 1 est associé à un élément du schéma 2 et dans ce cas, la cardinalité est de type 1-1. Parfois, on ne parvient pas à trouver ce type de correspondance et ainsi apparaissent des résultats plus complexes à exploiter où un élément d'un des schémas est associé à plusieurs éléments de l'autre schéma (cardinalités 1-n ou m-1) ou encore de cardinalité m-n si un ensemble d'éléments du schéma 1 est associé à un ensemble d'éléments du schéma 2, ce qui est encore plus difficile à

traiter.

Finalement, ces différents types d'approches peuvent être combinées. Une dernière distinction existe donc : les **opérations de mise en correspondance élémentaires** qui n'implémentent qu'une unique approche et les **opérations combinées**. Ces dernières peuvent être divisées en deux : les opérations dites **hybrides** qui consistent à enchaîner séquentiellement plusieurs approches en suivant un ordre de préférence défini préalablement et les opérations dites **composites** où les approches sont exécutées parallèlement pour ensuite déterminer quelle combinaison permet de déterminer les mises en correspondance les plus pertinentes en fonction de critères multiples [Rahm 01].

2.3.2 Approches au niveau *schéma*

Diverses classifications des techniques développées au niveau *schéma* ont été proposées. Nous avons choisi d'utiliser celle présentée dans [Shvaiko 05] mais de manière simplifiée en nous limitant à la classification ascendante qui est réalisée en fonction du type des objets qui sont manipulés par les méthodes de mise en correspondance (Figure 2.9 page ci-contre). Les auteurs ont décrit une classification suivant deux axes, celui que nous choisissons d'utiliser ainsi que l'axe descendant qui aborde les techniques existantes en fonction de la granularité des entrées (élément / structure). Cette classification est assez complexe et il nous a paru plus clair de prendre celle-ci suivant un axe unique. D'autre part, la classification réalisée par Rahm et al. pour les approches au niveau *schéma* est assez proche de l'axe descendant du travail précité mais elle est moins précise et moins avancée (absence des méthodes sémantiques) [Rahm 01]. Enfin, la classification proposée par Kalfoglou et al. ne nous convenait pas non plus car elle est effectuée suivant différentes catégories permettant de caractériser 35 travaux existants, mais pas réellement pour classer les techniques mises en œuvre dans ces derniers [Kalfoglou 03].

2.3.2.1 Terminologiques

Les approches terminologiques se focalisent sur les manipulations qu'il est possible de réaliser sur les termes ou ensembles de termes. Deux types existent : les approches syntaxiques qui traitent des chaînes de caractères, c'est-à-dire qui interprètent les termes comme des séquences de caractères, et les approches linguistiques qui, elles, considèrent les termes en tant qu'objets linguistiques. Nous détaillons dans les deux parties suivantes quelques approches de ces deux types[29].

2.3.2.1.1 Syntaxiques. Elles sont pour la plupart utilisées pour mettre en correspondance des noms ou des descriptions associés aux noms d'éléments de schémas. L'idée est que plus des chaînes de caractères sont similaires, plus elles ont de chance de décrire le même concept. Des exemples de techniques considérant en entrée les chaînes de caractères `Ch1` et `Ch2` (représentant des attributs, par exemple) issues de deux schémas différents sont ([Do 02]) :

– la comparaison de préfixe : `Ch1` commence par `Ch2` (ou inversement), permettant de mettre en correspondance des termes de même origine ou des acronymes similaires ;

[29] http ://www.lirmm.fr/ mroche/Recherche/Exposes/Roche_Forum.pdf

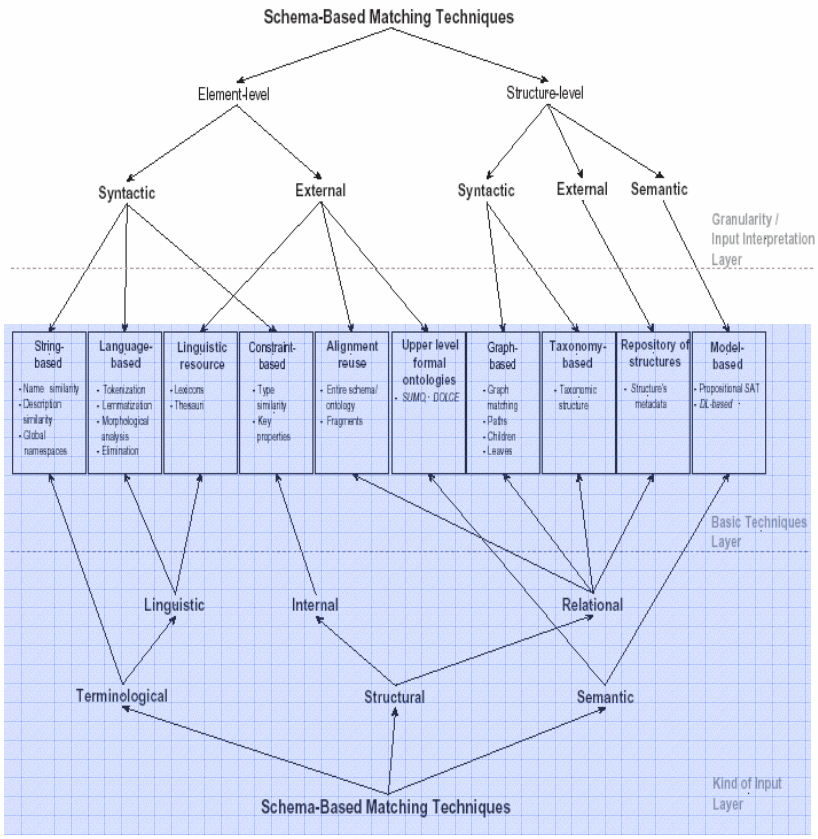


FIG. 2.9 – **Classification des approches de mises en correspondance situées au niveau *schéma*. Figure issue du papier de Shvaiko et al. ([Shvaiko 05]).** Nous nous basons uniquement sur la classification ascendante (surlignée en bleu).

– la comparaison de suffixe : `Ch1` termine par `Ch2` (ou inversement) ;
– la distance « Edit Distance » entre `Ch1` et `Ch2`, c'est à dire la somme minimale du coût des opérations élémentaires pour transformer `Ch1` en `Ch2` (par des opérations élémentaires comme l'ajout, le remplacement ou la suppression d'une lettre) ;
– le N-gramme qui identifie le nombre de séquences de N caractères qui sont communes à `Ch1` et `Ch2`.

Pour finir, on soulignera que ces techniques syntaxiques sont bien connues en bioinformatique car les algorithmes classiques de comparaison de séquences d'ADN ou d'ARN les utilisent, par exemple BLAST.

2.3.2.1.2 Linguistiques. Elles sont basées sur les techniques de traitement du langage naturel (ou NLP - *Natural Language Processing* en anglais) qui exploitent les propriétés lexicales des mots fournis en entrée (par exemple [Giunchiglia 05]). La *tokenisation* repère les mots dans les noms d'entités par reconnaissance de la ponctuation, des majuscules, des caractères blancs, etc. Par exemple, l'attribut de nom « Approved_Symbol » est découpé en deux mots « approved » et « symbols ». La *lemmatisation* détermine la forme normalisée des mots, c'est-à-dire le masculin singulier pour les noms et les adjectifs et l'infinitif pour les verbes. Ainsi, la lemmatisation des deux mots de l'exemple précédent donne respectivement « approve » et « symbol ». Enfin, des mots non pertinents tels que des articles, des prépositions ou des conjonctions sont marqués pour éventuellement être supprimés. Ces techniques servent généralement de pré-traitement à d'autres approches de manière à maximiser le nombre de mises en correspondance identifiées.

D'autres techniques linguistiques exploitent des ressources externes pour identifier des relations linguistiques, telles que la synonymie et l'antonymie, entre termes de schémas différents. Il est possible d'utiliser des thésauri de connaissances générales ([Kefi 06] avec WordNet) ainsi que des thésauri spécifiques d'un domaine ([Zhang 05] avec le FMA - Foundational Model of Anatomy - en anatomie) pour connaître le sens des termes considérés.

Ces différentes techniques se focalisent sur les termes et considèrent donc uniquement les éléments des schémas, et non pas la structure du schéma.

2.3.2.2 Structurelles

Certaines approches sont **basées sur les contraintes internes structurant le schéma**, telles que les types, l'échelle de valeurs des attributs et les clés primaires ou étrangères, qui sont appliquées aux définitions des éléments présents dans les schémas. Si ce type d'informations est présent dans les schémas à mettre en correspondance, il peut être intéressant de l'utiliser pour identifier des associations, comme dans [Larson 89]. Par exemple, l'équivalence des types des éléments ou la présence de caractéristiques communes sur des clés pourraient donner une indication sur l'éventuelle similarité des deux attributs considérés et permettre ainsi de les étudier plus en détail afin de déterminer s'ils sont réellement similaires.

D'autres méthodes **réutilisent des alignements pré-existants** dans des ressources externes, par exemple [Do 02]. Certains schémas à traiter ressemblent (tout du moins partagent des éléments communs) à d'autres qui ont déjà été mis en correspondance avec des schémas, en particulier quand on est dans le même domaine d'application. Une autre option consiste à **utiliser des entrepôts de structures** qu'une application remplit avec les similarités deux-

à-deux identifiées entre deux schémas, comme dans [Rahm 04]. Il est possible de ré-utiliser ces informations quand on relance l'application ou qu'on y ajoute un nouveau schéma à comparer avec d'autres. Dans ces cas, cela permet de récupérer des correspondances ou similarités ayant déjà été définies entre schémas pour une application donnée et de vérifier si elles peuvent être utiles pour la mise en correspondance d'un nouveau schéma avec un qui est déjà existant.

Certains algorithmes considèrent les schémas fournis en entrée comme des **structures en graphes** contenant des termes et leurs relations, comme dans [Zhang 05], [Ehrig 04] ou encore [Doan 03]. Cela permet d'exploiter des méthodes développées pour la comparaison de graphes afin d'identifier des similarités entre une paire de nœuds en fonction de leur position respective (et plus généralement leur contexte) dans chacun des graphes (ou des **taxonomies**, en fonction des schémas disponibles en entrée). Certains algorithmes exploitent la similarité entre les enfants de deux nœuds de graphes distincts pour calculer leur similarité. Ainsi, on considère que deux éléments non feuilles d'un graphe sont similaires structurellement si leur ensemble de feuilles enfants (ou même descendants) sont hautement similaires (à un certain pourcentage). Parallèlement, des similarités existant entre les parents (nœuds du niveau supérieur), voire même entre les ascendants (nœuds des niveaux supérieurs), des deux nœuds considérés peuvent permettre de mettre ces derniers en correspondance. D'autres approches existent, notamment basées sur le calcul de la similarité entre nœuds à partir de celle existant entre les relations associées. Plus globalement, certaines méthodes se basent sur la similarité entre deux chemins menant aux nœuds initiaux à mettre en correspondance, qui est déterminée en fonction du nombre de nœuds parcourus communs et de leur position.

Ces techniques structurelles exploitent par définition plutôt la structure des schémas mais elles peuvent aussi ne considérer que les éléments, notamment celle qui est basée sur les contraintes, où ce sont malgré tout les aspects structurels des schémas qui sont utilisés pour mettre en œuvre les méthodes de mise en correspondance.

2.3.2.3 Sémantiques

Peu d'approches purement sémantiques existent dans la littérature et trop peu de travaux ont mis en œuvre ce type de techniques pour qu'on puisse affirmer qu'elles sont efficaces [Noy 04]. Le principe consiste à se baser sur les propriétés sémantiques des classes pour reconnaître automatiquement des correspondances telles que des équivalences, ou au moins des subsomptions.

Les **ontologies formelles de haut niveau** peuvent être utilisées en tant que ressources externes de connaissance « commune » pour mettre en œuvre des techniques dites sémantiques puisqu'elles sont basées sur la logique. Celles-ci offrent la possibilité d'analyser et de raisonner sur des interprétations que l'on peut faire des éléments à associer (par exemple, au travers de descriptions disponibles associées à des concepts d'ontologies ou à des éléments de schémas). Un exemple d'ontologie de ce type est DOLCE qui a été utilisée notamment pour ajouter une couche formelle à WordNet [Miller 98], une base de données lexicales de termes anglais, en fournissant une spécification formelle du niveau supérieur de cette deuxième ressource [Gangemi 03].

Des algorithmes qui exploitent les entrées par rapport à leur interprétation sémantique sont basés sur des méthodes déductives. Les logiques de description, par exemple, permettent d'exprimer des relations d'inclusion à l'aide de la subsomption. En fait, pour effectuer un alignement entre deux schémas, on peut d'abord appliquer des méthodes plus classiques (terminologiques

et structurelles introduites précédemment) pour identifier des correspondances. Il serait ensuite possible de tester, au moyen d'un classifieur, chaque paire de concepts et relations préalablement trouvée pour ne garder que les correspondances pour lesquelles l'interprétation est similaire.

2.3.3 Approches au niveau *instances*

Les approches situées au niveau *instances* ont d'abord émergé pour traiter les cas où les informations disponibles concernant les schémas étaient limitées. Mais il est rapidement apparu qu'elles sont aussi bénéfiques même si des informations conséquentes sont accessibles au sujet des schémas puisqu'elles peuvent valider ou non des interprétations réalisées en exploitant uniquement le niveau *schéma* ou même permettre de désambiguïser des correspondances multiples obtenues pour deux éléments [Rahm 01], [Köhler 03]. L'approche la plus simple consiste à considérer les **ensembles de données** associées à chacun des éléments que l'on souhaite mettre en correspondance et d'en faire l'**intersection** pour déterminer la similarité existant entre les deux éléments [Euzenat 04].

Les techniques **terminologiques** présentées dans la section précédente peuvent être également mises en œuvre pour traiter les données au niveau *instances*. De même, les méthodes structurelles basées sur la **ré-utilisation d'alignements** sont susceptibles de fonctionner pour caractériser les données dans la mesure où on peut identifier des termes récurrents au niveau *instances* et qui sont également définis au niveau *schéma*. Par exemple, si le terme « Alzheimer » apparaît fréquemment parmi les données d'un attribut A et que ce même terme a déjà été mis en correspondance dans des alignements existants avec l'élément « Maladie » (relation de spécialisation), alors on va pouvoir proposer de mettre en correspondance l'attribut A avec l'élément « Maladie » (Figure 2.10 page suivante(a)).

Des méthodes sont malgré tout spécifiques au niveau *instances*, elles ont généralement pour objectif de préciser certaines informations existant au niveau *schéma*. Il est notamment possible, dans le cas où on dispose de **données structurées** comme des nombres ou des chaînes de caractères particulières, d'appliquer respectivement des intervalles de valeurs ou des patrons, de manière à **reconnaître des données de format connu**, telles que des dates, des adresses email, des numéros de téléphone, etc. Cela permet donc d'inférer des informations concernant les contraintes des éléments du schéma comme le type de données d'un attribut. Pour les données de type texte, des **approches développées en recherche d'informations** peuvent être utilisées. Leur but est d'extraire des mots-clés appartenant à un même thème à partir de leur fréquence d'apparition, par exemple. Ces mots-clés identifiés peuvent être répertoriés dans une ressource externe et ainsi permettre de préciser là aussi le type des attributs auxquels les données sont associées. Par exemple, pour un élément textuel contenant de nombreuses occurrences des noms de maladies « Cancer », « Alzheimer », « Épilepsie » et « Céphalée » qui sont pour la plupart représentées dans une ontologie en tant qu'instances d'un concept « Maladie », on peut déduire que l'attribut associé à cet élément est lié au concept « Maladie » (Figure 2.10 page ci-contre(b)). Enfin, puisque le volume de données à traiter est conséquent, des techniques de fouille de données et d'apprentissage (par exemple, basées sur les réseaux de neurones) sont particulièrement bien adaptées à ce type d'informations.

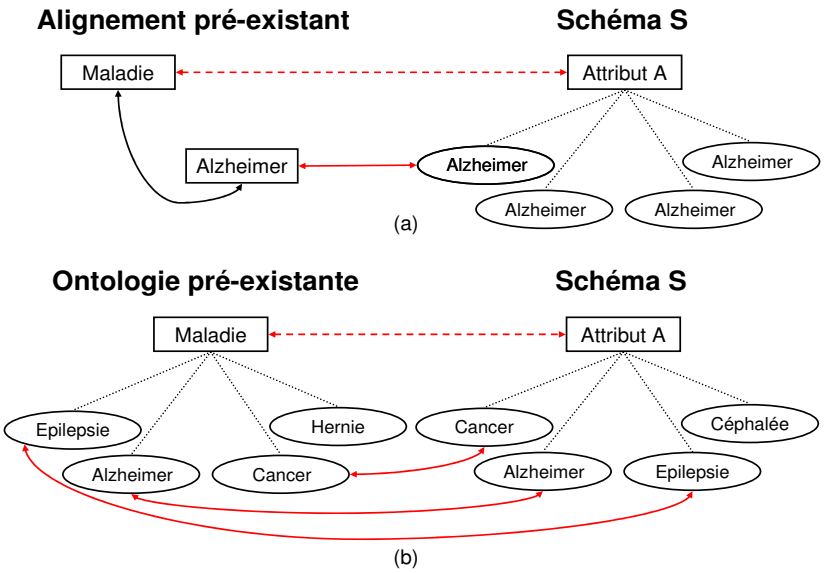


FIG. 2.10 – **Mise en correspondance au niveau *instances* en exploitant (a) des alignements pré-existants et (b) des ressources terminologiques pré-existantes.** Les traits grisés correspondent aux instanciations de concepts ou attributs (respectivement dans des ontologies et schémas), les flèches réflexives en rouge indiquent les mises en correspondance qui ont déjà été identifiées et les flèches pointillées réflexives sont les mises en correspondance induites.

2.3.4 Approches existantes

De nombreux travaux ont été menés pour résoudre les problèmes d'hétérogénéités entre ontologies / schémas et plus précisément pour les aligner. Il serait difficile de donner un inventaire exhaustif des différentes systèmes existants, raison pour laquelle nous choisissons ici de ne présenter que certains systèmes qui intègrent des méthodes situées au niveau *instances*, sachant que c'est souvent en complément d'approches exploitant le niveau *schéma*.

Le système **GLUE** [Doan 03] (initialement LSD pour Learning Source Descriptions) cherche à identifier des mises en correspondance de cardinalité 1-1 entre des concepts d'ontologies et suit une approche composite. Il fonctionne en plusieurs étapes ; d'abord une analyse statistique est effectuée sur les instances des concepts pour déterminer la probabilité qu'une instance du domaine décrit par les deux ontologies appartienne ou non aux deux concepts distincts. Ce calcul statistique est réalisé par différentes techniques d'apprentissage qui sont entraînées sur les instances d'un des concepts (les exemples positifs sont les instances présentes dans le concept et les négatifs celles qui sont absentes) et appliquées ensuite sur les instances de l'autre concept pour déterminer celles qui sont communes. Divers critères sont utilisés pour évaluer l'appartenance d'une instance à un concept (il y a autant de techniques d'apprentissage que de critères). Ces critères sont notamment la fréquence des mots dans le contenu de l'instance si celui-ci correspond à des données textuelles ou encore la comparaison des noms des instances. Les différentes probabilités obtenues sont ensuite utilisées par le coefficient de Jaccard [Van Rijsbergen 79] qui

détermine la similarité entre deux concepts en calculant le rapport des probabilités de l'intersection des instances associées aux concepts sur celle de leur union. Les similarités ainsi calculées pour les différentes techniques sont ensuite combinées et pondérées en fonction de leur degré de confiance (attribué arbitrairement) de manière à déterminer une valeur unique de similarité pour chaque paire de concept. La matrice de similarités des concepts deux à deux ainsi créée est finalement utilisée par un dernier module qui permet de déterminer les meilleures correspondances identifiées en appliquant des contraintes et des heuristiques. Un exemple de contraintes est que si tous les enfants d'un concept X sont mis en correspondance avec Y alors X peut être associé à Y. Comme heuristique, le voisinage du concept est exploité au sein d'une ontologie (similarité de ses voisins, parents, enfants, etc) pour essayer de trouver d'autres correspondances. Les correspondances déterminées comme les plus pertinentes après cette dernière étape sont proposées aux utilisateurs qui peuvent les valider ou non. En conclusion, ce système permet de combiner des techniques d'apprentissage qui sont appliquées aux instances pour déterminer la similarité entre les concepts de deux ontologies et complète ces résultats en appliquant des méthodes structurelles au niveau *schéma*. Le problème des approches d'apprentissage est qu'elles nécessitent l'intervention humaine au moment de la phase d'entraînement avant de pouvoir être appliquées automatiquement, ce qui peut constituer un lourd travail manuel.

QOM (Quick Ontology Mapping) [Ehrig 04] est une méthodologie visant à mettre en correspondance des ontologies par recherche de similarités entre deux entités étant des concepts, des relations ou des instances. Elle couvre des correspondances de cardinalité 1-1 et est une approche composite. De nombreuses méthodes sont implémentées :

- des techniques terminologiques déterminent que deux entités sont similaires si leur libellé sont identiques ou si elles ont le même identifiant ;
- des méthodes structurelles exploitent la similarité entre super/sous-concepts pour en inférer que les deux concepts initiaux sont semblables ;
- des approches utilisant les instances permettent également de définir une correspondance entre deux concepts si elles sont identiques. Il est aussi possible de déduire que si deux instances sont liées à une autre avec la même propriété alors les deux instances initiales sont identiques ;
- des techniques basées sur les constructeurs existant dans les langages permettant de décrire des ontologies sont également exploitées pour comparer des concepts ou des instances. En l'occurrence, des propriétés OWL telles que `sameClassAs` et `sameIndividualAs` (*individual* correspondant à la notion d'instances) peuvent être utilisées directement.

Plusieurs approches visant à déterminer la meilleure manière de combiner les similarités obtenues à partir des différentes méthodes sont proposées. La combinaison choisie est celle qui fournit les valeurs de similarités les plus élevées. Trois mesures différentes permettent de définir le seuil en dessous duquel le système exclura les correspondances identifiées, celles à valider (manuellement) et celles à garder. En résumé, ce système est flexible car il permet de nombreuses compositions des méthodes terminologiques et structurelles développées qui sont de plus générées automatiquement mais aussi une variabilité au niveau du choix du seuil de similarité. Cependant, les méthodes implémentées nécessitent que les ontologies soient riches (présence de propriétés, description dans un langage formel).

Xu et Embley [Xu 03] proposent une méthode automatique pour aligner deux schémas grâce à l'exploitation d'une ontologie de domaine. Le principe consiste tout d'abord à représenter les schémas source et cible sous forme de graphes conceptuels où les nœuds sont des éléments correspondant soit à des attributs soit à des identifiants des tables concernées, et les flèches représentent les relations entre ces éléments. Quatre techniques sont appliquées pour l'alignement : d'abord des méthodes terminologiques permettant de mettre en correspondance des éléments ayant le même nom sont implémentées. Elles exploitent une ressource externe, WordNet, et plus précisément ses propriétés, comme les synonymes et hypernymes (super-concepts dans le vocabulaire de WordNet). La seconde technique vise à mettre en correspondance des éléments ayant des valeurs de mêmes caractéristiques (méthode basée sur des contraintes donc structurelle). La troisième approche est celle qui nécessite la création d'une ontologie de domaine. Celle-ci, construite manuellement de manière à pouvoir couvrir les deux schémas, est constituée de concepts et relations et associe à chaque concept des expressions régulières qui permettent de représenter et identifier les valeurs et mots-clés apparaissant sous le concept. Au travers de l'ontologie, les valeurs associées aux éléments des schémas peuvent être retrouvées et utilisées pour l'alignement. Trois types de correspondances indirectes peuvent ainsi être identifiées si : 1) les valeurs associées à plusieurs éléments d'un des schémas correspondent aux valeurs d'un élément de l'autre ; 2) les valeurs d'un élément du schéma cible constituent un sous/sur-ensemble de plusieurs éléments du schéma source ; 3) les éléments d'un des schémas ayant pour valeurs des booléens correspondent à des valeurs dans l'autre schéma. Enfin, la dernière méthode proposée est basée sur la structure des schémas et exploite le contexte dans lequel s'expriment les différents éléments (éléments adjacents, parents, etc). Une valeur de confiance (déterminée expérimentalement) est associée à chacune des correspondances obtenues et les résultats des différentes techniques sont combinées de manière indépendante (approche composite). Ce travail regroupe donc de nombreuses méthodes de nature diverse (terminologiques et structurelles). En particulier, l'approche basée sur une ontologie de domaine, qui bien que construite manuellement (ce qui constitue une certaine limite à cette approche), est malgré tout intéressante puisqu'elle a l'intérêt de pouvoir aligner n'importe quels schémas de ce même domaine.

L'approche proposée par **Kang et al.** [Kang 03] est aussi basée sur les instances mais ce qui fait son originalité est que contrairement aux autres, elle n'interprète pas les données. L'objectif est d'identifier les corrélations entre colonnes, c'est-à-dire les relations de dépendance existant entre les attributs de chaque schéma. Pour cela, les deux schémas initiaux sont d'abord transformés en graphes de dépendance dont les valeurs sont calculées en considérant la quantité d'informations qu'un attribut contient à propos de l'autre attribut (en termes d'instances notamment). Les graphes sont ensuite mis en correspondance à partir de deux mesures de similarité : les distances euclidienne et normale adaptées spécifiquement pour comparer des graphes de dépendance, et les résultats sont fournis avec leurs précision et rappel. L'avantage de cette technique est que même si les données ne sont pas encodées de la même façon d'un schéma à l'autre, il est malgré tout possible de mesurer des relations de dépendance, ce qui fait que cette technique peut s'appliquer à d'autres domaines sans aucune modification. L'inconvénient est que c'est une approche exclusivement structurelle et qu'elle ne prend pas en compte le contexte

puisqu'elle n'interprète pas les données. C'est pour cette raison que les auteurs précisent bien qu'elle peut être utilisée en complément d'autres approches.

Clio [Miller 01] est un système visant à gérer et faciliter les tâches complexes nécessaires pour transformer et intégrer des sources de données hétérogènes. En particulier, un des buts de ce système est d'associer des éléments constituant un schéma source aux éléments d'un schéma cible, ces schémas devant être structurés ou semi-structurés. Différents composants sont mis en œuvre pour cela :

- un module permet de charger dans une interface les schémas des sources à mettre en correspondance et les augmente si cela est nécessaire. Plus précisément, il cherche à identifier des informations concernant les contraintes, comme la présence de clés primaires ou étrangères ;
- un second module [Naumann 02] génère un ensemble de correspondances candidates entre les attributs des deux sources. Pour cela, des vecteurs sont d'abord construits pour représenter les données associées aux attributs. En fonction du type des données, les vecteurs contiennent des valeurs différentes. Si les données sont textuelles alors on associe à chacune d'elles des valeurs booléennes suivant différentes caractéristiques, comme la présence de certains caractères spéciaux, tels que @ ou une majuscule au milieu d'un mot. Si les données sont numériques alors ce sont d'autres caractéristiques qui sont utilisées, comme la moyenne et la médiane. Les valeurs du vecteur de l'attribut correspondant sont définies par la moyenne des différentes valeurs pour chaque caractéristique. Une fois tous les vecteurs créés, c'est une classification grâce à une méthode d'apprentissage supervisée qui est réalisée. Une personne entraîne des exemples puis le classifieur se charge de trouver les autres correspondances. Ce module utilise également des techniques syntaxiques pour comparer les noms d'attributs (« Edit distance ») et préciser ainsi les correspondances candidates. Les utilisateurs peuvent ensuite choisir parmi ces propositions celles qui leur semblent correctes et éventuellement en ajouter d'autres ;
- le dernier module permet de faciliter le processus de mise en correspondance en ré-utilisant les correspondances déjà identifiées. Cela peut être utile si un nouveau schéma doit être associé à des schémas faisant déjà partie du système, voire même si les schémas existants sont modifiés. Les éléments n'ayant pas changé peuvent être mis automatiquement en correspondance avec les éléments auxquels ils avaient déjà été associés.

Cette approche hybride exploite donc les niveaux *schéma* et *instances* au travers de méthodes terminologiques et structurelles avec en particulier la ré-utilisation de correspondances existantes. Le problème est qu'un travail manuel assez lourd est nécessaire pour que les techniques d'apprentissage soient efficaces et l'autre inconvénient est que les schémas pouvant être traités doivent être structurés ou semi-structurés, ce qui n'est pas toujours le cas.

2.3.5 Conclusion

Pour conclure, il est important de pouvoir **combiner des méthodes** pour identifier des mises en correspondance entre éléments de schémas. Il est en effet bénéfique d'utiliser conjointement des techniques terminologiques, structurelles et sémantiques, appliquées aussi bien aux éléments ou bien à la structure en ce qui concerne le niveau *schéma* qu'aux valeurs associées aux

éléments des schémas pour ce qui est du niveau *instances*. Ainsi, de travaux nombreux existants offrent la possibilité de combiner plusieurs méthodes, soit de manière figée quand l'enchaînement de ces méthodes est fixée [Kang 03] et [Miller 01], soit de manière flexible et dans ce cas, les utilisateurs peuvent choisir l'ordre dans lequel ils souhaitent exécuter les différents algorithmes implémentant les techniques proposées ([Doan 03], [Ehrig 04] et [Xu 03]).

Le second aspect clé de la problématique de mise en correspondance, et plus généralement de l'alignement de schémas distribués et hétérogènes, concerne l'**automatisation des méthodes** utilisées. En effet, certains systèmes proposent des approches entièrement automatisées mais au prix de lourds efforts pour mettre préalablement en œuvre les outils et pré-requis nécessaires au bon fonctionnement du système. Par exemple, le système développé par Xu et Embley nécessite la description d'une ontologie du domaine pour représenter le contenu des sources dont on veut aligner les schémas [Xu 03], ce qui est très contraignant. Pour compenser cette limite, il faudrait travailler dans un domaine où ce type d'ontologie existe déjà, sachant que celle-ci doit en plus être peuplée d'instances (servant dans le processus de mise en correspondance). Les approches basées sur des méthodes d'apprentissage supervisées, elles, impliquent une phase manuelle d'entraînement avant de pouvoir être utilisées ([Doan 03] et [Miller 01]). En revanche, des systèmes implémentent des approches entièrement automatisées mais ils présentent malgré tout des inconvénients. La méthode proposée dans OQM s'applique à des ontologies et exploite leur sémantique riche, ce dont il est rare de disposer quand on travaille sur des schémas [Ehrig 04]. Enfin, l'approche introduite dans [Kang 03] nécessite d'être combinée avec d'autres méthodes de mise en correspondance car elle ne considère que des aspects syntaxiques, ce qui est limitatif.

Nous terminons par souligner que la plupart des schémas ont une sémantique pouvant influer sur (c'est-à-dire valider ou invalider) les correspondances identifiées mais qui n'est pas exprimée formellement ou même souvent pas du tout renseignée. Il apparaît donc nécessaire de laisser aux utilisateurs la possibilité d'accepter, refuser ou modifier ces correspondances candidates. Ce sont les deux raisons principales pour lesquelles la plupart des travaux existants n'offrent qu'une automatisation partielle, permettant aux utilisateurs de maîtriser la tâche délicate de validation.

2.4 Conclusions

En résumé, les biologistes et médecins doivent accéder à des informations réparties dans des sources de données distribuées, autonomes et hétérogènes à de nombreux niveaux. L'intégration de sources de données constitue une solution à ce problème. Elle permet de faciliter et de guider les biologistes et médecins dans leur recherche et leur collecte d'informations. Dans ce cadre, les **technologies du Web sémantique** peuvent être bénéfiques. Les ontologies présentent de nombreux avantages comme la possibilité de représenter des connaissances et garantissent l'interopérabilité. Elles sont par ailleurs un support pour résoudre les problèmes de mise en correspondance de schémas en tant que ressources externes, dans lesquelles se trouvent des synonymes ou concepts proches et parfois même des instances. Les méta-données permettent notamment de décrire les informations concernant les sources de données à intégrer (par exemple, le degré de confiance que l'on peut lui attribuer). Enfin, les langages sont indispensables car ils représentent

les différentes technologies afin qu'elles soient exploitables par les machines.

Différentes approches d'intégration existent et ont déjà considéré l'utilisation ce type de technologies mais certaines d'entre elles présentent des limites, principalement en ce qui concerne les méthodes ayant été mises en œuvre pour gérer les aspects suivants : le rythme rapide des modifications des schémas des sources et la résolution des hétérogénéités existant entre sources et enfin l'aide fournie aux biologistes et médecins pour construire leurs requêtes au moyen d'interfaces sophistiquées. Dans ce cadre, c'est l'**approche basée sur la médiation** qui semble la mieux adaptée, surtout si l'objectif est de proposer un système pour le domaine biomédical en général, et non un sous-domaine spécifique de celui-ci. Cependant, parmi les systèmes existants mettant en œuvre cette approche d'intégration, nous avons identifié un certain nombre de points qui ne les rendent pas tout à fait satisfaisants.

La découverte d'informations au sein même des sources afin d'en acquérir le schéma est faite en grande partie manuellement. La récupération automatique de ces informations permettrait de simplifier les tâches de conception et d'évolution dans un deuxième temps car cela rend possible une interrogation régulière des sources pour identifier des éléments nouveaux possiblement introduits dans les schémas des sources.

La définition d'un schéma global est un aspect qui est bien approfondi dans les systèmes d'intégration basé sur la médiation. Mais la constitution du schéma global y est faite manuellement. Cela présente des limites puisque c'est une tâche particulièrement difficile qui nécessite une connaissance suffisamment générale du domaine à couvrir ainsi qu'une description simple des éléments de l'ontologie de manière à ce qu'elles soient exploitables telles quelles par les utilisateurs biologistes ou médecins. Nous pensons que l'utilisation de systèmes terminologiques déjà existants ayant une étendue suffisamment large pour couvrir une bonne partie du domaine biomédical peut apporter une solution intéressante.

Un aspect très important dans les systèmes de médiation est la **mise en correspondance des éléments du schéma global avec ceux des schémas locaux**. Cette tâche est utile non seulement pour résoudre les hétérogénéités syntaxiques et sémantiques des sources à intégrer mais aussi pour traiter les requêtes posées dans les termes du schéma global par les utilisateurs afin de les reformuler dans les termes des schémas locaux. Les approches proposées dans ce cadre se situent presque exclusivement au niveau *schéma*. Pourtant, les approches de mise en correspondance généralement proposées au niveau *schéma* gagneraient à être complétées par des techniques situées au niveau *instances*. Pour cela, les méthodes implémentées dans des systèmes cherchant à aligner des schémas distincts peuvent être ré-utilisées (cf 2.3.4 page 79).

Enfin, la **maintenance des systèmes d'intégration** basée sur l'approche médiateur est rarement gérée ou uniquement au travers d'outils et interfaces assistant le travail des administrateurs. Certaines tâches nécessaires à cette maintenance sont facilitées par les points précédents, c'est-à-dire la récupération automatique des schémas des sources, la ré-utilisation de terminologies existantes pour constituer le schéma global et le développement de méthodes semi-automatiques pour mettre en correspondance les éléments du schéma global avec ceux des schémas locaux.

Chapitre 3

Objectifs

3.1 Objectif principal

L'objectif principal de ce travail est de concevoir un système d'intégration basée sur la médiation pour le domaine biomédical.

Comme nous l'avons décrit précédemment (2.2.3.3 page 53), des systèmes basés sur la médiation existent déjà mais ils présentent des limites. Dans ce cadre, nous proposons le développement d'un nouveau système, visant à mettre en œuvre les aspects suivants :

- l'**extraction automatique d'informations à partir des sources de données** de manière à faciliter la définition d'un schéma local par source à intégrer ;
- la **définition d'un schéma global** au moyen de méthodes semi-automatisées. Pour cette tâche, il nous semble important de pouvoir laisser aux concepteurs une marge de manœuvre qui leur permettra de vérifier que le schéma global a été construit de manière cohérente ;
- l'**utilisation de méthodes de mises en correspondance entre le schéma global et les schémas locaux situées au niveau *instances*** pour compléter les techniques mises en œuvre au niveau *schéma*. De plus, nous cherchons à proposer des techniques automatisées afin de faciliter cette tâche qui est également particulièrement lourde à réaliser par l'homme ;
- la **maintenance du système d'intégration au travers de méthodes automatiques**, ou semi-automatiques.

3.2 Objectifs spécifiques

Nos objectifs sont plus spécifiquement l'acquisition des schémas locaux de la manière la plus automatique possible, la conception du schéma global de manière automatisée, la mise en correspondance entre les éléments du schéma global et ceux des schémas locaux. Ces méthodes vont faciliter la création et la maintenance de notre système d'intégration.

3.2.1 Étape 1 : Acquisition des schémas locaux

Les schémas des sources de données étant rarement disponibles ou bien difficilement exploitables tels qu'ils sont fournis, nous allons définir une méthode visant à identifier des éléments utiles pour décrire le schéma des sources en exploitant le contenu de celles-ci. Après avoir extrait ces éléments des sources, nous proposons une méthode exploitant le niveau *instances* pour leur attribuer un type. À partir de ces éléments, de méta-données concernant les sources à intégrer et des références croisées, nous définissons les schémas locaux dans le langage XML.

3.2.2 Étape 2 : Conception du schéma global

La deuxième étape consiste en la conception du schéma global. Nous souhaitons ré-utiliser une ressource terminologique existante afin de faciliter et d'automatiser en partie la création du schéma global. Plus précisément, nous avons choisi l'UMLS pour jouer ce rôle puisqu'il couvre de nombreuses disciplines du domaine biomédical, en général. Cependant, cette ressource n'est pas une ontologie au sens formel du terme et pour pallier certaines de ses défaillances, en particulier

la présence de cycles, nous transformons l'UMLS en graphe orienté sans cycle. Cela permettra notamment d'exploiter sa hiérarchie de manière efficace, ce qui n'est pas possible sans traitement préalable de la ressource existante. Nous proposons deux approches pour éliminer les cycles dans l'UMLS et nous déterminons par un cas d'étude celle qui convient le mieux à nos besoins. Enfin, nous définissons le schéma global résultant au format OWL. Ce langage introduit précédemment dans le cadre du Web sémantique (2.1.2.1 page 19) permet de décrire des ontologies formelles. Même si l'UMLS n'en est pas une, nous verrons que l'utilisation d'un langage formel pourrait apporter des perspectives intéressantes à notre système (voir la discussion 7.4 page 178).

On notera que l'approche suivie pour la mise en correspondance sera de type LAV puisque le schéma global est décrit indépendamment des schémas des sources à intégrer. On définira donc ces derniers dans les termes utilisés dans le schéma global.

3.2.3 Étape 3 : Mise en correspondance du schéma global avec les schémas locaux

La dernière étape de conception du système consiste à identifier les correspondances existant entre les éléments des schémas locaux et ceux du schéma global. Pour cela, nous utilisons des méthodes terminologiques situées au niveau *schéma* qui visent à identifier les concepts de l'UMLS permettant de définir chaque élément présent dans les schémas des sources. Par ailleurs, l'UMLS étant spécifique du domaine biomédical, il ne permet pas de couvrir l'ensemble des éléments issus des sources, qui sont parfois assez généraux (par exemple, « Features »). Pour améliorer cette couverture, nous utilisons une ressource externe contenant des informations d'ordre général : WordNet. Nous montrons en quoi cette ressource complète les mises en correspondance identifiées dans l'UMLS directement et peut ainsi enrichir le schéma global. Pour cela, nous utilisons des approches terminologiques et structurelles situées au niveau *schéma*. Enfin, nous proposons une méthode située au niveau *instances* consistant à exploiter les valeurs des éléments constituant les schémas des sources afin d'identifier des correspondances supplémentaires dans le schéma global.

En résumé, nous souhaitons concevoir un système d'intégration basé sur l'approche médiateur pour le domaine biomédical. Sa constitution se fera au travers de trois étapes : l'acquisition des schémas locaux, la conception du schéma global et enfin la mise en correspondance des éléments du schéma global avec ceux des schémas locaux (Figure 3.1).

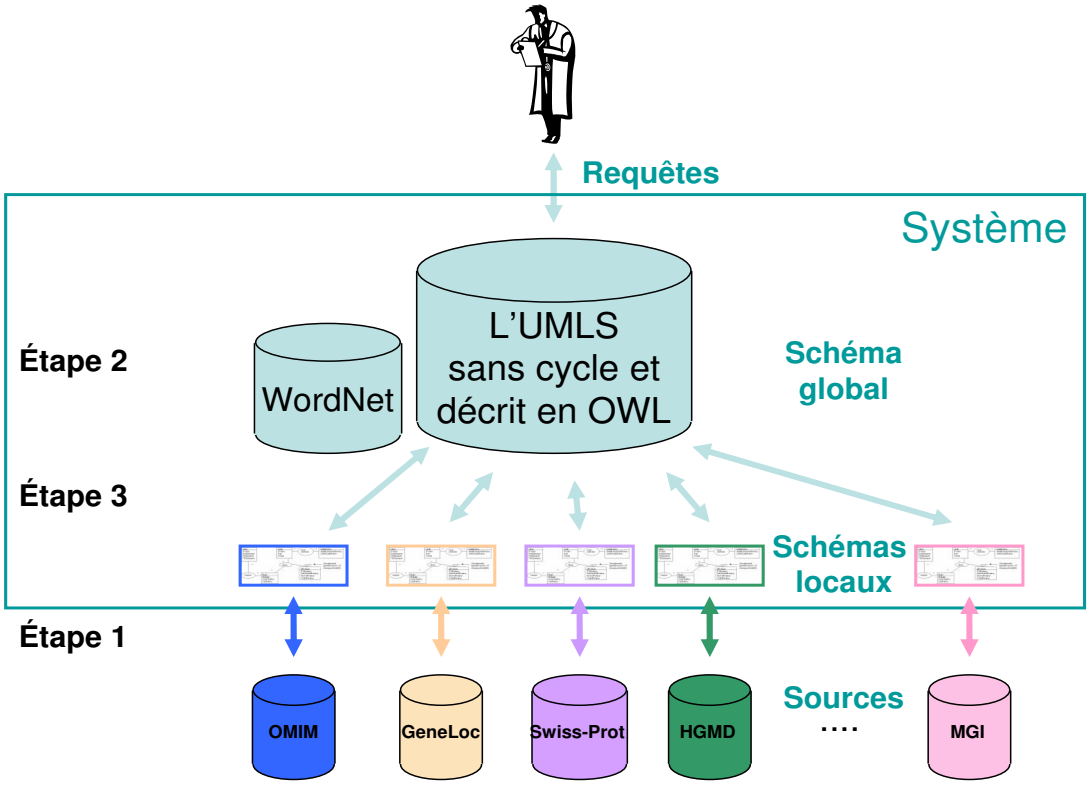


FIG. 3.1 – **Architecture du système attendu.** L'approche suivie est basée médiateur et ses composants sont : les sources de données originelles, leur schéma acquis au cours de l'étape 1, le schéma global construit à partir de l'UMLS et décrit dans le langage formel OWL lors de l'étape 2 et complété par WordNet à l'étape 3 au cours de laquelle sont également identifiées les correspondances entre les éléments du schéma global et ceux des schémas locaux, qui sont utilisées notamment pour la reformulation de requêtes.

Chapitre 4

Matériels et Méthodes

Dans ce chapitre, nous décrivons les ressources que nous avons utilisées pour réaliser ce travail. Plus précisément, nous présentons d'abord les sources de données que nous avons intégrées à notre système puis nous détaillons les ressources terminologiques que nous avons choisies pour résoudre l'hétérogénéité existant entre les sources : l'UMLS* et WordNet. Nous présentons ensuite les méthodes que nous avons développées pour réaliser la conception de notre système. Les différentes étapes sont détaillées : l'extraction d'éléments dans les sources de données considérées et plus généralement l'acquisition des schémas locaux. Ensuite, la conception ainsi que la description du schéma global sont données. Enfin, les méthodes pour mettre en correspondance les éléments des schémas locaux avec ceux du schéma global sont décrites.

4.1 Matériels

4.1.1 Les sources de données intégrées

Pour choisir les sources à intégrer à notre système, nous avons étudié les ressources utilisées par des biologistes travaillant au sein de l'unité INSERM U522, avec qui nous collaborons. Il est apparu que ceux-ci commençaient par interroger des sources généralistes pour avoir tout d'abord une vue d'ensemble concernant leurs expérimentations. Dans notre cas, les biologistes réalisent des expériences de puces à ADN et sont donc intéressés par les informations à propos des gènes étudiés par cette technologie. Des données additionnelles sont susceptibles de leur apporter des informations pertinentes : les protéines qui sont associées à ces gènes, ainsi que les maladies dans lesquelles les gènes sont potentiellement impliqués.

À partir des constats tirés de cette analyse, nous avons choisi onze sources en accord avec les critères suivants :

- elles doivent **contenir des informations autour des entités biologiques et médicales générales suivantes : le gène, la protéine et la maladie** ;
- elles doivent **être complémentaires**. Des sources généralistes ainsi que des sources spécifiques à un type de données précis (la localisation chromosomique d'un gène, le type de mutations connues que peut subir un gène, les protéines interagissant avec l'entité étudiée, les différents organismes affectés par la maladie identifiée) doivent être intégrées ;
- elles **ne doivent pas traiter uniquement de l'espèce humaine**. En effet, notre système n'ayant pas pour but d'être dédié uniquement aux données traitant de l'homme, il faut intégrer au moins une source d'un autre organisme particulier ;
- elles doivent **être accessibles librement sur Internet**.

4.1.1.1 Les sources génomiques

La source **Entrez Gene** [Maglott 05] (initialement LocusLink), développée par le NCBI, fournit des informations spécifiques aux gènes, et plus précisément se focalise sur les génomes qui ont été séquencés entièrement. Entre autres, on trouve comme type de données : des produits de gènes, des phénotypes, des références bibliographiques, des séquences, cartes et homologues. De nombreuses références croisées sont proposées, aussi bien vers l'ensemble des ressources du NCBI, notamment OMIM et GenBank, que vers des ressources externes, comme HPRD et

HGMD. Entrez Gene contient environ 2 000 000 d'entrées correspondant à des gènes spécifiques d'un organisme, celles-ci sont stockées dans une base de données relationnelle. Les données de cette source résultent d'informations récoltées automatiquement dans d'autres ressources du NCBI ou d'autres collaborations qui sont validées et corrigées par les experts responsables de cette tâche au NCBI. L'interrogation d'Entrez Gene est la même que pour l'ensemble des ressources du NCBI, elle se fait grâce au portail Entrez avec une requête écrite en language naturel.

La source de données **GeneCards** [Safran 03] intègre des informations très complètes extraites d'autres bases contenant des données qui concernent les gènes humains, les fonctions des protéines qu'ils encodent et les maladies dans lesquelles ils sont impliqués. En plus des données physiquement intégrées, GeneCards fournit de nombreuses références croisées vers des sources de référence, telles que OMIM, Swiss-Prot, HUGO et vers des sources spécialisées, comme Breast Cancer Gene Database[1] (qui est une base de données traitant d'une maladie spécifique : le cancer du sein), et enfin des sources d'ordre privé telles que Abcam®[2] qui recensent des antigènes et réactifs. Près de 48 000 cartes de gènes, résumant de nombreuses informations disponibles sur un gène donné, sont décrites et le stockage de ces données est réalisé dans des fichiers de types texte et XML. Les données présentes dans GeneCards sont extraites périodiquement de manière automatique grâce à des scripts Perl développés localement. Sur le site Internet, il est possible d'interroger la source avec n'importe quel mot-clé et un ensemble de symboles de gènes reliés à ce mot-clé sont proposés.

La source **GeneLoc** [Rosen 03] fournit une carte intégrée du génome humain. Plus précisément, elle offre une vue combinée de gènes, leurs marqueurs, leurs séquences génomiques et leur position absolue. Des références croisées avec des sources de référence, telles que GeneCards et Entrez Gene, sont disponibles. GeneLoc contient plus de 41 000 gènes et plus de 105 000 marqueurs stockés dans une base de données relationnelle. Ces informations sont obtenues d'ensemble de gènes récupérés dans les sources Ensembl[3] et Entrez Gene. L'algorithme consiste à comparer ces gènes pour les unifier (assignation d'un même identifiant si une similarité existe), en exploitant des références croisées identiques ou encore des localisations chromosomiques proches. L'interrogation de GeneLoc peut se faire de différentes manières, notamment en entrant le numéro d'un chromosome dont on veut obtenir la carte intégrée ou encore avec un nom de gène pour lequel on veut connaître les marqueurs existants.

La source **HGMD** (Human Gene Mutation Database) [Cooper 98] est une collection de données sur des mutations de gènes causant des maladies humaines héréditaires. On peut citer par exemple des substitutions d'une base, ou encore des délétions, duplications et insertions. Des références croisées vers des bases de données bibliographiques, comme PubMed, GDB et OMIM, sont disponibles. HGMD contient environ 2 000 gènes et plus de 53 000 mutations stockés dans une base de données relationnelle. L'acquisition des données a été réalisée par une combinaison de procédures manuelles et automatisées appliquées à des publications. On peut interroger la

[1] http ://condor.bcm.tmc.edu/ermb/bcgd/bcgd.html
[2] http ://www.abcam.com/
[3] http ://www.ensembl.org/index.html

source HGMD avec un symbole ou nom de gène, un nom de maladie ou encore des numéros d'accession GDB ou OMIM.

Le Human Gene Nomenclature Committee maintient la source **HGNC** (appelée Genew pendant quelques années) [Eyre 06] qui fournit une nomenclature de symboles et noms de gènes officiels. Des références croisées vers d'autres sources de référence, comme GeneCards, OMIM, MGI sont présentes. HGNC contient plus de 23 000 gènes (un symbole et un nom par gène) dans une base de données relationnelle. Cette nomenclature est constituée manuellement mais intègre désormais des données publiques et confidentielles soumises par des chercheurs indépendants ou projets plus larges. Ces dernières sont contrôlées par le comité avant d'être ajoutées à la source. Il est possible d'interroger la source directement sur le site Internet au moyen d'un mot-clé ou d'un symbole de gène permettant d'accéder à la fiche du ou des gènes associés.

La source **MGD** (Mouse Genome Database) [Blake 06] intègre des informations génomiques, génétiques, fonctionnelles et phénotypiques sur les gènes et produits de gènes de la souris. Des liens vers des sources de référence, comme InterPro et Entrez Gene, sont disponibles. Ce modèle a été réalisé pour faciliter l'utilisation des données obtenues chez la souris, modèle privilégié de la physiologie et de la pathologie. MGD contient plus de 30 000 gènes stockés dans une base de données relationnelle. Les données sont entrées par des experts pour ce qui concerne la littérature biomédicale alors que les informations récupérées et échangées, de manière hebdomadaire, avec des ressources génomiques majeures, telles que UniProt et OMIM, sont obtenues automatiquement et ensuite validées par les experts. Plusieurs possibilités sont offertes pour interroger MGD, notamment par un nom ou symbole de gène ou encore un nom de maladie.

4.1.1.2 Les sources protéiques

La base de données **HPRD** (Human Protein Reference Database) [Mishra 06] est une plateforme permettant de représenter et d'intégrer des informations comme les fonctions de protéines du protéome humain, les modifications apparaissant après la traduction et les interactions protéines-protéines existantes et des implications de gènes associés dans des maladies. La source offre des liens vers d'autres sources biomédicales de référence, telles que Entrez Gene ou SwissProt. HPRD contient environ 20 000 entrées de protéines qui sont stockées dans une base de données objet. Toutes les informations présentes dans HPRD sont extraites manuellement de la littérature par des experts biologistes qui lisent, interprètent et analysent les données publiées. Sur Internet, l'interrogation du système peut se faire par différents types de données (nom de protéine, de maladie, symbole de gène, etc) et le résultat présente le ou les protéines concernées par cette requête.

La source **InterPro** (Integrated resource of Protein) [Mulder 05] est une ressource documentaire fournissant des familles de protéines, leurs domaines et sites fonctionnels afin d'intégrer les sources majeures de signatures de protéines, telles que PIR[4], PROSITE[5] ou encore PRINTS[6].

[4]http ://pir.georgetown.edu/
[5]http ://www.expasy.org/prosite/
[6]http ://bioinf.man.ac.uk/dbbrowser/PRINTS/

Elle fournit également des références croisées vers des sources de référence, comme PubMed et PDB. InterPro contient un peu plus de 13 000 entrées (comprenant des domaines, familles, sites actifs, etc) stockées dans une base de données relationnelle. Les signatures sont intégrées à InterPro manuellement par des experts qui les vérifient et les corrigent si cela est nécessaire. On peut interroger la source en entrant du texte libre ou une séquence.

La source **PDB** (Protein structure DataBank) [Deshpande 05] est un entrepôt visant à répertorier les structures tri-dimensionnelles de macro-molécules biologiques. Des données additionnelles sont aussi disponibles : des relations avec les séquences génomiques et protéomiques, les fonctions biologiques, les localisations cellulaires ainsi que les maladies associées. Ces dernières informations sont collectées et intégrées à partir de ressources externes, telles que KEGG, Gene Ontology et PubMed. Les références croisées correspondantes sont malgré tout présentées aux utilisateurs qui peuvent accéder aux autres informations existant dans ces ressources mais qui ne sont pas stockées dans PDB. Celle-ci contient plus de 37 000 entrées qui sont stockées dans une base de données relationnelle, mise à jour de manière hebdomadaire. Là encore, le processus d'intégration des données dans PDB est semi-automatique, des programmes permettent la collecte d'informations dans les ressources externes qui sont ensuite validées par des experts. Enfin, il est possible d'interroger PDB sur son site Web en faisant une requête en texte libre ou au moyen de services Web.

La source **Swiss-Prot** [Boeckmann 03] fournit des séquences protéiques et à chaque protéine est associée au minimum son nom, sa séquence d'acides aminés ainsi que des données taxonomiques et bibliographiques. Des données supplémentaires sont parfois disponibles, par exemple la fonction de la protéine, sa structure, des similarités existant avec d'autres protéines et des maladies associées à des déficiences de cette protéine. Des références croisées vers de nombreuses ressources existent, notamment vers des entrepôts de séquences génomiques comme GenBank ou encore des sources de structures protéiques telles que PDB. Swiss-Prot contient plus de 227 000 entrées qui sont stockées dans des fichiers de type texte. Les informations répertoriées dans la source sont récupérées (et mises à jour très régulièrement) dans des publications reportant des nouvelles données de séquences mais aussi dans des articles révisés pour vérifier que certaines annotations sont encore valides. Ces tâches sont réalisées semi-automatiquement par des outils qui permettent d'extraire des informations à partir de sources bibliographiques que des experts annotateurs vérifient, valident, voire même complètent manuellement. L'interrogation peut se faire sur du texte libre et les entrées associées sont ensuite proposées.

4.1.1.3 Les sources de données médicales

La source **OMIM** (Online Mendelian Inheritance in Man) [Hamosh 05] est un catalogue de gènes humains et de maladies génétiques développé pour assister les chercheurs et enseignants en génomique humaine et à la pratique de la génétique clinique. Chaque entrée est un résumé en texte libre d'un phénotype et/ou d'un gène génétiquement déterminés et présente de nombreux liens vers d'autres sources de référence, comme des sources de séquences telles que GenBank, de références bibliographiques au travers de PubMed ou encore de mutations dans HGMD. Trois types d'entrées principales existent : celles dont l'identifiant est précédé du caractère *

correspondent à un gène dont la séquence est connue, un # indique une entrée descriptive d'un phénotype ou d'une famille de gènes et enfin, si aucun symbole ne précède l'identifiant OMIM, cela signifie que le mode de transmission n'a pas été prouvé ou bien que la distinction entre ce phénotype et un autre n'est pas claire. OMIM contient environ 17 000 entrées stockées dans une base de données relationnelle. Comme la plupart des sources présentées, l'intégration de données au sein d'OMIM se fait de manière semi-automatique ; les informations bibliographiques et autres (plus générales comme le nom d'un gène et son symbole) sont extraites de ressources externes automatiquement puis validées par des experts (qui peuvent bien entendu les compléter ou les modifier si nécessaire) avant d'être ajoutées. L'interrogation peut se faire, comme pour Entrez Gene, au travers du portail Entrez, avec du texte libre ou plus précis (par exemple, avec un symbole de gène) et les entrées correspondantes sont listées.

4.1.1.4 Conclusion

Nous avons choisi d'intégrer virtuellement des sources traitant aussi bien de données génomiques, protéiques et cliniques. Ces sources sont donc **complémentaires** mais il faut également souligner que certaines d'entre elles se recoupent, notamment GeneLoc qui est étroitement liée à GeneCards. Cette dernière inclut également diverses informations qui ont été extraites d'autres sources, telles que Swiss-Prot dans laquelle GeneCards récupère des données, comme la fonction de gènes ou encore les domaines de protéines, qu'elle intègre et propose ensuite dans chacune de ses cartes de gènes. Il y a donc une certaine **redondance** entre les informations fournies par les diverses sources de données biomédicales. C'est pour répondre à ce type de constat que Sujansky a souligné l'intérêt de réaliser une intégration *verticale* (en opposition à *horizontale* qui correspond à l'aspect de complémentarité offert par des sources distinctes) dont le but est de fusionner les données similaires récupérées dans les diverses sources intégrées dans un même système [Sujansky 01]. L'exploitation du recouvrement de données peut s'avérer utile pour différentes fonctionnalités, comme le contrôle sur la complétude ou sur l'incohérence de ces données dans certaines sources, rendant le système plus fiable.

Les schémas de ces sources ne sont, pour la plupart, pas disponibles sur Internet dans un format structuré. En effet, seul un descriptif des différents champs contenant des données est proposé, en général dans un tableau au format HTML (par exemple, HGNC et HGMD), ce qui n'est pas très facilement exploitable. D'autres sources fournissent leur schéma sur leur site, mais leur format est uniquement lisible par les hommes et donc inexploitable par les machines. C'est le cas notamment de HPRD et InterPro. Enfin, certaines sources, telles que GeneCards ou les sources du NCBI, s'efforcent d'offrir aux utilisateurs des descriptions détaillées de leur schéma dans un format structuré.

L'ensemble des informations concernant ces onze sources est récapitulé dans le tableau 4.1 page suivante. Les noms des sources sont indiqués puis leur catégorie selon la classification fournie dans l'édition spéciale sur les bases de données biomédicales du journal Nucleic Acids Research du 1er janvier 2006 [Galperin 06]. Sont également donnés un bref descriptif des sources, l'URL de leur site Web, le type d'implémentation qu'elles utilisent et enfin l'URL des schémas, quand ils sont disponibles, ainsi que leur format.

TAB. 4.1 – Sources de données intégrées au système

Nom de la source	**Nom complet**	**Catégorie**	**Bref descriptif**	**URL**	**Implémentation**	**Schéma**
Entrez Gene	-	Séquences d'ADN codantes et non codantes	Informations généralistes (motifs, site de régulation, etc) entrées sur le gène, créé par le NCBI	http ://www.ncbi.nlm.nih.gov/ entrez/query.fcgi ? db=gene	Base de données relationnelle	http ://www.ncbi.nlm.nih.gov/data_specs/dtd/ (DTD)
GeneCards	-	Base de données de génome humain	Base de données intégrées de gènes humains, cartes, protéines et maladies	http ://bioinfo.weizmann.ac.il/cards/	Fichiers texte et XML	http ://www.genecards.org/GeneCardByFunction.xsd (XSD)
GeneLoc	-	Base de données de génome humain	Base de données présentant une carte intégrée pour chaque chromosome humain	http ://genecards.weizmann.ac.il/geneloc/	Base de données relationnelle	-
HGMD	Human Gene Mutation Database	Base de données de mutations humaines	Représente une collection de données sur les mutations de gènes causant ou étant associées à une maladie humaine et héréditaire	http ://www.hgmd.org/	Base de données relationnelle	http ://www.hgmd.cf.ac.uk/ac/index.php (HTML)
HGNC	Human Gene Nomenclature Committee Database	Nomenclature de gènes	Base de données du HGNC qui fournit des données pour tous les gènes humains qui ont des symboles officiels	http ://www.gene.ucl.ac.uk/ nomenclature	Base de données relationnelle	http ://www.gene.ucl.ac.uk/cgi-bin/ nomenclature/gdlw.pl (HTML)
MGD	Mouse Genome Database	Modèles d'organismes, génomique comparative	Base de données fournissant un accès intégré à des données génétiques, génomiques et biologiques sur les souris de laboratoire	http ://www.informatics.jax.org/	Base de données relationnelle	http ://www.informatics.jax.org/schema/ (HTML)

Nom de la source	Nom complet	Catégorie	Bref descriptif	URL	Implémentation	Schéma
HPRD	Human Protein Reference Database	Protéines humaines	Plate-forme centralisée contenant de l'information sur l'architecture de domaine, les modifications post-traductionnelles et la maladie associée	http ://www.hprd.org	Base de données objet	http ://www.hprd.org /schema (UML)
InterPro	Integrated resource of Protein	Domaines de protéines	Base de données de familles de protéines, domaines et sites fonctionnels	http ://www.ebi.ac.uk/interpro	Base de données relationnelle	http ://www.ebi.ac.uk/interpro/interpro_schema_diagram.pdf (PDF)
PDB	Protein structure DataBank	Structures de protéines	Banques de données contenant les structures 3D de protéines et acides nucléiques disponibles publiquement	http ://www.rcsb.org/pdb	Base de données relationnelle	http ://www.rcsb.org/pdbschema/ (HTML)
Swiss-Prot	-	Séquences protéiques d'ordre général	Catalogue d'information sur les protéines	http ://www.expasy.org/sprot/	Fichiers texte	http ://www.expasy.org/sprot/userman.html#linetypes (HTML)
OMIM	Online Mendelian Inheritance in Man	Base de données générale de gènes humains et maladies	Catalogue des maladies humaines, d'ordre génétique et génomique	http ://www.ncbi.nlm.nih.gov/ entrez/query.fcgi ? db=OMIM	Base de données relationnelle	http ://www.ncbi.nlm.nih.gov/ data_specs/dtd/ (DTD)

4.1.2 Ressources terminologiques

4.1.2.1 L'UMLS

Le projet **UMLS** (Unified Medical Language System) [Lindberg 93] a été développé par la National Library of Medicine dans le but de faciliter la recherche et l'intégration d'informations provenant de sources aussi diverses que des bases bibliographiques médicales, des dossiers cliniques, des bases de connaissances, etc. L'accès à ces multiples bases se heurte à l'obstacle que constitue la multiplicité des terminologies et classifications, chacune d'elles ayant été développée pour un objectif spécifique. Le but de l'UMLS est de dépasser ces particularismes par l'assimilation de l'ensemble de ces vocabulaires et le développement de liens entre concepts, que ceux-ci aient leur origine dans la même nomenclature ou qu'ils proviennent de sources différentes. Elle inclut trois sources d'information sémantique : le Metathesaurus, le Réseau Sémantique (Semantic Network) et les ressources lexicales (Lexical Resources). La version 2005AA de l'UMLS est utilisée dans ce travail.

Le **Metathesaurus**, intégrant plus de 100 vocabulaires sources, constitue un large graphe comprenant un peu plus d'un million de nœuds (concepts) et 22 millions de relations entre ces concepts. Les concepts sont constitués de termes synonymes provenant des divers vocabulaires sources, présentant ainsi un sens unifié pour l'ensemble des termes. Chaque concept a un identifiant unique, appelé CUI (Concept Unique Identifier) et parfois une définition ou annotation associée. Par exemple, le concept ***Chromosomes*** a pour CUI ***C0008633*** et pour définition « *In a prokaryotic cell or in the nucleus of a eukaryotic cell, a structure consisting of or containing DNA which carries the genetic information essential to the cell (From Singleton & Sainsbury, Dictionary of Microbiology and Molecular Biology, 2d ed)* ». La quantité importante de relations présentes dans le Metathesaurus est due au fait que, par convention, l'ensemble des relations existant dans les vocabulaires sources doivent être intégrées dans l'UMLS, pour éviter la perte d'informations pertinentes spécifiques à un vocabulaire source donné dans son contexte propre. La plupart des relations hiérarchiques présentes dans le Metathesaurus sont valides, comme par exemple, la relation PAR (`est-père-de`) entre ***Neoplasms (C0027651)*** et ***Malignant neoplasm of breast (C0006142)***, provenant de plusieurs sources. Pourtant, certains vocabulaires utilisent, pour organiser leurs termes hiérarchiquement, des relations qui ne sont pas vraiment hiérarchiques. Par exemple, dans le thesaurus Alcohol and Other Drug[7] (AOD), une relation hiérarchique existe entre les concepts ***biological rest (C0678686)*** et ***Fatigue (C0015672)***, qui est utile pour la recherche d'informations mais ne correspond en réalité pas à une relation hiérarchique. Cependant, cette relation est préservée comme telle (relation PAR) dans le Metathesaurus. En effet, par convention, lors de l'intégration dans l'UMLS, toutes les relations utilisées pour organiser les vocabulaires sources hiérarchiquement participent à la structure hiérarchique du Metathesaurus. Enfin, d'autres types de relations existent et nous les qualifions d'associatives [Zhang 04]. Par exemple, une relation étiquetée `other-relation` relie les concepts ***Malignant neoplasm of breast (C0006142)*** et ***Mastectomy (C0024881)***, indiquant qu'un certain lien existe, même s'il est imprécis, entre la présence d'une tumeur ma-

[7] http ://etoh.niaaa.nih.gov/aodvol1/aodthome.htm

ligne au sein et son ablation.

Le **Réseau Sémantique** est beaucoup plus restreint et comporte un ensemble de 135 catégories larges nommées types sémantiques (Voir l'annexe A - page 208 pour la liste complète) et qui permettent de fournir une catégorisation cohérente de tous les concepts du Metathesaurus ainsi qu'un ensemble de 54 relations sémantiques qui existent entre les types sémantiques. Les relations de type *est-un* permettent d'organiser le réseau de manière arborescente. D'autres types de relations associent les types sémantiques entre eux, par exemple *traite*. Chaque concept du Metathesaurus est donc catégorisé par au moins un type sémantique du Réseau Sémantique, indépendamment de sa position hiérarchique dans le vocabulaire dont il est issu. Dans tous les cas, c'est le type sémantique le plus spécifique existant dans la hiérarchie qui est assigné au concept. Enfin, un niveau d'agrégation supplémentaire a été ajouté au-dessus des types sémantiques en les regroupant sous quinze groupes sémantiques afin de réduire la complexité conceptuelle [McCray 01] : Activities & Behaviors, Anatomy, Chemicals & Drugs, Concepts & Ideas, Devices, Disorders, Genes & Molecular Sequences, Geographic Areas, Living Beings, Objects, Occupations, Organizations, Phenomena, Physiology, Procedures. Ces groupes sémantiques peuvent notamment fournir une vue d'ensemble plus large de l'espace conceptuel formé par l'UMLS et servir à identifier des incohérences dans la représentation faite du domaine biomédical. Nous donnons un exemple de ce type d'utilisation dans la partie Résultats (cf 5.4 page 139).

Les **ressources lexicales** comprennent le SPECIALIST Lexicon et des outils lexicaux [McCray 94]. Le premier est un lexique d'informations syntaxique, morphologique et orthographique sur des mots courants de langue anglaise et sur le vocabulaire biomédical dans le but d'être utilisé pour le traitement du langage naturel. Les outils lexicaux utilisent le lexique pour aider les utilisateurs à détecter et supprimer l'inflexion possible des concepts (par exemple, les formes au singulier et au pluriel) et autres dérivés (par exemple, la forme adjectivale d'un nom ou le substantif d'un adjectif), la casse et les variations dans l'ordre des mots. En effet, l'API du développeur disponible sur le site de l'UMLS[8], entre autres, propose différentes méthodes pour identifier des concepts du Metathesaurus dans des termes fournis en entrée par des recherches exactes et normalisées. Le processus permet notamment de rendre compatibles les textes (ou listes de termes) donnés en entrée par rapport aux termes cibles en supprimant, par exemple, des éléments sémantiquement peu importants, tels que les tirets bas ou variations de virgules ainsi que les mots non pertinents (*stop-words* en anglais). Le programme MetaMap (MetaMap Transfer ou MMTx) [Aronson 01], lui, permet d'extraire des concepts du Metathesaurus d'un texte. L'entrée est un texte quelconque de n'importe quelle taille et le format de sortie est un fichier texte répertoriant la liste des concepts associés avec chaque partie du texte. MetaMap réalise des mises en correspondance exactes, normalisées mais aussi approximatives en exploitant les variants des termes et permettant de découvrir des correspondances partielles. Plus précisément, le texte en entrée est traité par une série de modules. D'abord, il est découpé en composants, tels que des phrases, paragraphes, éléments lexicaux et tokens. Les variants sont ensuite générés à partir des phrases résultant de la première étape. Des concepts candidats sont retrouvés et

[8] http ://umlsks.nlm.nih.gov/kss/servlet/Turbine/template/docs,Capi,CapiDownload.vm

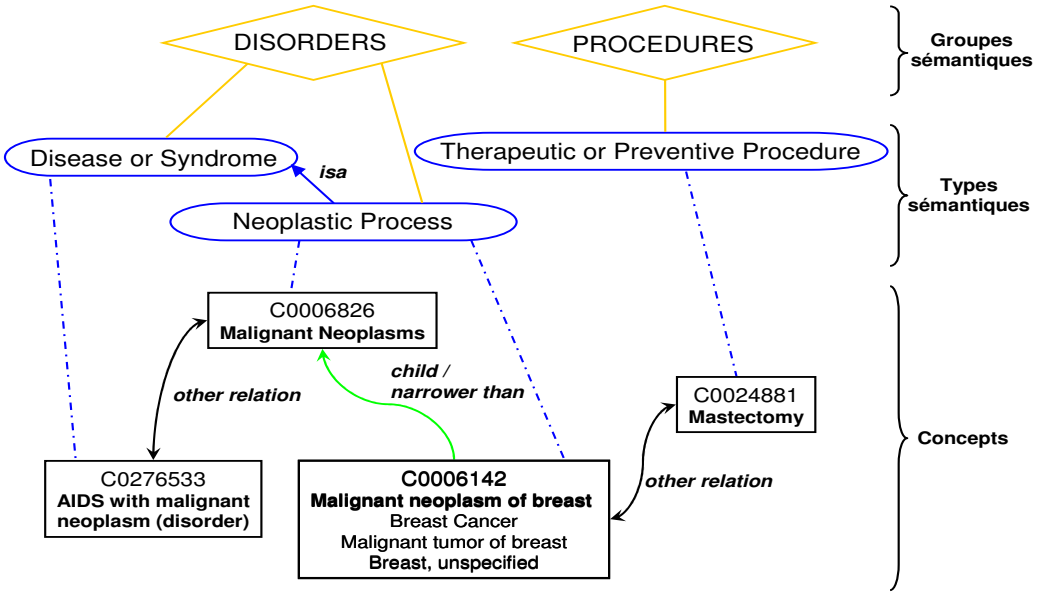


FIG. 4.1 – **Portion de l'UMLS.** Les éléments rectangulaires font partie du Metathesaurus, les relations hiérarchiques et associatives sont représentées par des flèches, les traits pointillés correspondent à la catégorisation des concepts à un (ou plusieurs) type(s) sémantique(s) et les traits pleins correspondent à l'association types/groupes sémantiques.

évalués de manière à donner un score de correspondance par rapport aux phrases. Les meilleurs candidats sont finalement déterminés, choisis et organisés de manière à correspondre au mieux au texte.

La figure 4.1 illustre une portion de l'UMLS constituée de quatre concepts, les relations hiérarchiques et associatives existant entre eux, les trois types sémantiques auxquels ils sont associés et les différents types de relations existant entre ces derniers ainsi que le groupe sémantique regroupant ces types sémantiques.

4.1.2.2 WordNet

WordNet (WN*) [Miller 98] est une base de données lexicale de langue anglaise. Des noms, verbes, adjectifs et adverbes sont organisés en ensembles de synonymes, appelés synsets, chacun représentant un concept sous-jacent. Notons qu'en cas de polysémie, un terme peut appartenir à plusieurs synsets. Elle contient plus de 155 000 items lexicaux regroupés dans près de 117 000 synsets. La version 2.1 de WN est utilisée dans ce travail.

Les synsets sont liés entre eux par des relations binaires qui diffèrent suivant les quatre catégories syntaxiques pré-citées couvertes par WN. Les synsets de type « nom » sont organisés suivant la relation hiérarchique *est-un* et la relation de composition *partie-tout* (*meronym*).

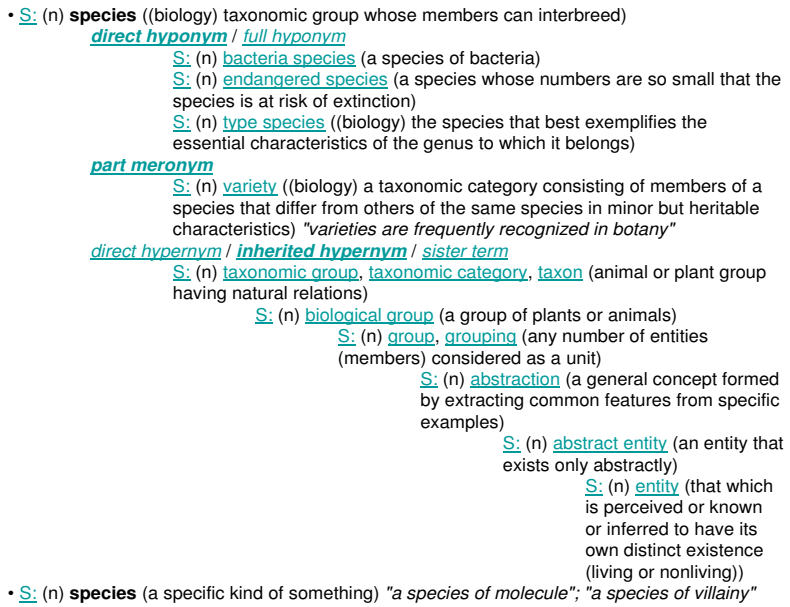


FIG. 4.2 – **Exemple des synsets WordNet associés au mot `Species` (Espèce).** Deux sens différents existent dont l'un concerne le domaine de la biologie (le domaine sémantique est indiqué entre parenthèses au début de la définition). Seules les propriétés associées au premier synset sont détaillées : ses trois hyponymes directs (direct hyponym), le synset dont il fait partie (part meronym) et enfin l'ensemble de ses hypernymes indentés en fonction de leur niveau dans la hiérarchie (inherited hypernym).

Les verbes sont connectés par des relations hiérarchiques et d'implication, telles que *`montre`*. Les adverbes et adjectifs sont liés par des relations de synonymie et d'antonymie. Ainsi, les lexiques de noms et verbes sont structurés en hiérarchie, sachant que celle des noms est beaucoup plus profonde que celle des verbes. Dans le vocabulaire utilisé par WN, les synsets plus généraux et plus spécifiques sont respectivement nommés **hypernymes** et **hyponymes**. À chaque synset est associée une définition en texte libre donnant son sens et pouvant également inclure des exemples d'utilisation des termes constituant le synset ainsi que le domaine que ce dernier recouvre. 165 domaines sémantiques (Voir l'annexe B - page 209) pour la liste complète) ont été définis de manière à indiquer le contexte d'utilisation des mots. Certains synsets ne sont pas associés à un domaine précis puisqu'ils ne touchent pas à un domaine particulier. La figure 4.2 présente un exemple de synsets WN ainsi que les propriétés qui y sont associées.

WN offre un certain nombre de fonctionnalités aux utilisateurs. Il est possible soit d'interroger WN en ligne, soit de le télécharger et utiliser ensuite des outils existants pour requêter et exploiter la base de données lexicale. En particulier, le programme *wn* permet de rechercher des synsets dans du texte libre. Bien que seules les formes de base de chaque mot soient généralement stockées dans WN, les utilisateurs peuvent faire leurs recherches sur des formes dérivées puisqu'un traitement morphologique est appliqué sur la chaîne de caractères fournie en entrée pour générer une forme présente dans WN (par exemple, la forme de base de « names » est « name »).

Depuis des années, WN est largement utilisé par la communauté informatique du traitement du langage naturel, qui a contribué à rendre la base de données plus volumineuse et à en améliorer la modélisation afin d'en faire un outil fonctionnel de choix. Son attrait principal est sa capacité à désambiguïser les mots grâce à leur sens. En effet, de nombreux mots sont polysémiques. L'ambiguïté de ces mots peut être résolue en fonction des liens qu'ils ont avec d'autres mots au sein de WN. L'environnement de chaque mot sert alors à déterminer le synset auquel il fait référence.

4.2 Méthodes

Dans cette section, nous présentons les méthodes que nous avons mises en œuvre pour créer notre système de médiation. Tout d'abord, nous définissons un certain nombre de notions que nous utilisons pour décrire et présenter ce travail. Dans un deuxième temps, nous décrivons le corpus de gènes que nous avons constitué de manière à interroger les sources automatiquement et y récupérer des informations pertinentes concernant leur contenu. Nous présentons ensuite une méthode permettant d'acquérir un schéma pour chaque source de données à intégrer. Puis nous définissons le schéma global issu de l'UMLS. Enfin, à partir des termes et concepts nécessaires pour la description des schémas des sources, nous décrivons les approches développées pour mettre en correspondance les éléments des schémas locaux avec ceux du schéma global.

4.2.1 Définitions

Il existe différentes manières de nommer les concepts situés au-dessus et au-dessous dans la hiérarchie d'un concept C donné, sachant que l'on inclut non seulement les concepts situés directement au-dessus (respectivement au-dessous), mais aussi ceux situés à un niveau encore supérieur (respectivement inférieur) de manière récursive. Les termes plus généraux (respectivement plus spécifiques) ou encore super-concepts (respectivement sous-concepts) sont généralement utilisés. Cependant, cette terminologie sous-entend que les relations qui lient C aux autres concepts sont de type *is-a* ou encore de généralisation (respectivement spécialisation). Or, nous avons vu que les relations hiérarchiques représentées dans l'UMLS (cf 4.1.2.1 page 97), sont en réalité de différents types (*parent*, *plus-proche-de*, etc). Dans ce cadre, nous avons choisi de définir d'une autre manière les super-concepts et les sous-concepts de C :

Définition 1 *Les concepts plus généraux et plus spécifiques d'un concept donné seront définis dans l'UMLS comme ses* ***ascendants*** *(ou ancêtres) et ses* ***descendants****, respectivement.*

Comme nous l'avons défini dans l'état de l'art (cf 2.3 page 71), l'opération de mise en correspondance de schémas de sources de données prend en entrée deux schémas (représentant chacun une source), chacun consistant en un ensemble d'éléments (classes, attributs, relations, etc), et détermine les relations (équivalence, subsomption, etc) existant entre ces éléments. Rappelons que ces éléments ont été définis dans l'état de l'art (cf 2.2.1.1 page 33). Dans ce travail, nous

restreignons cette définition de la manière suivante :

Définition 2 *L'opération de mise en correspondance de schémas de sources de données prend en entrée deux schémas (représentant chacun une source) et détermine les relations d'équivalence existant entre les attributs des schémas.*

4.2.2 Constitution d'un corpus de gènes pour l'interrogation des sources

Pour interroger dynamiquement les sources à intégrer dans notre système, nous avons besoin de *points d'entrée*, c'est-à-dire de termes pour lesquels on récupère des informations au sein des sources.

Or nous avons constaté que les noms et symboles de gènes sont suffisants pour interroger l'ensemble des sources. En effet, les onze sources peuvent être interrogées avec du texte libre, garantissant que même si leur entrée ne traite pas principalement des gènes (comme Swiss-Prot qui est centré sur les protéines), les sources fournissent quand même des informations concernant les gènes de manière indirecte, notamment les pathologies dans lesquelles ces derniers sont impliqués ou encore les protéines qui leur sont associées. Nous avons ainsi créé un corpus de gènes à l'aide d'une ressource de référence dans le domaine biomédical : Genetics Home Reference[9] (GHR). Cette dernière fournit des informations concernant les gènes impliqués dans les maladies génétiques. L'avantage de cette ressource est qu'elle contient des données connues et donc pour lesquelles on s'attend à ce que les sources contiennent des entrées les concernant. Un échantillon de 100 paires (symbole d'un gène, nom associé), extraites de manière aléatoire de GHR, constitue notre corpus. Des exemples de paires sont (HFE, hemochromatosis) et (BRCA1, breast cancer early onset) et le détail de l'ensemble est donné en annexe C (page 215).

4.2.3 Étape 1 : Acquisition des schémas locaux

Nous cherchons à acquérir le schéma des sources de données que l'on va intégrer à notre système. Pour cela, nous souhaitons mettre en œuvre une méthode qui soit le plus automatique possible de manière à minimiser au maximum les tâches manuelles de conception du système. Nous présentons cette méthode dans la partie qui suit en commençant par définir la notion d'éléments de données, puis nous détaillons l'algorithme d'extraction d'éléments de données à partir de sources biomédicales. Ensuite, nous complétons cet algorithme par l'attribution d'un type à ces éléments. Enfin, nous décrivons le contenu des schémas des sources.

4.2.3.1 Définition des éléments de données

Les **éléments de données** (EDs*) (*Data Elements* en anglais) peuvent être définis ainsi : « une unité d'information de base construite sur des structures standard ayant un sens unique et des valeurs distinctes[10] ». Cette notion a été introduite par la norme ISO/IEC 11179[11] qui a

[9] http ://ghr.nlm.nih.gov/

[10] http ://www.atis.org/tg2k/_data_element.html

[11] La normalisation et l'enregistrement des éléments de données tels qu'ils sont décrits dans ISO/IEC 11179 permettent de créer, en beaucoup moins de temps et avec beaucoup moins d'efforts que les méthodes conventionnelles de gestion de données, un environnement propre au partage des données, http ://metadata-standards.org/11179/

pour thématique les registres de méta-données comme technologies de l'information. Elle explique comment normaliser et enregistrer les EDs de manière à assurer la lisibilité et l'interchangeabilité des données.

Le « National Cancer Institute » (NCI) a créé un entrepôt de données standard pour le cancer ou **caDSR** (pour Cancer Data Standards Registry)[12] dans le cadre du projet caCORE visant à définir une infrastructure commune concernant les outils et modélisations informatiques pour le cancer [Covitz 03]. Il mutualise différents efforts, destinés à fournir un accès global et centralisé à l'ensemble de ces informations. Son objectif principal est de définir un ensemble de descripteurs de méta-données standard et compréhensibles pour représenter la terminologie utilisée dans la recherche sur le cancer, qui est utilisée pour la collecte et le traitement de l'information. De nombreux groupes du NCI et organisations partenaires ont permis le développement du caDSR, en définissant des EDs basés sur et issus de standards de données, de bases de données diverses, d'applications cliniques, de formats de données, modèles UMLS et vocabulaires. Conformément à la norme ISO/IEC 11179, des informations concernant les noms, définitions, valeurs possibles et concepts sémantiques pour chaque élément de données ont été incluses dans le caDSR. Nous avons étudié l'éventuelle utilisation de cet entrepôt dans notre système mais il n'était pas adapté pour représenter les EDs dont nous disposions [Mougin 06b], [Mougin 06c].

Des exemples d'EDs dans le domaine biomédical incluent `Gene Symbol` et `Pathology Name`. Les ensembles de valeurs correspondantes seraient respectivement un ensemble de symboles de gènes (par exemple, dans un organisme précis) et un ensemble de maladies données. La partie suivante décrit la manière dont nous avons extraits les EDs des sources à intégrer.

4.2.3.2 Extraction des éléments de données

De nombreuses recherches ont été effectuées dans le but d'extraire des informations à partir de documents Web. Le terme généralement utilisé est la **génération de wrapper** (ou *wrapper induction*) [Kushmerick 97] où la notion de *wrapper* est différente de celle traduite par adaptateur comme composant des systèmes de médiation. Ici, ce terme correspond à un programme qui se charge de parcourir une page Web afin d'en inférer la structure, et donc une grammaire pour le code HTML. Un bon rapport recensant les différentes techniques d'extraction d'informations développées pour l'induction de wrapper est donné dans [Eikvil 99]. Nous ne rentrons pas dans les détails ici mais donnons quelques exemples d'approches et de travaux proposés dans ce cadre. Nous montrons en quoi ils ne répondent pas à nos besoins et en conséquence pourquoi nous avons développé notre propre méthode. Nous présentons ensuite l'approche que nous proposons.

4.2.3.2.1 Travaux existants en *wrapper induction*. Des approches diverses ont été proposées afin d'extraire la structure d'un document HTML, et ce pour des objectifs différents. Certaines exploitent la connaissance d'une ontologie pour identifier les méta-données reflétant le document HTML considéré [Stuckenschmidt 01]. Cette méthode nécessite une intervention

[12] http ://ncicb.nci.nih.gov/NCICB/infrastructure/cacore_overview/cadsr

humaine pour définir l'ontologie source et les règles permettant ensuite l'identification des méta-données au sein des pages Web par rapport aux concepts pertinents dans l'ontologie. D'autres approches utilisent des techniques d'apprentissage nécessitant au préalable d'étiqueter manuellement un certain nombre de pages HTML à partir desquelles il sera possible d'extraire automatiquement les données relatives à la structure du document HTML [Soderland 97], [Kushmerick 97]. Le problème principal de ces approches est qu'elles requièrent l'existence d'informations additionnelles par rapport aux pages HTML à traiter mais surtout qu'elles nécessitent une intervention manuelle, aspect que nous souhaitons éviter.

Certains travaux proposent cependant des approches automatiques. On citera d'abord InfoDiscoverer, un système qui identifie automatiquement les parties pertinentes de pages HTML d'un même site Web [Lin 02]. Les auteurs se basent sur la similarité intra-pages présente dans un même site Web pour éliminer cette information redondante et ne garder que ce qui est différent d'une page à l'autre considérant que seule cette information est pertinente. Nous souhaitons extraire les éléments communs d'une page à l'autre, ce qui revient au travail inverse du leur : garder uniquement les informations communes entre les pages de notre ensemble.

Le deuxième système que nous souhaitons présenter est RoadRunner [Crescenzi 01]. Les auteurs se basent sur le fait que des pages HTML générées à partir du contenu de bases de données sont produites à partir de scripts et ont la même structure. Le principe suivi consiste à d'abord transformer les deux pages HTML fournies en entrée en listes de tokens (lemmatisation). Le contenu de la première page sert de référence et la seconde est parcourue pour être comparée à l'autre. Les termes communs sont gardés ensemble et il faut une intervention manuelle pour indiquer le libellé à leur associer tandis que les différences permettent d'identifier des champs optionnels. De plus, ce travail étudie les valeurs différentes associées à des termes communs d'une page à l'autre afin de définir un type pour le champ correspondant. Nous ne voulons pas que les éléments extraits des sources soient étiquetés manuellement car, dans les sources biomédicales, ils correspondent en soi aux libellés qui nous intéressent. Cela est un avantage des sources de données dans ce domaine. Elles ont, en effet, été développées pour être consultées par des biologistes et médecins et donc pour lesquelles les sites Web fournissent des informations étiquetées pour que les utilisateurs sachent à quel type d'informations ils vont accéder. L'inconvénient d'une extraction automatique est qu'elle peut résulter en des noms d'éléments peu appropriés, incohérents ou ambigus. Cependant, nous réglons ce problème en typant les valeurs associées aux EDs de manière plus précise que cela est réalisé dans RoadRunner (cf 4.2.3.4 page 108). D'autre part, l'utilisation de deux pages HTML dans l'approche de Crescenzi et al. n'est pas satisfaisante car on n'est pas certain (et c'est même peu probable) que les champs optionnels soient tous représentés dans ces deux pages uniquement. Dans notre approche, l'exploitation de 100 pages HTML est plus efficace même s'il faut noter qu'elle ne garantit pas non plus une entière exhaustivité.

4.2.3.2.2 Notre approche. Les sources de données que nous cherchons à intégrer au sein de notre système fournissent des interfaces Web permettant d'accéder à leur contenu via des requêtes. La plupart des sources sont implémentées à l'aide d'une base de données ou de fichiers

plats structurés. Des programmes CGI (Common Gateway Interface) vont interroger ces bases ou parcourir ces fichiers pour récupérer les données correspondant à la requête posée (enregistrements de tables ou contenu du fichier). Les résultats sont fournis sous la forme d'une page Web, où sont associés des noms plus généraux aux champs ou à des libellés particuliers (répétés) dans les fichiers : les noms externes. Ce sont ces noms externes qui nous intéressent et que nous cherchons à extraire car ils correspondent aux EDs. Par exemple, HGNC stocke ses données dans une base constituée d'une table unique (cf figure 2.4 page 35). Les champs de celle-ci ont des noms externes qui sont plus explicites que les noms de champs de la table. Des exemples de noms de champs incluent gd_app_sym et gd_pub_acc_ids dont les noms externes sont respectivement `Approved Symbol` et `Accession Numbers`.

Afin de repérer et d'extraire les EDs, notre méthode repose sur ce que l'on appelle la redondance intra-pages Web (cf figure 4.4 page 107). En effet, au sein d'une même source, des requêtes différentes permettent d'obtenir en retour des pages Web répétant les mêmes EDs [Mougin 04]. Nous exploitons cette caractéristique pour extraire de manière automatique les EDs dans un ensemble de pages HTML de chaque source.

Le processus que nous décrivons ci-dessous s'applique à chacune des sources de données à intégrer, indépendamment les unes des autres. Les différentes phases d'extraction des EDs de la source S sont les suivantes (Figure 4.3 page suivante) :

- interrogation de S pour les 100 gènes (nom ou symbole) du corpus (cf 4.2.2 page 102). Ce processus, entièrement automatique, utilise l'URL du programme CGI disponible sur le site Internet de S en lui fournissant en entrée les 100 gènes un par un. Cela permet de constituer un échantillon de 100 pages Web dont la structure HTML est identique ;
- un pré-traitement des 100 pages Web de l'échantillon est d'abord effectué : les pages sont nettoyées afin d'éliminer des informations non pertinentes (en-tête et pied de page). De plus, parmi les balises HTML, on distingue différentes catégories qui sont plus ou moins informatives. Ainsi, nous avons supprimé les balises HTML qui ne sont *a priori* pas pertinentes pour représenter la structure de la source. On donnera comme exemple la balise
 qui permet d'aller à la ligne, en comparaison à des balises telles que <H1> et <TABLE> qui sont, elles, susceptibles de contenir des éléments importants ;
- parallèlement, l'ensemble des liens hypertextes existant dans ces pages Web sont répertoriés dans une base de données, de manière à identifier automatiquement des références croisées entre S et d'autres sources qui pourront compléter la description de S ;
- dans chaque page, des paires associant une balise HTML à son contenu (termes) sont récupérées ;
- les paires (balise HTML, termes) présentes dans plus de 75% des pages Web de S sont sélectionnées. Cette sélection permet d'éliminer les données spécifiques telles qu'un nom de gène donné (par exemple, « HFE » ou « BRCA1 ») tandis qu'elle conserve l'information d'ordre général qui se retrouve dans chacune des pages Web de S, comme le terme `Gene Name`. Le terme correspond ainsi à un ED de la source S.

Nous avons choisi d'identifier des paires plutôt que seulement les termes récurrents dans les pages Web afin d'exploiter réellement les caractéristiques offertes par les programmes CGI.

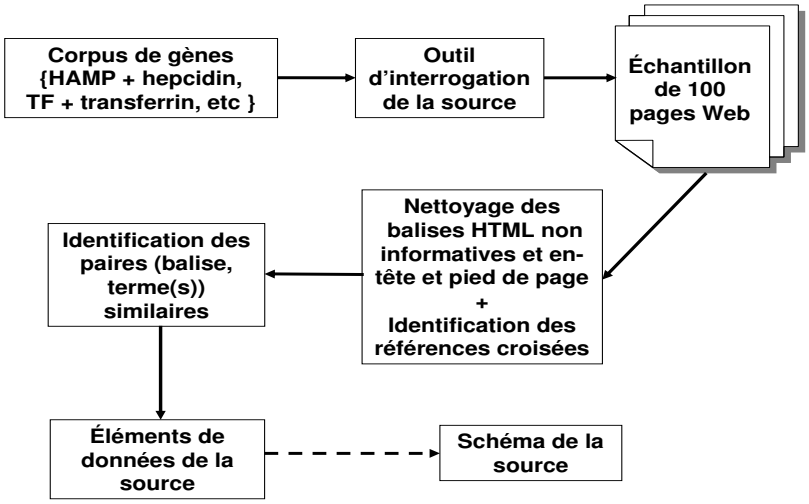


FIG. 4.3 – **Algorithme pour l'extraction d'éléments de données dans une source.** À partir d'un corpus de gènes, la source est interrogée au travers du programme CGI fourni par l'interface Web du site associé. Un échantillon de 100 pages Web est récupéré. Les données non pertinentes sont d'abord nettoyées pour identifier ensuite les paires (balise HTML, termes) présentes sur chaque page et parallèlement les références croisées vers d'autres sources sont récoltées. Les paires communes à la plupart des pages Web correspondent aux éléments de données de la source, la balise HTML n'étant pas gardée. Ces éléments de données sont la base du schéma de la source.

En effet, ces derniers générant des pages de structure similaire, les mêmes termes pertinents apparaissent généralement dans les mêmes balises HTML d'une page à l'autre. Le fait d'utiliser l'information de la balise associée au terme garantit que le terme récurrent considéré sera conservé uniquement s'il est repéré dans cette même balise dans la plupart des pages Web de la source. Ainsi, un terme comme « different » qui apparaît de manière répétitive dans une source textuelle, telle que OMIM, mais qui n'est pas contenu dans la même balise d'une page à une autre, ne correspond pas à un ED auquel sont associées des valeurs et n'est donc pas extrait par notre méthode.

Un exemple des EDs extraits de la source HGNC est donné figure 4.4 page suivante.

4.2.3.3 Traitement des références croisées

À partir des liens hypertextes (présents sur les différentes pages Web) que nous avons récupérés par l'algorithme d'extraction des EDs (présenté précédemment), il est parfois possible de définir le nom de la source dont une référence croisée a été trouvée. En effet, le texte associé aux hyperliens est soit le numéro de l'entrée correspondante dans la source référencée, soit le nom de la source elle-même (par exemple, OMIM dans la figure 4.4 page ci-contre). Dans ce deuxième cas, on peut définir directement une référence croisée de la source interrogée vers celle qui est référencée (de HGNC vers OMIM, dans l'exemple). Dans le premier cas, un traitement

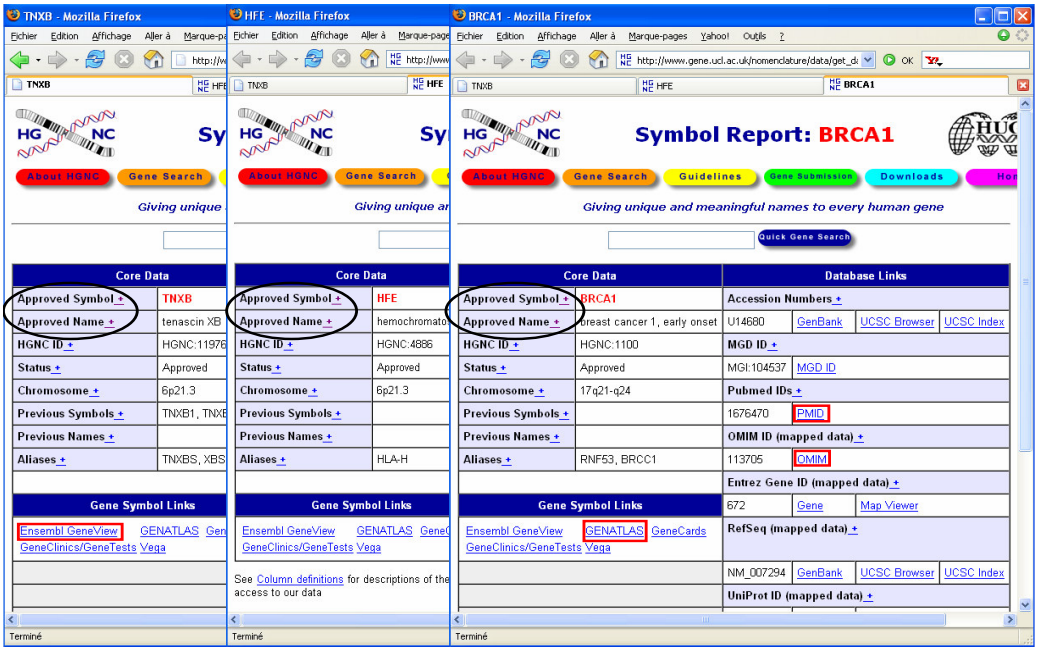


FIG. 4.4 – **Interrogation de la source HGNC pour les trois symboles de gènes « TNXB », « HFE » et « BRCA1 ».** À titre d'exemple, nous avons entouré les termes `Approved Symbol` et `Approved Name` qui apparaissent dans les trois différentes pages Web résultats et sont donc des éléments de données (les termes situés en dessous le sont également puisqu'on les retrouve dans chaque page). Des liens vers ENSEMBL, GENATLAS, PubMed et OMIM sont encadrés pour illustrer certaines des références croisées que nous avons identifiées dans HGNC.

supplémentaire est nécessaire. Il faut d'abord identifier la racine de l'URL pour vérifier si celle-ci correspond à l'URL du site Internet d'une source déjà identifiée et dans ce cas, on parvient à préciser la source référencée. Si ce n'est pas le cas, une intervention humaine est nécessaire pour préciser le nom de la source dont on a récupéré l'URL. Cette dernière sert ensuite de référence lors du traitement d'une nouvelle source.

4.2.3.4 Typage des éléments de données : exploitation des valeurs associées

L'acquisition des EDs permet d'avoir une bonne base pour connaître le type d'informations contenues dans les sources, constituant ainsi une solution intéressante pour compenser en partie l'absence de schémas ou l'exploitation difficile des schémas existants. Le problème est que ces EDs sont parfois ambigus. En effet, il n'est pas rare qu'ils acquièrent une partie de leur sens dans le contexte même de la source dont ils sont issus. Par exemple, l'ED `Name` peut se référer à un nom de gène ou de protéine suivant qu'il existe respectivement dans HGNC ou Swiss-Prot. Parallèlement, certaines sources utilisent des EDs qui sont spécifiés de manière complète, comme `Protein Name`. Afin de mettre en correspondance automatiquement un ED `Name` défini dans le contexte de protéines avec l'ED `Protein Name` tout en évitant qu'il soit mis en correspondance avec un autre ED `Name` défini dans un contexte différent (noms de gènes, par exemple), nous proposons une méthode permettant d'attribuer un type aux EDs en utilisant les données qu'ils contiennent [Mougin 06b], [Mougin 06c]. Ce typage nécessite de récupérer préalablement les valeurs associées aux EDs et de mettre ces valeurs en correspondance dans une terminologie biomédicale déjà introduite précédemment : l'UMLS.

4.2.3.4.1 Extraction des valeurs associées aux éléments de données. La méthode d'extraction des valeurs associées aux EDs se base sur le corpus de gènes (cf 4.2.2 page 102). Pour chaque source, nous avons développé un programme exploitant les CGI. 100 pages Web correspondant aux 100 gènes du corpus sont récupérées et analysées de manière à identifier l'emplacement où se situent les différents EDs de la source considérée et en récupérer les valeurs. Des exemples de valeurs associées à l'ED `Approved Name`, extrait de la source HGNC, sont « Tenascin XB », « Hemochromatosis » et « Breast Cancer 1, early onset » (Figure 2.4 page 35). Il faut noter que, bien que l'interrogation se fasse pour 100 gènes, on ne récupère pas forcément 100 valeurs pour chaque ED puisque certaines d'entre elles peuvent être vides. Par exemple, si on considère cette fois l'ED `Previous Names` de la source HGNC, il n'y a pas de valeur associée à cet ED dans la page Web décrivant le gène « BRCA1 ».

4.2.3.4.2 Typage des éléments de données. On dispose pour chaque source d'un ensemble d'EDs auxquels sont associées entre 0 et 100 valeurs récupérées sur les pages Web fournies par la source. À ce stade, un traitement pour chaque ED est effectué, indépendamment de la source dont il est issu. Pour déterminer quel type d'informations représentent les valeurs d'un ED, nous utilisons l'UMLS, dans laquelle on va chercher des correspondances avec des concepts du Metathesaurus. Ce sont les types sémantiques catégorisant les concepts représentant les valeurs qui vont permettre de préciser quel type de données un ED contient.

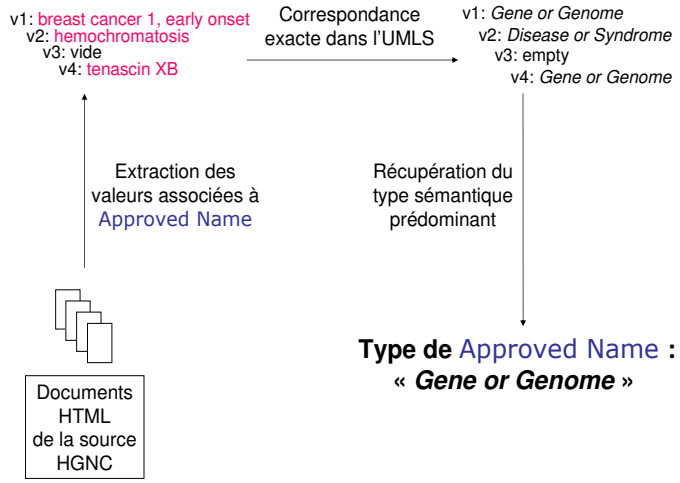


FIG. 4.5 – **Exemple de typage d'un élément de données au travers de ses valeurs.** Pour certaines valeurs non vides récupérées dans la source HGNC pour l'élément de données `Approved Name`, on trouve une correspondance exacte dans l'UMLS pour trois des quatre valeurs présentées ici. Les types sémantiques associés sont DISEASE OR SYNDROME et GENE OR GENOME mais la proportion de ce dernier étant plus importante par rapport à l'autre, le type sémantique choisi pour attribuer un type à `Approved Name` est GENE OR GENOME. On peut en déduire que cet élément de données décrit des noms de **gènes** officiels.

La mise en correspondance des valeurs avec des concepts du Metathesaurus est implémentée au moyen des outils lexicaux fournis par l'UMLS, que nous avons adaptés à nos besoins. Nous effectuons des recherches exactes et normalisées (décrites dans la partie Ressources lexicales de la section 4.1.2.1 page 98), de manière automatique, pour chaque valeur non vide contenue dans l'ED. En pratique, le processus de normalisation consiste à rendre les termes fournis en entrée et les termes cibles potentiellement compatibles en supprimant des différences peu importantes, telles que l'inflexion, la casse, les tirets bas, la présence de virgules ou encore des variations dans l'ordre des mots [McCray 94]. On obtient ainsi un ensemble de concepts associés à chaque ED et les types sémantiques les catégorisant. Ensuite, nous déterminons quel type sémantique permet de catégoriser au moins 50% des concepts correspondant aux valeurs. C'est ce type sémantique qui permet d'attribuer un type à l'ED contenant ces valeurs. Si on considère à nouveau l'exemple de l'ED `Approved Name`, notre méthode détermine que cet ED se réfère à des noms de **gènes** (et non pas des noms de protéines, ou autres) puisque la majorité de ses valeurs sont catégorisées par le type sémantique GENE OR GENOME (Figure 4.5).

Cette méthode n'est efficace que dans les cas où les valeurs des EDs appartiennent à l'UMLS, ce qui est limitatif. En effet, les valeurs sont parfois simplement des nombres ou encore des identifiants qu'il n'est pas possible de trouver en tant que tels dans l'UMLS. Nous proposons donc une solution alternative pour les cas où la méthode précédente ne permet pas de représenter l'ensemble de valeurs associées à un ED. Nous cherchons à assigner des types prédéfinis plus

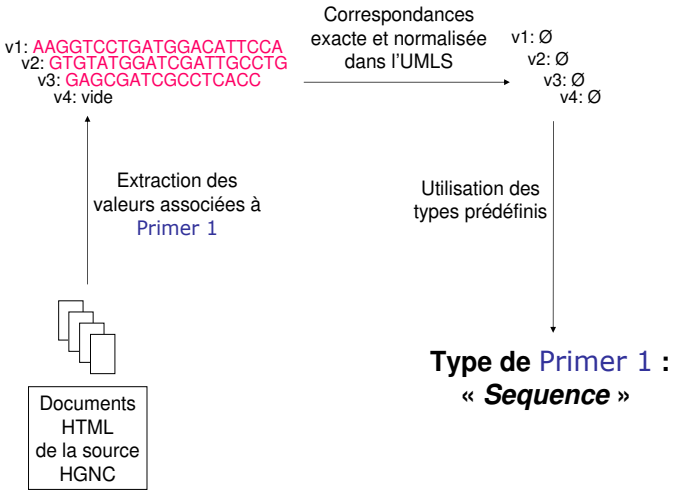


FIG. 4.6 – **Exemple de précision d'un élément de données au travers de ses valeurs.** Les valeurs récupérées dans la source Geneloc (on n'en illustre ici que quatre) pour l'élément de données `Primer 1` n'existent pas dans l'UMLS donc un type prédéfini plus général lui est attribué. Ses valeurs correspondant à des chaînes de caractères contenant uniquement les lettres « A », « T », « G » et « C », on associe à `Primer 1` le type *Sequence*.

« larges » de la façon suivante :

- les EDs dont les valeurs contiennent des termes spécifiques, tels que « ID(s) », « identifier » ou « accession », sont tout d'abord isolés. L'ED correspondant est typé comme un identifiant, c'est-à-dire avec *Identifier* ;
- les valeurs sont ensuite analysées comme des chaînes de caractères. Si chacune des valeurs de l'ED est une série de « A », « T », « G » et « C » alors on assigne à l'ED le type *Sequence* (voir exemple figure 4.6) ;
- enfin, les EDs restants sont typés comme *Integer* ou *String* en fonction de leurs valeurs.

4.2.3.5 Définition des schémas locaux au format XML

Nous définissons le schéma de chaque source de données à intégrer en nous basant sur les éléments importants identifiés précédemment (cf 4.2.3.2.2 page 104). Chaque schéma contient :

- des méta-données correspondant à des informations générales qui sont collectées manuellement. Ces informations incluent le nom de la source associée, l'URL permettant d'interroger cette dernière (programme CGI) et le type de données que fournit la source, c'est-à-dire l'entité sur laquelle elle est focalisée (par exemple, le gène pour HGNC) ;
- les EDs extraits automatiquement de leur source ainsi que le type qui leur a été attribué, ce qui donne de l'information sur son contenu ;
- enfin, les références croisées vers d'autres sources de données sont indiquées.

4.2.4 Étape 2 : Conception du schéma global

Pour concevoir le schéma global, nous avons utilisé l'UMLS comme élément central car c'est un système terminologique qui fournit une large couverture du domaine biomédical. De plus, l'UMLS contient des références croisées vers des sources de données telles que OMIM [Bodenreider 04]. Enfin, il offre de nombreuses fonctionnalités au travers des outils lexicaux introduits précédemment. Le problème est que le Metathesaurus contenant l'essentiel des données conceptuelles de l'UMLS est un graphe présentant des cycles, comme l'ont montré de nombreux travaux [Cimino 98], [Pisanelli 98], [Hahn 04]. Comparativement à celui des vocabulaires sources qu'il intègre, le graphe du Metathesaurus est à la fois plus large et plus profond et contient environ cinq millions de relations hiérarchiques qui sont représentées sous les formes *est-père / fils-de* (*parent / child*) et *est-plus-général / spécifique-que* (*broader / narrower than*). Pour pouvoir exploiter l'UMLS dans notre système, nous supprimons tout d'abord les cycles qu'il contient, de manière à ce qu'il constitue un graphe orienté sans cycle. Nous introduisons certaines des raisons pour lesquelles l'UMLS présente des cycles puis nous proposons deux approches différentes pour éliminer ces derniers. Nous comparons ensuite ces approches pour déterminer celle qui est la meilleure. Enfin, nous définissons notre schéma global, orienté UMLS, dans un langage formel du Web sémantique.

4.2.4.1 Origine des cycles dans l'UMLS

Une notion fondamentale est nécessaire pour comprendre les mécanismes à l'origine des relations hiérarchiques circulaires dans l'UMLS. Bien qu'enregistrées et utilisées au niveau conceptuel, de nombreuses relations hiérarchiques ont été définies au niveau du terme. Autrement dit, le regroupement de termes synonymes sous un même concept modifie la structure originelle des vocabulaires sources. Tandis que le processus produit un système dont la structure est poly-hiérarchique, unifiée et utile, les relations hiérarchiques circulaires peuvent être vues comme son effet secondaire. Dans une étude précédente, différents facteurs à l'origine de cycles dans l'UMLS ont été identifiés [Bodenreider 01]. Nous décrivons brièvement les principales catégories recensées dans cette étude.

4.2.4.1.1 Granularité. Le niveau de granularité du vocabulaire source est parfois différent de celui de l'UMLS. Si le premier est plus fin que le second, deux termes très proches (mais cependant distincts dans le vocabulaire source) sont susceptibles d'être regroupés dans un même concept et générer ainsi, au cours du processus d'intégration, une relation réflexive, qui était à l'origine une relation hiérarchique (Figure 4.7 page suivante).

4.2.4.1.2 Termes composés. Les termes comprenant des conjonctions de type « et » et « ou » posent des problèmes par leur imprécision [Mendonca 98]. En effet, `C1 et C2` peut être interprété comme `C1` *avec* `C2`. Dans ce cas, le concept `C1 et C2` est fils à la fois de `C1` et du concept `C2` puisque c'est un concept plus précis. L'autre interprétation du concept `C1 et C2` est `C1 ou C2`, rendant ainsi les concepts `C1` et `C2` tous deux fils de `C1 et C2` qui est donc, dans ce cas, plus général. Cela entraîne l'apparition de relations hiérarchiques circulaires directes (Figure 4.8 page suivante). Par exemple, le concept `Veines de la tête et du cou` peut désigner des structures

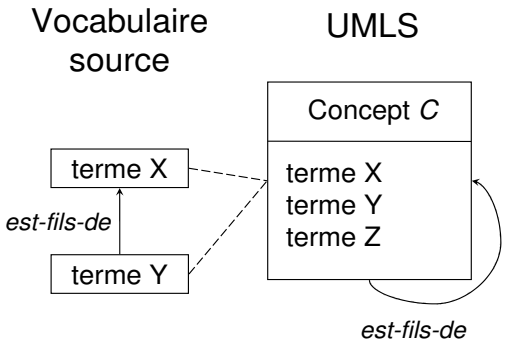


FIG. 4.7 – **Exemple de relation réflexive dans l'UMLS.** Ce cycle est du à une différence de granularité existant entre le vocabulaire source (qui décrit ses concepts de manière plus fine) et l'UMLS.

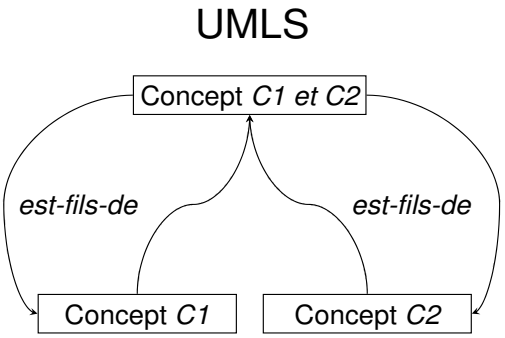


FIG. 4.8 – **Exemple de deux relations hiérarchiques circulaires directes dans l'UMLS.** Ce cycle est du à la présence de termes composés qui peuvent avoir une double interprétation.

anatomiques communes aux deux sites (par exemple, l'artère carotide) ainsi que l'ensemble des structures appartenant soit à la tête, soit au cou.

4.2.4.1.3 Conventions organisationnelles. Certains vocabulaires sources utilisent des relations non hiérarchiques pour organiser leurs termes hiérarchiquement. C'est le cas par exemple de la relation entre `acides` et `sels` : un acide n'est pas une sorte de sel, mais, combiné à une base, l'acide produit un sel et de l'eau. Par convention, certaines terminologies comme MeSH représentent l'acide comme le fils du sel (ce qui est reproduit dans l'UMLS). Cette représentation est utile pour la structuration des termes et la recherche d'informations même si elle n'est pas correcte. En effet, il ne s'agit pas réellement d'une relation hiérarchique et d'autres terminologies peuvent adopter une autre convention pour représenter les acides et les sels qui en dérivent.

4.2.4.1.4 Termes sous-spécifiés. Dans certains vocabulaires sources, on trouve des termes qui, à dessein, sont sous-spécifiés. Cela peut poser le même type de problème que lorsque la

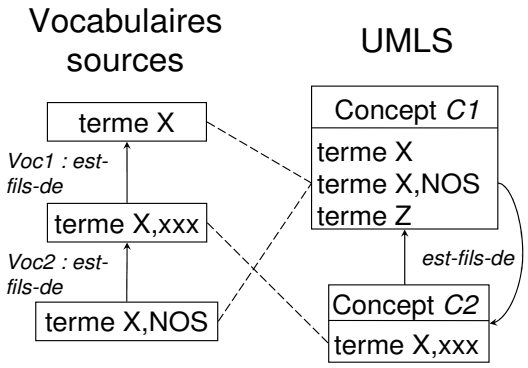


FIG. 4.9 – **Autre cause de la constitution d'une relation hiérarchique circulaire directe dans l'UMLS.** Ce cycle est du à la présence de termes sous-spécifiés dans certains vocabulaires sources et qui ne sont pas différenciés d'autres termes proches mais plus précis.

granularité du vocabulaire source est différent de l'UMLS (cf 4.2.4.1.1 page 111). En fait, le processus d'intégration ne les différencie pas des termes plus généraux. Par exemple, on trouve des termes contenant des expressions telles que « not otherwise specified » ou « NOS ». Les termes de la forme X, NOS sont souvent classés comme *est-fils-de* X dans les sources, ce qui semble erroné. Par contre, ils sont généralement associés au même concept que le terme X dans l'UMLS, ce qui paraît plus correct mais a pour effet de provoquer une relation réflexive. Des cas de figure plus complexes où X et X, NOS sont reliés dans différents vocabulaires par l'intermédiaire d'un terme X,xxx et qui sont regroupés lors du processus d'intégration, créent cette fois une relation hiérarchique circulaire directe (Figure 4.9).

D'autres facteurs plus complexes, non présentés ici, peuvent être à l'origine de relations circulaires hiérarchiques indirectes [Bodenreider 01]. Deux approches peuvent être implémentées pour supprimer les cycles dans l'UMLS. L'approche **naïve** consiste à éviter les boucles lors du parcours du graphe terminologique. L'approche **formelle** s'attache à définir un certain nombre de règles permettant d'éliminer les cycles a priori et de transformer le graphe terminologique en un graphe orienté sans cycle. Nous présentons ici ces deux approches ainsi que leurs avantages et inconvénients respectifs [Mougin 05], [Mougin 06a].

4.2.4.2 Approche naïve pour éliminer les cycles de l'UMLS

L'approche naïve s'implémente de manière ponctuelle en fonction de l'application nécessitant un parcours de graphe. Elle consiste simplement à marquer les nœuds visités pendant le parcours du graphe, de manière à éviter de visiter ces mêmes nœuds une deuxième fois. Cette approche est efficace pour empêcher les boucles, mais *naïve* dans le sens où c'est uniquement l'ordre dans lequel les nœuds sont visités qui détermine quelle relation sera ignorée dans le cas d'un cycle.

Supposons que l'on souhaite obtenir les descendants d'un concept donné dans le Metathesaurus de l'UMLS. La méthode que nous adoptons dans ce cas consiste à parcourir le graphe

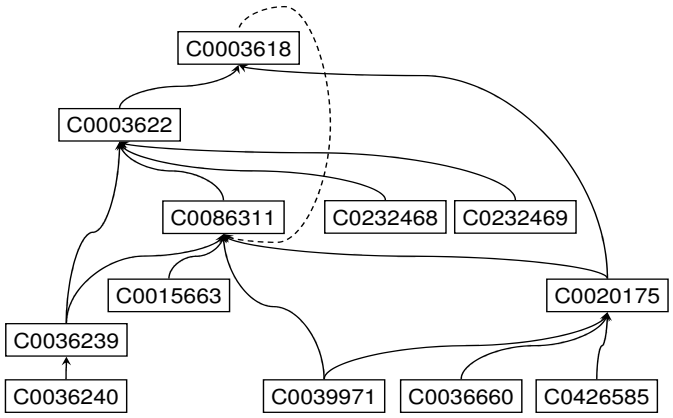


FIG. 4.10 – **Les descendants de *Desire for food (C0003618)* calculés avec l'approche naïve.** La relation ***C0086311*** *est-père-de* ***C0003618*** est ignorée (en pointillés sur la figure).

en profondeur d'abord, en traitant les fils d'un nœud donné dans l'ordre alphabétique des étiquettes (c'est-à-dire des CUIs) pour des raisons de reproductibilité. Ce choix est arbitraire. Les concepts ***Desire for food (C0003618)***, ***Appetite Regulation (C0003622)*** et ***Food Intake Regulation (C0086311)*** sont choisis pour illustrer le problème posé par l'utilisation de cette approche. La figure 4.10 montre qu'avec l'approche naïve et en partant du concept ***C0003618*** pour obtenir ses descendants, la relation ***C0086311*** *est-père-de* ***C0003618*** est ignorée car elle causerait un cycle dans le graphe (***C0003618*** → ***C0003622*** → ***C0086311*** → ***C0003618***). Au contraire, la même relation est utilisée quand le graphe est parcouru en commençant par ***C0086311*** (Figure 4.11 page ci-contre), tandis que la relation ***C0003622*** *est-plus-général-que* ***C0086311*** utilisée dans le graphe précédent est maintenant ignorée en raison de son implication dans le cycle (***C0086311*** → ***C0003618*** → ***C0003622*** → ***C0086311***).

L'avantage de l'approche naïve est qu'elle est relativement **simple à implémenter** et permet une **suppression automatique des cycles**, sans recourir à un expert du domaine. Cependant, elle présente des limites puisqu'elle **ne garantit pas que les liens ignorés dans le parcours de graphe soient effectivement ceux qui sont sémantiquement incorrects**.

4.2.4.3 Approche formelle pour éliminer les cycles de l'UMLS

L'approche formelle, plus théorique, consiste en un ensemble d'heuristiques et de règles définies de manière à identifier et éliminer *a priori* tous les cycles du graphe global. En fonction du type de relations responsables du cycle, le traitement diffère.

Le traitement des **relations réflexives** est trivial, il suffit de supprimer les relations du type `C est-père / fils-de C` ainsi que `C est-plus-général / spécifique-que C`.

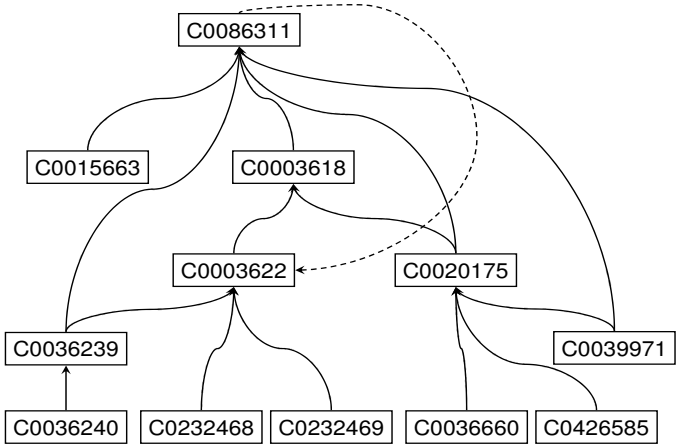


FIG. 4.11 – **Les descendants de *Food Intake Regulation (C0086311)* calculés avec l'approche naïve.** Cette fois, c'est la relation ***C0003622*** *est-plus-général-que* ***C0086311*** qui est ignorée.

Différents types de traitement peuvent être proposés pour résoudre les **relations circulaires directes**. Par exemple, la redondance des relations présentes entre deux concepts dans différents vocabulaires sources peut être utilisée. En d'autres termes, si la relation `C1` *est-père-de* `C2` existe dans trois vocabulaires différents, tandis que la relation `C2` *est-père-de* `C1` n'apparaît que dans un seul, on préfèrera garder la relation `C1` *est-père-de* `C2` et éliminer l'autre. Des critères de confiance associés à tel vocabulaire source plutôt qu'à un autre peuvent également être pris en compte dans les règles pour déterminer si une relation doit être supprimée, ou au contraire, préservée.

Le détail des règles et heuristiques de la méthode formelle définie pour éliminer les cycles dans le graphe du Metathesaurus de l'UMLS est donné dans [Bodenreider 01]. Pour l'exemple présenté dans la partie précédente illustrant les limites de l'approche naïve, nous avons montré que le choix de la relation à supprimer était aléatoire alors que la méthode formelle identifie, elle, de manière cohérente la relation ***C0086311*** *est-père-de* ***C0003618*** comme étant incorrecte en raison du vocabulaire source dont elle provient, auquel un critère de confiance faible a été associé.

Les règles et heuristiques permettent de traiter certaines **relations circulaires indirectes** mais certains cas de figure nécessitent l'intervention d'un expert du domaine pour déterminer quelle relation doit être préservée par rapport à une autre.

L'approche formelle permet de construire un graphe orienté sans cycle à partir du graphe global, ce qui facilite les parcours d'arbres. Dans le cas de systèmes comprenant un grand nombre de cycles comme l'UMLS, **l'intervention d'un expert du domaine** constitue une limite. Par contre, en opposition à l'approche naïve, l'approche formelle **sélectionne de manière systématique et cohérente les relations qui doivent être supprimées**.

C'est pour ces différentes raisons que nous avons voulu comparer les résultats obtenus par ces deux approches et évaluer le bénéfice d'éliminer les cycles d'une manière complexe et coûteuse par rapport à une approche plus simple mais moins rigoureuse.

4.2.4.4 Méthode de comparaison des approches naïve et formelle

Nous comparons les approches naïve et formelle sur le parcours de graphe du Metathesaurus de l'UMLS en utilisant une application des graphes terminologiques : le calcul de l'ensemble des descendants d'un concept[13]. Pour cela, nous considérons chaque concept du Metathesaurus et calculons ses descendants avec chaque méthode. Notre hypothèse est que l'approche formelle réduit le nombre de descendants obtenus et améliore la cohérence sémantique des ensembles de descendants.

4.2.4.4.1 Calcul de descendants. L'ensemble des descendants d'un concept consiste est calculée en effectuant la fermeture transitive de ses relations hiérarchiques. Elle est réalisée par un parcours du graphe. Comme décrit précédemment, l'**approche naïve** consiste à marquer les nœuds visités pendant le parcours de graphe. Pour éviter de construire de trop larges graphes dus à des relations erronées, nous limitons la profondeur maximum à 50 niveaux, sachant qu'une telle profondeur n'est jamais atteinte dans le graphe sans cycle du Metathesaurus. L'**approche formelle** transforme le Metathesaurus en un graphe orienté sans cycles préalablement au calcul des ensembles de descendants.

4.2.4.4.2 Méthodes d'évaluation. L'évaluation porte sur la cohérence sémantique des ensembles de descendants obtenus par les deux approches. Nous définissons d'abord certaines notions requises pour l'évaluation.

Définition 3 *Deux types sémantiques sont* compatibles *s'ils appartiennent au même groupe sémantique.*

Par exemple, les types sémantiques Disease or Syndrome et Finding, bien que non liés hiérarchiquement, sont compatibles puisqu'ils appartiennent tous les deux au groupe sémantique Disorders.

Définition 4 *Un descendant est* sémantiquement cohérent *avec son concept source si et seulement si le type sémantique catégorisant le descendant est le même que le type sémantique du concept source ou un descendant de ce type sémantique.*

On notera qu'en cas de catégorisation multiple, c'est-à-dire lorsque les concepts sont catégorisés par plusieurs types sémantiques, la cohérence est requise pour au moins une des paires de types sémantiques. Cette définition est similaire en ce qui concerne la cohérence sémantique par rapport aux groupes sémantiques.

[13]Les ensembles de descendants sont utilisés par exemple pour calculer tous les médicaments d'une classe thérapeutique donnée (par exemple tous les antibiotiques, représentés comme les descendants du concept ***Antibiotique***)

Définition 5 *La* cohérence sémantique d'un ensemble de concepts *est mesurée par le rapport entre le nombre de concepts sémantiquement cohérents avec le concept source et le nombre total de concepts dans cet ensemble.*

Par exemple, le concept ***Adrenal cortex diseases (C0001614)*** est catégorisé comme Disease or Syndrome, un type sémantique du groupe DISORDERS. Tous ses descendants appartiennent également au groupe sémantique DISORDERS. La cohérence sémantique des descendants du concept source est donc parfaite (valeur à 100% d'après la définition 5) du point de vue groupe sémantique. 263 des 317 descendants sont catégorisés comme Disease or Syndrome ou un de ses fils, Neoplastic Process. Les 54 descendants restants sont incohérents sémantiquement avec leur concept source car leurs types sémantiques sont incompatibles avec Disease or Syndrome (définition 3). En effet, Anatomical Abnormality, Congenital Abnormality, Finding, Injury or Poisoning, Pathologic Function et Sign or Symptom permettent de catégoriser des descendants de ***Adrenal cortex diseases*** alors qu'ils ne sont pas des descendants de Disease or Syndrome (définition 4). La cohérence sémantique de l'ensemble de descendants du concept ***Adrenal cortex diseases*** est donc de 83% du point de vue type sémantique (définition 5).

4.2.4.4.3 Comparer les ensembles de descendants.

Les ensembles de descendants obtenus pour un concept donné du Metathesaurus par les approches naïve et formelle sont comparés comme suit. D'abord, une simple intersection des deux ensembles est réalisée pour identifier les concepts communs aux deux ensembles et ceux qui sont spécifiques à chacun. Nous recherchons ensuite la cohérence sémantique des deux ensembles en étudiant la distribution des types (et groupes) sémantiques dans ces ensembles. De plus, nous vérifions la compatibilité de chaque type (et groupe) sémantique représenté dans les descendants, par rapport à ceux du concept source. Le nombre de types (et groupes) sémantiques représentés dans les ensembles de descendants constitue l'aspect *quantitatif* de la cohérence sémantique, tandis que la compatibilité des types (et groupes) sémantiques représentés dans les descendants, par rapport à ceux du concept source, définit la cohérence sémantique de manière *qualitative*.

Les résultats de la comparaison sont détaillés dans la section Résultats (cf 5.2 page 133). Nous montrerons que l'approche formelle est la plus cohérente. Cette approche a donc été utilisée pour traduire le Metathesaurus de l'UMLS en graphe orienté sans cycle.

4.2.4.5 Définition du schéma global au format OWL

Nous avons choisi de décrire le schéma global en OWL, langage introduit précédemment dans le cadre du Web sémantique (cf 2.1.2.1 page 19), qui permet de décrire des ontologies formelles. Même si l'UMLS n'en est pas une, il nous a semblé intéressant d'utiliser ce langage qui pourrait permettre (dans un second temps) à notre système d'offrir des fonctionnalités plus avancées (voir la discussion 7.4 page 178). Ainsi, une fois l'UMLS traduit en graphe orienté sans cycle, nous définissons le schéma global à l'aide du langage OWL. Nous avons représenté les différents éléments constituant l'UMLS de la manière suivante :

- les types sémantiques sont représentés par une classe au sens OWL (<owl :Class>) dont la classe racine est *Root_Semantic_Type* ;
- les concepts UMLS sont également représentés par une classe et leur classe racine est *Root_Concept* ;
- les concepts UMLS et les types sémantiques sont tous reliés par des relations de type *is_a*, c'est-à-dire à l'aide de la balise <rdfs :subClassOf>. On notera que la notion de catégorisation des concepts par des types sémantiques est confondue en relation de type *is_a* de manière délibérée. Gu et al. ont en effet montré qu'il est possible de représenter les concepts de l'UMLS comme des instances des types sémantiques [Gu 00]. La distinction entre classes et instances n'étant pas toujours clairement définie, nous avons simplifié leur représentation en définissant les concepts comme des enfants des types sémantiques ;
- chaque type sémantique a un identifiant unique (son TUI - Type Unique Identifier, défini dans l'UMLS), un libellé correspondant à son intitulé et une propriété *isDefinedBy* contenant sa définition ;
- chaque concept a un identifiant unique (son CUI), un libellé correspondant au terme préféré du concept UMLS, une propriété *isDefinedBy* contenant sa (ses) définition(s) et une propriété *has_synonyms* dont la valeur est la liste de ses synonymes (dans une chaîne de caractères unique). Cette dernière propriété est définie comme *DataTypeProperty* en OWL avec pour domaine n'importe quel concept UMLS, représentée par la classe *Root_Concept*, et comme co-domaine un élément de type *String*.

Nous verrons par la suite comment nous avons enrichi l'UMLS au moyen d'informations recueillies dans WN (cf 6.1.1.1 page 148).

4.2.5 Étape 3 : Mise en correspondance des schémas locaux avec le schéma global

Nous proposons trois approches pour mettre en correspondance les schémas locaux avec le schéma global. Les deux premières approches se situent au niveau *schéma* et mettent en œuvre des techniques terminologiques et structurelles. Tout d'abord, nous cherchons des correspondances directement dans l'UMLS puis nous réalisons les mises en correspondance en exploitant une ressource externe : WordNet. Enfin, nous avons implémenté une approche située au niveau *instances* qui compare les valeurs des éléments de données.

4.2.5.1 Mise en correspondance directe des éléments de données dans l'UMLS

Chaque source a donc sa propre façon de nommer les EDs qu'elle utilise, causant une disparité sémantique entre les EDs de sources différentes. Par exemple, un ED décrivant une pathologie sera nommé `Disease` dans une source et `Disorder` dans une autre. Pour pallier ce problème, nous avons développé une approche terminologique permettant de mettre en correspondance les EDs avec des concepts du Metathesaurus de l'UMLS. Une recherche exacte est d'abord réalisée puis si aucun concept n'est trouvé, l'ED est normalisé pour identifier une correspondance proche. Pour cela, les outils lexicaux de l'UMLS sont utilisés (cf 4.1.2.1 page 98). Ensuite, une correspondance approximative est recherchée au moyen du programme MetaMap pour tous les

EDs auxquels aucun concept n'a été associé. Cette procédure de mise en correspondance est entièrement automatique et s'arrête lorsqu'un ED est associé à un concept UMLS.

Cette étape résulte en différentes catégories de correspondances :

- **correspondance unique**. C'est une correspondance de cardinalité 1-1, c'est-à-dire qu'à un ED est associé un unique concept UMLS. Par exemple, l'ED `mRNA sequence` est associé, de manière exclusive, au concept ***RNA, Messenger (C0035696)*** ;
- **correspondance multiple**. Dans ce cas, des correspondances de cardinalité 1-n sont trouvées, c'est-à-dire qu'à un ED est associé plusieurs concepts UMLS. Par exemple, l'ED `Protein` résulte en trois concepts : ***Protein (C0033684)***, ***Protein measurement (C0202202)*** et ***Protein location (C1325816)*** ;
- **aucune correspondance**. Certains EDs sont absents de l'UMLS car ils ne sont pas spécifiques du domaine biomédical et nécessitent d'être représentés à un niveau plus général (cardinalité 1-0). Des exemples de tels EDs incluent `Topology`, `Features`, `Keywords` et `Domains`.

Les résultats sont satisfaisants pour les correspondances uniques mais incomplets dans les autres cas. Les correspondances multiples nécessitent d'être désambiguïsées et pour les EDs non trouvés dans l'UMLS, il est nécessaire d'utiliser une ressource externe permettant de les mettre en correspondance de manière indirecte dans l'UMLS.

4.2.5.2 Mise en correspondance via une ressource externe : WordNet

WN fournit une couverture plus large que le domaine biomédical, il peut donc permettre de compléter les correspondances trouvées directement par l'UMLS. Nous avons donc choisi d'utiliser WN comme ressource externe pour garantir une couverture plus large des EDs représentés dans les sources à intégrer [Mougin 06d]. Parallèlement, WN pose un problème, qui est la contrepartie de sa large couverture. Nombreuses de ses entrées lexicales sont polysémiques et donc il existe plusieurs synsets associés à un même terme. Pour pallier ce problème, nous proposons une méthode simple de désambiguïsation de WN avant de rechercher les synsets associés aux EDs. Puis nous mettons en correspondance les synsets obtenus avec des concepts de l'UMLS. Finalement, nous montrons qu'en mettant en correspondance l'ensemble des EDs avec des synsets de WN, nous allons non seulement augmenter la couverture des EDs dans l'UMLS, mais aussi améliorer les correspondances trouvées dans l'UMLS.

4.2.5.2.1 Désambiguïsation de WordNet. Une approche proposée par Fellbaum et al. utilise WN comme base pour créer un nouveau type de ressource pour le domaine de la santé [Fellbaum 06] : MedicalWordNet. Elle ne consiste pas en une extension de WN mais cherche à exploiter les connaissances du domaine présentes dans WN pour construire un entrepôt contenant des mots de WN, des phrases contenant des faits médicaux validés par des experts ainsi que des expressions utilisées dans le domaine de la santé par des personnes non initiées à celui-ci. Les raisons pour lesquelles nous proposons une méthode alternative à cette approche sont les suivantes : la constitution de MedicalWordNet nécessite une intervention humaine, ce que nous souhaitons éviter, elle contient des données qui ne nous sont pas utiles (les phrases principalement) et enfin, c'est d'ailleurs le point le plus important, nous voulons exploiter WN pour

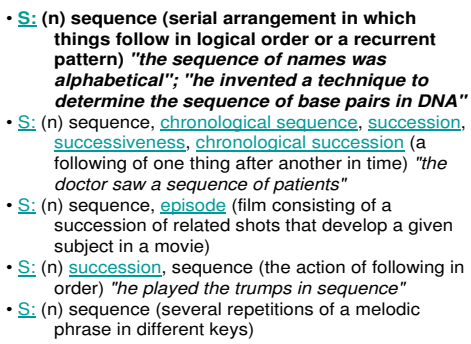


FIG. 4.12 – **Synsets de WordNet candidats pour le terme « Sequence ».** Cinq sens différents existent pour ce terme et seul le premier nous intéresse car il concerne le domaine biomédical. C'est la présence du terme « DNA » dans sa définition qui permet de savoir que c'est le synset que nous souhaitons garder.

le langage général qu'il apporte de manière à ce qu'il compense les limites des ressources du domaine biomédical alors que MedicalWordNet adapte WN à ses besoins avec l'objectif inverse.

La désambiguïsation que nous cherchons à faire consiste à filtrer les synsets associés à des mots spécifiques du domaine biomédical lorsque plusieurs sens pour un même mot existent. Par exemple, le terme « Sequence » a un sens particulier dans le domaine biomédical et d'autres sens dans des contextes différents (Figure 4.12) et dans ce cas, nous voulons éliminer les synsets correspondant à des domaines différents de ceux qui nous intéressent. Notre méthode consiste à exploiter les propriétés associées aux synsets de WN, à savoir les domaines, les définitions et les hypernymes.

Tout d'abord, nous avons défini manuellement la liste des domaines qui nous intéressent. Cela permet de garder uniquement les synsets dont la définition commence par le domaine que couvre ce synset, quand cette information existe. Les noms de domaines pour lesquels nous souhaitons garder les synsets sont : *Genetics*, *Biology*, *Medicine*, *Pharmacy*, *Psychiatry*, *Radiology*, *Surgery*, *Biochemistry*, *Anatomy* et *Physiology*. Par exemple, la désambiguïsation du terme « Species » de la figure 4.2 page 100 est résolue parce que son domaine, *Biology*, fait partie de notre ensemble, ce qui indique qu'il est du domaine biomédical. Seul le synset numéro 1 est donc gardé, les autres sont supprimés. Mais il existe de nombreux termes auxquels sont associés des synsets dont aucune définition ne présente d'informations concernant le domaine dans lequel ce synset s'exprime. Dans ces cas là, la désambiguïsation n'est pas possible en utilisant cette liste de domaines. Ce cas de figure est illustré par le terme « Sequence » de l'exemple ci-dessus ; aucun synset ne contenant d'informations au sujet du domaine qu'il couvre, il n'est pas possible de déterminer quels synsets supprimer ou garder.

La deuxième étape consiste à rechercher dans la définition et les hypernymes des synsets des mots ou racines de mots du domaine biomédical dont nous avons créé une liste manuellement en considérant des cas dont nous disposions. Ces racines sont *genetic, chemical, anatomic, medic, biolog, dna, rna, chromosom, nucleotide, protein, genom, enzym, tumor* et *molecul*. Cette liste

n'est, bien sûr, pas exhaustive mais elle sera complétée au fur et à mesure que cela sera nécessaire. Cette méthode permet de désambiguïser le terme « Sequence » dont le premier synset associé contient le terme « DNA » et qui correspond donc au seul synset qui sera gardé après cette seconde étape.

Enfin, la dernière étape cherche des domaines reliés au domaine biomédical (mais qui sont plus éloignés) dans les définitions des synsets. Ces domaines secondaires sont *Physics*, *Chemistry*, *Physics and chemistry*, *Dentistry*, *Ecology*, *Botany*, *Zoology*, *Entomology*. De plus, les synsets dont les hypernymes n'incluent pas les mots suivants sont filtrés : *measure, organism, psychologic, physical, social, substance, neural*.

4.2.5.2.2 Méthode de mise en correspondance des éléments de données dans WordNet. Pour mettre en correspondance les EDs avec des synsets de WN, nous avons utilisé le programme *wn* (cf 4.1.2.2 page 99). Quand un ED consiste en plus d'un mot, la correspondance couvrant le plus long syntagme est sélectionné. Par exemple, l'ED `Mus Musculus` est mis en correspondance avec le synset ***mus_musculus#n#1*** plutôt que les deux synsets ***mus#n#2*** (du genre Muridae) et ***musculus#n#1*** (muscle). Quand des correspondances multiples dans WN sont identifiées, la méthode présentée dans la section précédente est appliquée. Ensuite, s'il reste encore plusieurs synsets candidats associés à un ED, on analyse les synonymes de chacun et si l'un d'entre eux contient un (ou plusieurs) autre ED parmi ses synonymes, c'est ce synset là qui est choisi. Par exemple, l'ED `Data` est trouvé dans deux synsets : ***data#n#1*** et ***data#n#2***. Leurs synonymes respectifs sont *information* pour le premier synset et *datum* ainsi que *data point* pour le second. Or, parmi les EDs extraits des sources à intégrer se trouve le mot *information* alors que les synonymes de l'autre synset n'apparaissent pas dans notre ensemble d'EDs. C'est donc le synset ***data#n#1*** qui est gardé pour être mis en correspondance avec l'ED `Data`. Enfin, les synsets candidats sont filtrés suivant leur catégorie syntaxique grâce au logiciel TreeTagger[14], développé à l'université de Stuttgart. Ce logiciel analyse des phrases fournies en entrée et retourne, pour chaque mot constituant les phrases, la catégorie syntaxique auquel il appartient. Considérons par exemple l'ED `Detailed gene map`, le mot « detailed » a trois synsets candidats ; un adjectif et deux verbes. TreeTagger détermine que dans cet ED, ce mot correspond à un adjectif, ce qui permet de choisir uniquement le synset adjectival. Cette méthode de mise en correspondance des EDs dans WN est entièrement automatique et résulte en un (ou des) synset(s) pour chaque ED, avec leurs définition, synonymes et hypernymes.

4.2.5.2.3 Mise en correspondance des concepts UMLS avec les synsets WordNet. Pour mettre en correspondance les synsets (préalablement associés à des EDs) dans l'UMLS, nous avons réalisé une recherche exacte d'abord puis si aucun concept n'est trouvé, le synset est normalisé pour identifier une correspondance qui ne soit pas tout à fait parfaite (méthode basée sur les outils lexicaux de l'UMLS - cf 4.1.2.1 page 98). Comme lors de la mise en correspondance directe des EDs dans l'UMLS (cf 4.2.5.1 page 119), les correspondances obtenues sont de différentes cardinalités.

[14]http ://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/DecisionTreeTagger.html

Burgun et al. ont souligné les caractéristiques communes existant entre l'UMLS et WN et ont proposé des stratégies possibles basées sur les synonymes et les descendants pour mettre leurs éléments en correspondance [Burgun 01a]. Nous proposons ici des méthodes terminologiques et structurelles comparant les propriétés des concepts et des synsets en fonction des **critères** suivants :

1. similitude des définitions ;
2. présence de synonymes communs ;
3. présence d'ancêtres communs.

Pour le critère 1, les définitions sont d'abord découpées en mots. Les mots présents le plus fréquemment dans l'ensemble des mots constituant l'ensemble des définitions des concepts et synsets sont éliminés. Ils correspondent à des termes du langage courant et n'apportent pas d'information pertinente. Des exemples sont *a, of, or, and, that, something, with* et *be*. Ensuite, chaque mot constituant les définitions est mis sous sa forme normalisée au moyen du logiciel TreeTagger. Par exemple, le mot *modified* est mis sous la forme infinitive du verbe, il devient donc *modify*. Enfin, les définitions sont comparées mot à mot et leur similitude est déterminée par le coefficient de Dice [Rasmussen 92] :

$$Sim_{Dice} = \frac{NbMotsCommuns*2}{NbMotsTot}$$

où *NbMotsCommuns* correspond au nombre de mots communs et *NbMotsTot* est le nombre total de mots présents dans les deux définitions (après suppression des mots non pertinents). Cette méthode est également utilisée dans [Knight 94], où les définitions d'un dictionnaire sont comparées avec celles de WN pour des termes lexicalement identiques (les synonymes sont également utilisés, comme nous le faisons avec le second critère). Si le coefficient de Dice est différent de 0, cela signifie qu'au moins un mot est commun aux deux définitions. Comme nous avons préalablement éliminé les mots non pertinents, nous considérons que dès que ce coefficient est différent de 0, le critère 1 est vérifié.

Pour les critères 2 et 3, des concepts UMLS sont associés aux synonymes et hypernymes de WN grâce aux méthodes de recherche exacte et normalisée proposées dans les outils lexicaux de l'UMLS (cf 4.1.2.1 page 98). La présence de synonymes ou d'ancêtres communs (entre le concept et le synset) permet de vérifier respectivement les critères 2 et 3.

4.2.5.3 Comparaison des approches directe et indirecte

Caractéristiques de l'approche directe.

L'approche directe est plus pertinente dans le cas des correspondances uniques pour lesquelles elle permet de trouver l'ED tel quel dans l'UMLS alors que l'approche via WN identifie plusieurs synsets (et par conséquent plusieurs concepts). Par exemple, l'ED `Northern Blot*` qui existe tel quel dans l'UMLS (concept ***C1148548***) est mis en correspondance partiellement dans WN au travers des deux mots `northern` (quatre synsets adjectivaux) et `blot` (deux synsets nominaux).

Caractéristiques de l'approche indirecte.

- **Pour les EDs qui n'ont pas pu être trouvés directement dans l'UMLS**, l'approche via WN propose deux solutions : 1) si un des synonymes des synsets associés à ce type d'ED peut être mis en correspondance avec un concept UMLS alors celui-ci est candidat ; 2) si un des hypernymes directs des synsets associés à ce type d'ED peut être mis en correspondance avec un concept UMLS alors celui-ci est candidat ;
- **pour les correspondances multiples** obtenues par l'approche directe, l'approche via WN permet parfois de les désambiguïser. Pour cela, on consulte la correspondance obtenue via WN, qui peut être unique ou multiple. Si l'on parvient à déterminer quelle paire (concept, synset) est la meilleure (suivant les trois critères présentés précédemment), alors le concept faisant partie de cette paire est sélectionné. La correspondance devient ainsi unique ;
- **pour les correspondances uniques** obtenues par l'approche directe, l'approche via WN permet parfois de les valider. Il faut vérifier que le concept identifié par l'approche est le même que celui identifié par l'approche directe. Si c'est le cas, la correspondance unique directe est validée par l'approche indirecte. Cela peut notamment être utile pour les EDs comportant des acronymes, que les outils lexicaux de l'UMLS interprète parfois de manière erronée.

Conclusions. Ces processus sont entièrement automatisés. L'approche via WN permet parfois de valider, désambiguïser ou identifier des correspondances indirectes dans l'UMLS de manière automatique. L'intervention humaine est nécessaire dans les cas où aucun critère n'a permis de conclure. Le processus général de mise en correspondance des EDs dans l'UMLS directe et via WN est illustré figure 4.13 page suivante.

4.2.5.4 Mise en correspondance des éléments de données au niveau *instances*

Cette dernière approche vise à compléter les correspondances identifiées au niveau *schéma*. Certains EDs ont des noms qui ne sont pas réellement informatifs quant à leur contenu et nécessitent d'être précisés. Par exemple, dans la figure 4.4 page 107, l'ED `Chromosome` laisse penser, d'après son nom, que les valeurs qui lui sont associées correspondent au numéro du chromosome sur lequel se situe le gène dont la page Web est affichée. Or les valeurs associées (« 6p21.3 » et « 17q21-q24 ») sont plus précises, elles fournissent plus exactement la localisation du gène sur le chromosome. Un nom tel que *Chromosomal Location* aurait été mieux adapté pour qualifier cet ED.

Pour pallier ce problème, nous proposons d'exploiter les valeurs associées aux EDs. Des EDs issus de différentes sources, et ayant les mêmes valeurs, peuvent être mis en correspondance et permettre parfois de **trouver de nouvelles correspondances** dans le schéma global. Par exemple, l'ED `Chromosome`, qui est issu de HGNC, peut être mis en correspondance avec l'ED `Chromosomal Location`, extrait de la source HGMD, car la plupart de leurs valeurs sont les mêmes. Ce deuxième ED est mis en correspondance dans l'UMLS avec le concept ***Location (C0450429)***. Il est alors possible de préciser l'ED `Chromosome` en lui associant une correspondance supplémentaire dans l'UMLS (avec le concept ***Location***).

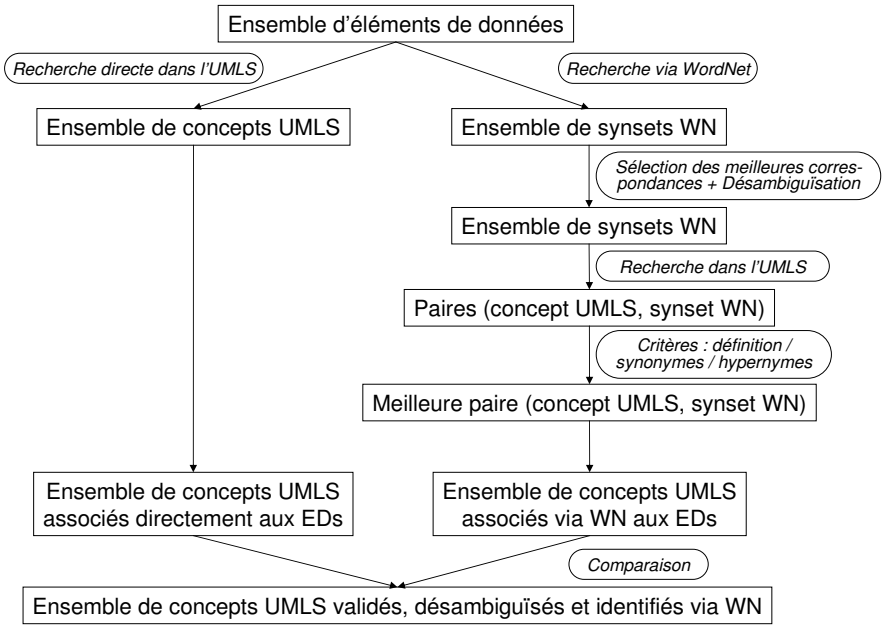


FIG. 4.13 – **Mises en correspondance directes et indirectes via WordNet des éléments de données dans l'UMLS.**

Cette approche permet aussi de **valider des correspondances identifiées au niveau *schéma***. Par exemple, des EDs nommés `Official Symbol` et `Approved Symbols` seront tous deux mis en correspondance avec le concept UMLS ***Symbols (C0679214)*** grâce aux méthodes terminologiques. Si en plus, leurs valeurs sont identiques alors on pourra valider la correspondance initialement identifiée au niveau *schéma*.

Cette méthode ré-utilise les valeurs de chaque ED que nous avons extraites précédemment pour typer les EDs (cf 4.2.3.4.1 page 108). Une mesure de similarité permettant de comparer l'ensemble de valeurs pour chaque paire d'EDs issus de sources différentes est calculée. Nous avons choisi pour cela d'utiliser l'indice de Jaccard qui détermine la similarité entre deux ensembles de valeurs de cardinalité respective c1 et c2 [Van Rijsbergen 79]. Elle est définie par :

$$Sim_{Jaccard} = \frac{c1c2}{c1+c2-c1c2}$$

où c1c2 correspond à la cardinalité de l'ensemble de valeurs communes aux deux ensembles. La valeur de similarité varie de 0 (aucune similarité) à 1 (similarité complète).

Dans ce chapitre, nous avons tout d'abord présenté les sources que nous intégrons virtuellement à notre système d'intégration et les ressources terminologiques utilisées pour sa conception. Puis nous avons décrit les méthodes mises en œuvre pour acquérir automatiquement le schéma de ces sources. Nous avons ensuite proposé deux approches afin d'éliminer les cycles présents

dans l'UMLS pour pouvoir l'utiliser comme schéma global. Finalement, nous avons détaillé les méthodes que nous avons développées pour mettre en correspondance les éléments de données issus des sources avec les concepts UMLS.

Chapitre 5

Résultats

Dans cette partie, nous reprenons les différentes étapes détaillées dans la section Méthodes (cf 4.2 page 101) et donnons les résultats que nous avons obtenus pour chacune.

5.1 Étape 1 : Acquisition des schémas locaux

5.1.1 Extraction des éléments de données

Au total, 474 EDs distincts (548 tokens) ont été extraits des onze sources de données. Parmi eux, 47 (9,9%) apparaissent dans plus d'une source (la casse est ignorée) et les plus fréquents sont `Name` et `Symbol` qui sont présents dans six sources différentes. Notons que ces EDs sont tous les deux ambigus et que sans savoir dans quel contexte ils s'expriment, on ne peut pas les mettre en correspondance directement avec un unique concept commun.

Nous avons dégagé les catégories suivantes parmi les EDs obtenus à partir des pages Web résultats :

- les **EDs attendus** qui correspondent aux EDs recensés par la source comme étant les noms externes (au sens de la source HGNC - cf 2.2.1.1 page 34) correspondant aux attributs constituant son schéma et retrouvés par notre méthode ;
- les **EDs références croisées** qui sont en fait des hyperliens identifiés lors du parcours des pages Web de la source ;
- les **EDs supplémentaires** qui sont des EDs que nous avons extraits alors qu'ils ne sont pas répertoriés par la source comme faisant partie de son schéma mais que nous avons jugés comme corrects et complémentaires par rapport aux EDs attendus. Ils peuvent donc s'avérer utiles pour la recherche d'informations dans la source ;
- les **EDs erronés** qui correspondent à des EDs extraits par notre méthode alors qu'ils ne sont pas répertoriés par la source comme faisant partie de son schéma et que nous avons jugés comme incorrects, dans le sens où ils n'apportent pas d'informations intéressantes.

Une autre catégorie d'EDs existe : les **EDs manquants**. Ils correspondent aux EDs recensés par la source comme étant les noms externes correspondant aux attributs constituant son schéma mais que notre méthode n'a pas retrouvés.

Pour illustrer les résultats fournis par notre méthode, nous reprenons l'exemple de la source HGNC. Les 25 EDs constituant son schéma ainsi qu'une page Web résultat sont présentés dans la figure 2.4 page 35. Nous détaillons les EDs que notre méthode a permis d'extraire de cette source (Figure 5.1 page 130) :

- 18 EDs attendus sur les 25 au total. Par exemple, les EDs `Approved Symbol` ou encore `Chromosome` ;
- 17 EDs références croisées, telles que `GENATLAS`, `OMIM` ou `PMID` ;
- 3 EDs supplémentaires qui apportent une information utile. Ce sont les EDs `Core Data`, `Database Links` et `Gene Symbol Links` qui correspondent en fait à des catégories plus générales regroupant plusieurs EDs. Il peut donc être intéressant d'exploiter ces EDs auxquels les utilisateurs pourraient accéder afin de disposer d'une information plus globale

(c'est-à-dire issue de plusieurs EDs leur appartenant). Par exemple, `Core Data` permet de fournir les données associées aux EDs attendus (c'est-à-dire appartenant au schéma de HGNC) `Approved Symbol`, `Approved Name`, `HGNC ID`, `Status`, `Chromosome`, `Previous Symbols`, `Previous Names` et `Aliases`, ;

– 2 EDs erronés qui correspondent à des informations présentes (au même endroit et dans la même balise d'une page à l'autre) sur la plupart des pages Web mais ne sont pourtant pas informatives. C'est le cas de `Approved` et `Giving unique and meaningful names to every human gene`.

Sur les 25 EDs recensés par la source HGNC, sept ne sont pas extraits pas notre méthode. Ces EDs sont de deux types : deux d'entre eux sont des EDs qui n'ont pas été extraits de HGNC parce que sur les 100 pages Web de l'échantillon que nous avons utilisé, ces EDs n'apparaissaient pas au moins dans 75% des pages (`Misc IDs` et `GDB ID (mapped data)`). Les cinq autres EDs qui n'ont pas été extraits sont des EDs qui sont disponibles lorsque l'on récupère les données de la source HGNC en local mais qui ne sont pas fournis sur les pages Web résultats auxquelles accèdent les utilisateurs qui interrogent le site Web dynamiquement. Ces EDs incluent `Locus Type` et `Date Approved` ;

Nous ne détaillons pas l'ensemble des résultats pour les dix autres sources. Cependant, nous avons vérifié nos résultats et ils sont cohérents. Pour les sources dont le schéma est accessible ou dont un descriptif des attributs utilisés est donné sur leur site Web (précisé dans le tableau 4.1 page 95), nous avons utilisé ces informations pour les comparer aux EDs que nous avons extraits. Dans le cas de GeneLoc, ne disposant pas de ce type d'informations, nous avons vérifié directement sur des pages Web fournies pour des gènes donnés que les EDs extraits étaient corrects. Pour garder le maximum d'automatisation, nous avons décidé de ne pas imposer de validation à cette étape puisque les résultats observés sont intéressants tels quels. Pour ce qui concerne les EDs manquants, nous considérons que cela est du, la plupart du temps, au fait que certains éléments du schéma n'apparaissent pas sur les pages Web fournies par la source et dans ce cas, nous estimons que cela ne constitue pas une information capitale. Ensuite, nous avons regardé en détail les EDs erronés et avons opté pour appliquer un certain nombre de filtres visant à diminuer ce bruit. Par exemple, nous avons éliminé les EDs comprenant plus de 50 caractères (car ils correspondent à des phrases), ceux de moins de 4 caractères (car ils correspondent à des abréviations) et ceux constitués uniquement de chiffres (car ils correspondent à des identifiants donnés).

5.1.2 Traitement des références croisées

Parmi les EDs récupérés se trouvent des références croisées. Ces EDs correspondant à des liens hypertextes, ils sont automatiquement étiquetés comme référence croisée et stockés dans une base de données avec comme nom associé celui de la source référencée. Si le libellé de l'ED est l'identifiant d'une entrée donnée de la source référencée, aucun nom n'est attribué. En revanche, si l'URL extraite existe dans la base, le champ contenant les sources faisant référence à cette entrée est simplement complété par le nom de la source dans laquelle cette référence croisée vient d'être identifiée. Ensuite, l'administrateur du système doit vérifier les nouvelles

Attributs	Noms externes
gd_hgnc_id	HGNC ID
gd_app_sym	Approved Symbol
gd_app_name	Approved Name
gd_status	Status
gd_locus_type	Locus Type
gd_prev_sym	Previous Symbols
gd_prev_name	Previous Names
gd_aliases	Aliases
gd_pub_chrom_map	Chromosome
gd_date2app_or_res	Date Approved
gd_date_mod	Date Modified
gd_date_name_change	Date Name Changed
gd_pub_acc_ids	Accession Numbers
gd_enz_ids	Enzyme IDs
gd_pub_eg_id	Entrez Gene ID
gd_mgd_id	MGD ID
gd_other_ids	Misc IDs
gd_pubmed_ids	Pubmed IDs
gd_pub_refseq_ids	RefSeq IDs
gd_gene_fam_name	Gene Family Name
md_gdb_id	GDB ID (mapped data)
md_eg_id	Entrez Gene ID (mapped data)
md_mim_id	OMIM ID (mapped data)
md_prot_id	UniProt ID (mapped data)
md_refseq_id	RefSeq (mapped data)

EDs attendus
HGNC ID
Approved Symbol
Approved Name
Status
-
Previous Symbols
Previous Names
Aliases
Chromosome
-
-
-
Accession Numbers
Enzyme IDs
Entrez Gene ID
MGD ID
-
Pubmed IDs
RefSeq IDs
-
-
Entrez Gene ID (mapped data)
OMIM ID (mapped data)
UniProt ID (mapped data)
RefSeq (mapped data)

EDs non attendus
PMID
MGD ID
Enz ID
GeneClinics/GeneTests
Vega
Ensembl GeneView
GeneCards
HGNC
Map Viewer
GenBank
Gene
UCSC Browser
GENATLAS
UCSC Index
SwissProt
UniProt
OMIM
Database Links
Core Data
Gene Symbol Links
Giving unique and meaningful names to every human gene
Approved

FIG. 5.1 – **Tableaux représentant les éléments de données répertoriés par HGNC, ceux obtenus par notre méthode et qui sont attendus puis ceux qui sont non attendus.** Le tableau 1 présente les attributs et leurs noms externes utilisés par la source HGNC (http ://www.gene.ucl.ac.uk/cgi-bin/nomenclature/gdlw.pl). Le tableau 2 liste les EDs attendus que nous avons obtenus en rouge et les tirets de même couleur indiquent les EDs manquants. Le tableau 3 présente les EDs non attendus ; les EDs références croisées y sont représentées en vert, les EDs supplémentaires en bleu et les EDs erronés en orange.

CR	URL	SOURCE
LocusLink query	http://www.ncbi.nlm.nih.gov/LocusLink/LocRpt.cgi	mgi,genecards,hgnc,gene
HUGO	http://www.hugo-international.org	hgnc
UCSC	http://genome.cse.ucsc.edu	hgnc
Wellcome trust	http://www.wellcome.ac.uk	hgnc
HUGO / HGNC	http://www.gene.ucl.ac.uk	hgmd,genecards,swissprot,omim,gene,hgnc
GenAtlas	http://www.dsi.univ-paris5.fr/genatlas	hgmd,genecards,hgnc,swissprot
Entrez Gene	http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=gene&cmd=retrieve	hgnc
IMGT	http://imgt.cines.fr	gene,hgnc,gdb,genecards
MRC	http://www.mrc.ac.uk	hgnc
UCL	http://www.ucl.ac.uk	hgnc
PubMed	http://www.ncbi.nlm.nih.gov:80/entrez/query.fcgi?cmd=Retrieve&db=PubMed	hgnc
Swissprot	http://www.expasy.org	hgnc,swissprot
W3C	http://www.w3.org	hgnc
UCSC	http://genome.ucsc.edu	mgi,genecards,hgnc
Ensembl	http://www.ensembl.org	genecards,mgi,hgnc,swissprot
MGD	http://www.informatics.jax.org	genecards,hgnc,swissprot,gene,mgi
OMIM	http://www.ncbi.nlm.nih.gov/entrez/dispomim.cgi	hgnc,gene,hgmd,omim
RefSeq	http://www.ncbi.nlm.nih.gov/entrez/viewer.fcgi	hgnc
GeneTests	http://www.genetests.org	hgnc,genecards,omim
Vega	http://vega.sanger.ac.uk	hgnc
GeneCards	http://bioinfo.weizmann.ac.il/cards-bin	gdb,hgnc,swissprot,hgmd
Human Cell Differentiation Molecules	http://www.hlda8.org	hgnc

FIG. 5.2 – **Références croisées de la source HGNC.** Ces références sont stockées dans une base de données relationnelle. Le premier champ donne le nom de la source référencée, le second son URL et le troisième répertorie l'ensemble des sources y faisant référence. Notons que parmi ces références ne se trouvent pas uniquement des sources de données. En effet, le lien *http ://www.hlda8.org* par exemple correspond à l'URL d'un laboratoire (HCDM) dont l'une des activités est de répertorier des informations concernant des antigènes.

références croisées qui ont été ajoutées dans la base et dont le nom n'a pas été renseigné. Il peut compléter ce nom au travers d'une interface que nous avons développée pour faciliter cette tâche.

Les EDs correspondant à des références croisées sont ajoutés dans le schéma des sources mais ne sont pas mis en correspondance avec le schéma global. À titre d'exemple, la figure 5.2 donne le détail des références croisées identifiées pour la source HGNC.

La figure 5.3 page suivante donne la liste des EDs extraits de HGNC auxquels ont été appliqués les différents filtrages. Seuls les EDs qui vont être mis en correspondance avec le schéma global sont représentés. On remarque la présence de la référence croisée `Enzyme ID`. Le libellé de son hyperlien n'a pas permis de faire le lien avec le nom de la référence croisée dans la base de données. Les EDs supplémentaires sont les mêmes qu'avant les filtrages tandis que l'un des EDs erronés a été éliminé car sa chaîne de caractères est trop longue.

EDs restants
Approved Symbol
Approved Name
Status
Previous Symbols
Previous Names
Aliases
Chromosome
Accession Numbers
Enzyme IDs
Database Links
Core Data
Gene Symbol Links
Approved

FIG. 5.3 – **EDs extraits de la source HGNC à mettre en correspondance avec le schéma global.** Cette liste présente les EDs après filtrage des EDs non conformes aux prérequis et élimination des références croisées.

5.1.3 Typage des éléments de données : Exploitation des valeurs associées

Nous avons cherché à typer l'ensemble des EDs extraits, en considérant les 548 (et non pas les 474 distincts) puisque nous souhaitons justement pouvoir différencier les EDs de même nom mais qui peuvent contenir des informations différentes. Cette méthode nous a permis d'identifier des EDs comme distincts alors qu'ils étaient lexicalement similaires.

D'une manière globale, 62 EDs (11,3% de l'ensemble) ont pu être caractérisés par des types autres que *String* (Tableau 5.1 page ci-contre). 36 EDs ont des valeurs ayant été catégorisées par des types sémantiques de l'UMLS et ont donc été typés comme tels. Les résultats, que nous avons évalués manuellement, sont de différentes natures :

- corrects (11). Un exemple est l'ED `Previous symbols`, extrait de la source HGNC. 90% de ses 46 valeurs non vides ont été catégorisées par le type sémantique GENE OR GENOME, ce qui a permis de déterminer que l'ED `Previous symbols` dans le contexte de HGNC correspond à des anciens symboles de **gènes** (dont la nomenclature a changé). D'autres exemples incluent `Function` et `Component`, extraits de la source MGD et dont les valeurs ont été catégorisées respectivement par les types sémantiques MOLECULAR FUNCTION et CELL COMPONENT ;
- ambigus (21). Par exemple, l'ED `Official Symbol`, extrait notamment de la source Entrez Gene est assigné aux types sémantiques GENE OR GENOME et AMINO ACID, PEPTIDE, OR PROTEIN. En effet, de nombreuses valeurs associées à l'ED `Official Symbol` correspondent en fait aussi bien à des noms de gènes qu'à des noms de protéines. Par exemple, la valeur « BRCA1 » est trouvée dans l'UMLS (par correspondance exacte grâce à ses synonymes) dans un nom de gène (***BRCA1 Gene - C0376571***) et dans un nom de protéine (***BRCA1 Protein - C0259275***) ;

– erronés (4). Cette catégorie correspond aux EDs eux-mêmes erronés à l'origine. Par exemple, `Approved`, extrait de HGNC ou encore `Not Applicable`, issu de GeneCards qui sont présents dans la plupart des pages Web de ces sources mais ne sont pourtant pas des EDs.

Les 26 autres EDs ont été assignés à des types prédéfinis plus généraux que sont *Integer*, *Identifier* et *Sequence*. Pour chaque cas, des exemples sont :

– *Integer* : `Molecular Weight`, un ED extrait de HPRD et dont les valeurs incluent « 207732 » et « 464482 » (en Dalton) pour les gènes « BRCA1 » et « TNXB », respectivement ;
– *Identifier* : `Accession Numbers`, un ED extrait de HGNC et dont les valeurs incluent « U14680 » et « X71923 » (qui sont des identifiants de la source GenBank) pour les gènes « BRCA1 » et « TNXB », respectivement (cf figure 4.4 page 107) ;
– *Sequence* : une illustration de l'ED `Primer 1`, extrait de la source GeneLoc, est donné en figure 4.6 page 110.

Les 486 EDs restants n'ayant pu être typés autrement, on leur attribue le type *String*.

Tab. 5.1 – Résultats obtenus pour le typage des éléments de données extraits des sources.

Type	**Nombre d'EDs de ce type**	**Pourcentage d'EDs de ce type**	**Exemple d'ED typé**
Type sémantique	36	6,6%	`Previous symbols` (Gene or Genome)
Integer	18	3,3%	`Product size`
Identifier	6	1,1%	`Other accession IDs`
Sequence	2	0,3%	`Primer 2`
String	486	88,7%	`Bibliography`

5.1.4 Définition des schémas locaux au format XML

Ces schémas ont été définis dans le langage XML et sont accessibles sur Internet à l'adresse http ://medcin.med.univ-rennes1.fr :81/˜mougin/schemas/.

5.2 Étape 2 : Conception du schéma global

Nous avons utilisé l'UMLS pour constituer le schéma global de notre système. Cette ressource présente des cycles et pour les éliminer, nous avons proposé deux approches possibles. Nous comparons ces approches (naïve et formelle) au moyen de résultats quantitatifs et qualitatifs. L'approche déterminée comme la plus pertinente est utilisée pour définir l'UMLS sous la forme d'un graphe orienté sans cycle. Finalement, nous décrivons notre schéma global au travers d'un sous-ensemble de cet UMLS acyclique.

5.2.1 Élimination des cycles dans l'UMLS

5.2.1.1 Résultats globaux

En comparant les ensembles de descendants pour chaque concept source donné du Metathesaurus obtenus par les approches naïve et formelle respectivement, nous avons identifié quatre cas distincts, présentés dans le tableau 5.2.

1. les ensembles de descendants sont tous les deux **vides** (le concept source est une feuille) ;
2. les ensembles de descendants sont **identiques** ;
3. le parcours du graphe a été interrompu après avoir atteint plus de 50 niveaux de profondeur (avec l'approche naïve). L'ensemble de descendants enregistrés est **incomplet** ;
4. les ensembles de descendants sont complets et **différents**. L'analyse plus approfondie des différences concerne ce groupe.

TAB. 5.2 – Catégories de concepts du Metathesaurus en fonction des différences existant parmi leurs descendants obtenus par les approches naïve et formelle.

Catégorie	**Nombre de concepts sources**	**Pourcentage de concepts sources**
Aucun descendant	765 811	75,0%
Mêmes descendants	221 641	21,7%
Incomplet (interrompu)	6 830	0,7%
Descendants différents	26 584	2,6%
Total	1 020 866	100,0%

5.2.1.2 Nombre de descendants

Nous considérons uniquement les 26 584 concepts sources dont les ensembles de descendants sont complets et présentent des différences. Le nombre de descendants est toujours plus grand avec l'approche naïve (environ 75% de descendants en plus). Plus précisément, quel que soit le concept source, l'ensemble de descendants obtenus avec l'approche formelle est inclus dans celui calculé par l'approche naïve.

5.2.1.3 Cohérence sémantique : aspects quantitatifs et qualitatifs

Comme nous venons de le voir, l'approche naïve apporte un nombre élevé de descendants supplémentaires par rapport à l'approche formelle. Nous avons analysé les types et groupes sémantiques associés à ces descendants et nous avons constaté que (Tableau 5.3 page ci-contre) :

- au niveau quantitatif, le nombre de types et groupes sémantiques est respectivement environ 1,5 fois et 2,3 fois plus élevés avec l'approche naïve ;
- au niveau qualitatif, ces types et groupes sémantiques apparaissent très différents de ceux des concepts sources. Seulement 11% des types sémantiques additionnels sont cohérents avec ceux des concepts sources et 27% des groupes sémantiques additionnels sont cohérents avec ceux des concepts sources.

Des résultats plus détaillés peuvent être trouvés dans [Mougin 05] et [Mougin 06a].

TAB. 5.3 – Résultats quantitatifs et qualitatifs de la cohérence sémantique des descendants additionnels obtenus par l'approche naïve (comparativement aux résultats de l'approche formelle) aux niveaux type sémantique et groupe sémantique.

	Résultats quantitatifs	**Résultats qualitatifs**
	Naïve vs. formelle : % d'additionnels	Cohérence sémantique des additionnels
Type sémantique	49%	11%
Groupe sémantique	127%	27%

5.2.1.4 Exemple

Nous utilisons le concept ***Generally contracted pelvis in pregnancy, labour, and delivery (C0156969)*** pour illustrer les différences observées dans les ensembles de descendants obtenus par les deux approches (Figure 5.4 page suivante). Les types sémantiques de ce concept sont ACQUIRED ABNORMALITY et DISEASE OR SYNDROME.

Avec l'approche formelle, ***C0156969*** a deux descendants : ***C0156971*** et ***C0156972***, catégorisé par au moins un des deux types sémantiques du concept source. L'approche naïve identifie quatre descendants supplémentaires pour ***C0156969*** : ***C0156970*** catégorisé par ACQUIRED ABNORMALITY et DISEASE OR SYNDROME et ses trois enfants : ***C0426852***, ***C0405009*** et ***C0558374***. Le premier est catégorisé par FINDING et les deux autres par ANATOMICAL ABNORMALITY. La relation existant entre ***C0156969*** et ***C0156970*** a été éliminée par l'approche formelle à cause de la présence du mot « unspecified » dans l'un des termes (issu du vocabulaire ICD9CM[1]) constituant le concept ***C0156970*** (terme sous-spécifié causant parfois un cycle - cf 4.2.4.1.4 page 112).

Les descendants directs de ***C0156969*** sont cohérents et compatibles avec le concept source car les deux types sémantiques représentés dans cet ensemble sont les mêmes que ceux du concept source. Par contre, les types sémantiques additionnels obtenus par l'approche naïve catégorisant les descendants de niveau 2 incluent ANATOMICAL ABNORMALITY et FINDING, qui ne sont pas des descendants des types sémantiques du concept source. La cohérence sémantique des descendants calculés par l'approche naïve du point de vue type sémantique est donc de 50% (alors qu'elle est de 100% avec l'approche formelle). Elle est en revanche de 100% du point de vue groupe sméantique car les six descendants de ***C0156969*** appartiennent au groupe sémantique DISORDERS).

5.2.1.5 Conclusion

L'approche formelle réduit le nombre de descendants par rapport à l'approche naïve. Nous avons vu que l'approche naïve obtient tous les descendants de l'approche formelle, plus certains descendants qui lui sont propres. L'approche formelle améliore aussi la cohérence sémantique des ensembles de descendants par rapport à l'approche naïve.

[1] http ://www.cdc.gov/nchs/about/otheract/icd9/abticd9.htm

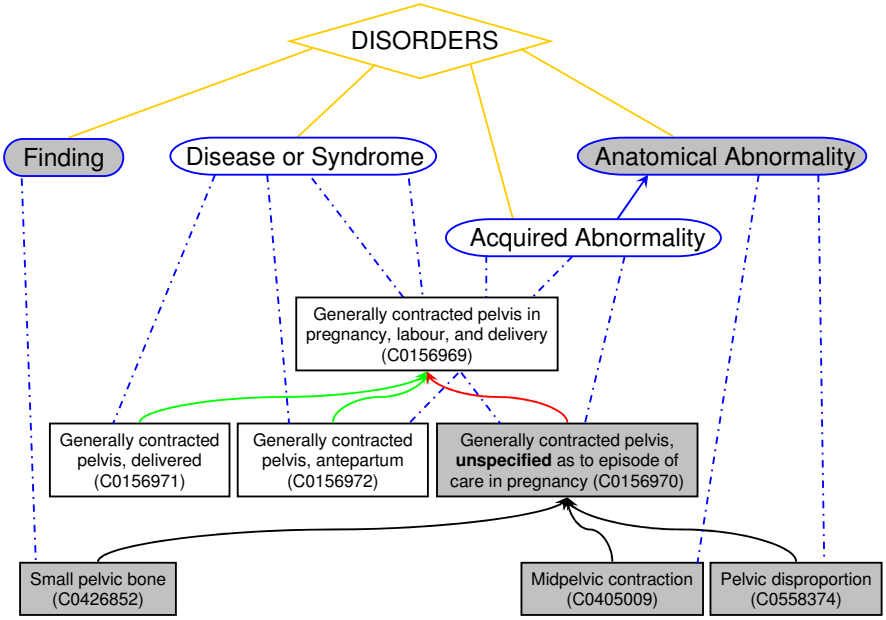


FIG. 5.4 – **Les descendants, types sémantiques, et groupe sémantique du concept *Generally contracted pelvis in pregnancy, labour, and delivery (C0156969)*.** Les conventions de couleur sont les mêmes que précédemment avec en plus les éléments grisés qui sont spécifiques à l'approche naïve et la relation en rouge qui correspond à celle qui est erronée en réalité selon l'approche formelle.

Nous avons également démontré que l'approche formelle sélectionne de manière systématique et cohérente les relations qui doivent être supprimées. Au contraire, c'est l'ordre dans lequel le graphe est parcouru qui détermine quels liens sont ignorés avec l'approche naïve. Il faut également souligner que seule l'approche formelle est reproductible. Finalement, nous avons mis en évidence qu'en pratique, elle nécessite moins de ressources pour construire les ensembles de descendants et, plus généralement, pour parcourir le graphe du Metathesaurus de l'UMLS. Avec l'approche naïve, des profondeurs de plus de 50 niveaux sont au contraire assez communes à cause des relations hiérarchiques non filtrées dans le Metathesaurus, produisant des graphes souvent plus larges, complexes et donc plus difficiles à exploiter.

Pour ces différentes raisons, nous avons choisi d'utiliser l'approche formelle pour transformer l'UMLS en graphe orienté sans cycle.

5.2.2 Définition du schéma global au format OWL

Pour des raisons de simplification, nous avons représenté uniquement la partie de l'UMLS nous étant utile pour décrire les EDs présents dans les sources intégrées. En pratique, cela signifie que nous décrivons l'ensemble des concepts associés à au moins un ED, ainsi que tous leurs ancêtres. Ces ancêtres sont calculés simplement en parcourant le graphe orienté sans cycle, dans le sens ascendant jusqu'à un concept racine (c'est-à-dire n'ayant aucun père). De plus, nous avons intégré l'ensemble des types sémantiques puisqu'ils sont susceptibles d'apporter une information pertinente.

La figure 5.5 page suivante présente une portion du schéma global ainsi construit. L'ensemble du schéma global est accessible à l'adresse http ://medcin.med.univ-rennes1.fr :81/~ mougin/onto/schema_global.owl.

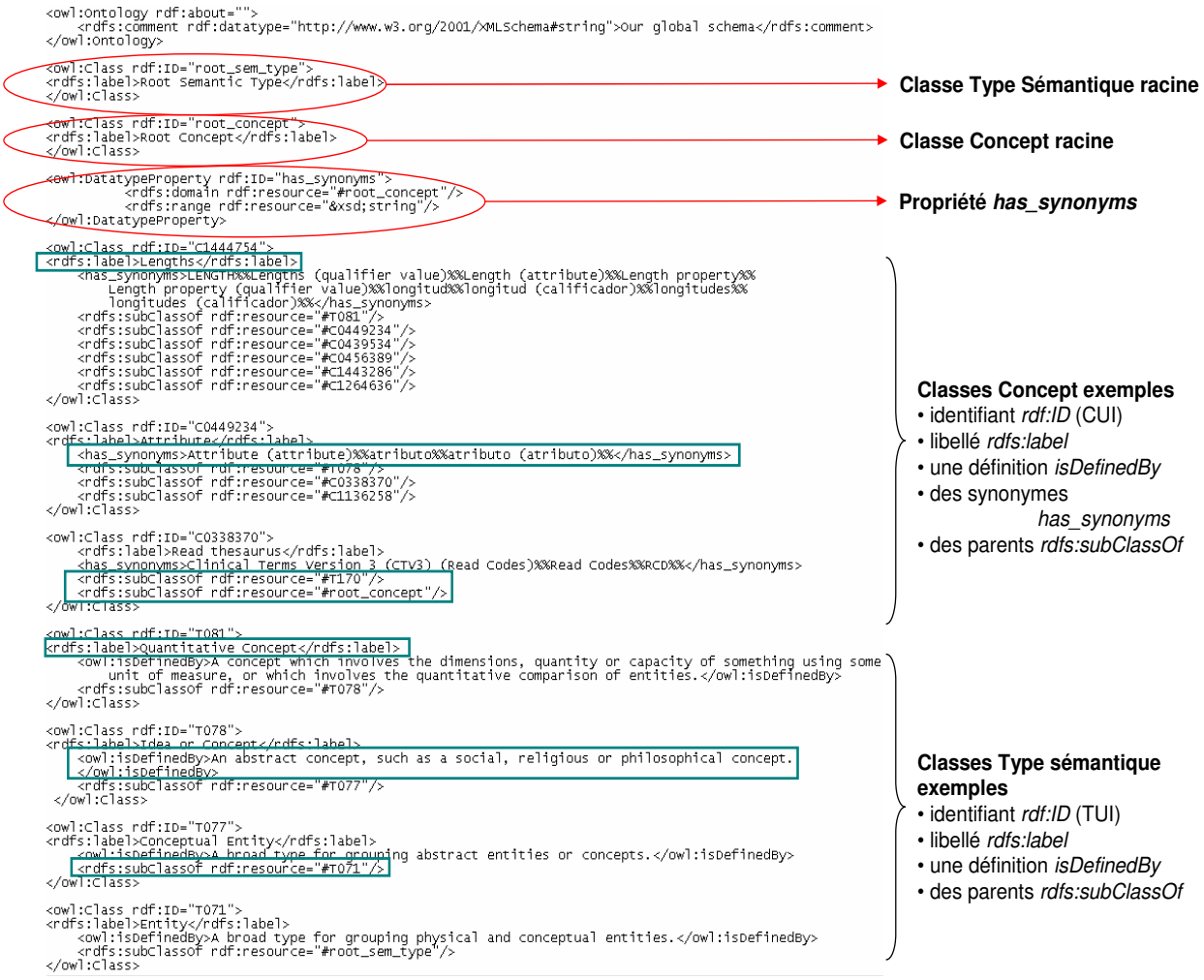


FIG. 5.5 – **Portion du schéma global à partir du concept UMLS *Lengths (C1444754)*.** Sont indiquées les définitions des classes racines et de la propriété ***has_synonyms*** que nous avons définie et suivent des illustrations de classes de concepts et de types sémantiques avec pour chacun un identifiant, un libellé et un ou des parents et optionnellement une définition. En plus, certains concepts ont des synonymes.

5.3 Étape 3 : Mise en correspondance des schémas locaux avec le schéma global

Nous présentons les résultats obtenus lors de la mise en correspondance des EDs dans l'UMLS pour les approches directe et indirecte indépendamment. Nous détaillons ensuite les apports de chaque approche (niveau *schéma*). Enfin, nous présentons les résultats obtenus en comparant les valeurs d'EDs issus de différentes sources (niveau *instances*).

5.3.1 Mise en correspondance directe des éléments de données dans l'UMLS

Sur les 474 EDs extraits, 387 EDs ont été mis en correspondance avec des concepts de l'UMLS. Le tableau 5.4 montre le nombre d'EDs trouvés lors de chaque étape, ainsi que le nombre total de concepts associés à ces EDs. De plus, deux exemples d'EDs sont donnés pour illustrer chaque catégorie.

TAB. 5.4 – Résultats obtenus pour chaque étape de mise en correspondance des éléments de données dans le Metathesaurus de l'UMLS.

Correspondance	**Nombre d'EDs**	**Nombre de concepts associés**	**Exemple d'EDs**	**Concept(s) UMLS associé(s)**
Exacte	135	204	`Molecular Weight` `Northern Blot`	***Molecular Weight (C0026385)*** ***Northern Blot (C1148548)***
Normalisée	20	23	`Cellular component` `Molecular function`	***cellular_component (C1166607)*** ***molecular_function (C1148560)***
Approximative (MetaMap)	232	333	`Gene Symbol` `mRNA sequence`	***Genes (C0017337)*** ***Symbol (C0679214)*** ***RNA, Messenger (C00035696)***

Nous avons également étudié la répartition des résultats au niveau des types sémantiques (tableau 5.5 page suivante). Cela donne une idée du type d'informations que représentent les EDs. Le type sémantique le plus représenté, c'est-à-dire celui sous lequel sont catégorisés le plus grand nombre des concepts associés aux EDs, est INTELLECTUAL PRODUCT qui correspond à des concepts génériques et donc peu spécifiques du domaine biomédical. C'est le cas notamment des EDs `Synonyms`, `Nomenclature` et `Database`. La catégorisation sémantique des EDs aide également à évaluer la qualité des correspondances identifiées. Par exemple, des associations d'EDs avec des concepts catégorisés sous le type sémantique MEDICAL DEVICES, c'est-à-dire corres-

pondant à des appareils médicaux (comme le scanner), sont probablement incorrectes.

TAB. 5.5 – Répartition des éléments de données sous les types sémantiques (les plus représentés) du Réseau Sémantique de l'UMLS. Dans chaque cas, un exemple d'EDs est présenté ainsi que le concept avec lequel il a été mis en correspondance. On notera que ces concepts peuvent être catégorisés par d'autres types sémantiques que celui représenté dans ce tableau.

Nombre de concepts	**Type sémantique**	**Exemple d'EDs**	**Concept associé**
37	INTELLECTUAL PRODUCT	`Gene Name`	***Names (C0027365)***
26	FUNCTIONAL CONCEPT	`Genomic context`	***Context (C0542559)***
25	QUALITATIVE CONCEPT	`Mutation type`	***Type (C0332307)***
19	SPATIAL CONCEPT	`Site of expression`	***Site (C0205145)***
17	NEOPLASTIC PROCESS	`Malignant neoplasms`	***Malignant neoplasms (C0006826)***
17	QUANTITATIVE CONCEPT	`Sensitivity`	***Statistical sensitivity (C0036667)***
16	PHARMACOLOGIC SUBSTANCE	`Drug similarity`	***Drugs (C0013227)***
14	BODY SYSTEM	`Immune system`	***Immune system (C0020962)***
14	DISEASE OR SYNDROME	`Disorders & mutations`	***Disease (C0012634)***

La répartition du nombre d'EDs dans les différentes catégories de correspondances est la suivante :

- correspondance unique : 187 EDs distincts (39,5%) ;
- correspondances multiples : 200 EDs distincts (42,1%) ;
- aucune correspondance : 87 EDs distincts (18,4%).

5.3.2 Mise en correspondance indirecte des éléments de données dans l'UMLS

Pour améliorer et valider les correspondances des EDs identifiées directement dans l'UMLS, la deuxième étape consiste à utiliser une ressource externe : WordNet.

5.3.2.1 Mise en correspondance des éléments de données dans WN

Dans un premier temps, nous avons déterminé le nombre d'EDs trouvés dans WN et la catégorie (c'est-à-dire unique, multiple ou aucune) à laquelle les correspondances identifiées appartiennent. Le détail des résultats obtenus avant et après la désambiguïsation est donnée dans le tableau 5.6 page ci-contre[2]. Le nombre total d'EDs trouvés dans WN diminue logiquement (de 429 à 394) puisque certains synsets initialement proposés ont été supprimés dans le cas où

[2]Nous avons inclus dans la désambiguïsation la phase où les synsets associés partiellement à un ED lorsqu'un concept unique de l'UMLS est identifié de manière totale ont été éliminés. C'est le cas de l'ED `Northern Blot`

la méthode directe était plus pertinente. La désambiguïsation augmente aussi logiquement le nombre d'EDs non trouvés dans WN. Le nombre de correspondances multiples a diminué (de 324 à 135) tandis que celui des correspondances uniques a augmenté (de 105 à 259), prouvant quantitativement que la désambiguïsation est effective.

TAB. 5.6 – Résultats quantitatifs obtenus pour la mise en correspondance des éléments de données (EDs) dans WordNet. Les résultats sont donnés avant et après désambiguïsation. Pour chaque cas, le nombre d'EDs trouvés est d'abord donné puis leur répartition en fonction de la catégorie de correspondance trouvée et enfin le nombre total de synsets distincts associés à ces EDs.

	Nombre d'EDs trouvés	**Correspondances uniques**	**Correspondances multiples**	**Aucune correspondance**	**Nombre de synsets distincts**
Avant désambiguïsation	429 (90,5%)	105	324	45	1,878
Après désambiguïsation	394 (83,1%)	259	135	80	558

5.3.2.2 Mise en correspondance des synsets WN avec des concepts UMLS

Dans un deuxième temps, les synsets WN ont été mis en correspondance avec les concepts de l'UMLS. Parmi les 394 EDs trouvés dans WN, 339 ont été mis en correspondance indirectement dans l'UMLS. En particulier, sur les 87 EDs non trouvés dans l'UMLS avec la méthode directe, 36 ont pu l'être via WN. Ainsi, le nombre d'EDs trouvés dans l'UMLS passe de 387 à 423. Le détail des 36 correspondances indirectes trouvées via WN est le suivant :

Grâce aux synonymes présents dans WN, 16 correspondances indirectes d'EDs dans l'UMLS ont pu être suggérées. Par exemple, l'ED `Topology` n'a pas été trouvé directement dans l'UMLS car aucun concept UMLS n'a le mot « topology » comme synonyme. Cependant, cet ED est associé au synset ***topology#n#2***, qui lui-même a pour synonyme « regional anatomy ». Contrairement à « topology », « regional anatomy » est trouvé dans l'UMLS. L'ED `Topology` peut donc être mis en correspondance avec le concept UMLS ***Regional anatomy (C0002812)***, de manière indirecte au travers d'un synonyme présent dans WN.

Les hypernymes directs de WN ont permis d'identifier 20 correspondances indirectes additionnelles. Par exemple, l'ED `Product` n'est pas présent dans l'UMLS. Via WN, il est associé au synset ***product#n#4*** qui n'a pas de synonyme mais dont l'hypernyme direct (c'est-à-dire son père) « chemical substance » est un concept de l'UMLS (***C0220806***). On obtient là aussi une correspondance indirecte dans l'UMLS pour l'ED `Product`. Par ailleurs, l'ED `Contributor` qui n'existe pas non plus dans l'UMLS est associé à deux synsets : ***contributor#n#1*** ayant pour hypernyme direct « Donor » qui est présent dans l'UMLS (***C0013018***) et ***contributor#n#2***

déjà introduit dans la section Méthodes (cf 4.2.5.3 page 122) ; il est associé à deux mots séparés dans WN alors qu'il existe tel quel dans l'UMLS.

dont les hypernymes directs « Writer » et « Author » sont aussi dans l'UMLS (***C0341628*** et ***C0221192***, respectivement). Dans ce genre de cas, une intervention manuelle est nécessaire pour déterminer laquelle des correspondances indirectes trouvées est la meilleure.

5.3.3 Comparaison des approches directe et indirecte

Nous donnons d'abord quelques résultats globaux obtenus avec chaque approche. Nous présentons ensuite l'apport de chacune d'elle en commençant par l'approche directe. Puis nous rappelons que WN a permis d'identifier de nouvelles correspondances dans l'UMLS, de désambiguïser des correspondances multiples et de valider des correspondances uniques. La validation de ces résultats puis un exemple les illustrant sont finalement présentés.

5.3.3.1 Résultats globaux

Nous avons donc vu que parmi les 474 EDs constituant notre ensemble initial, 387 ont été mis en correspondance directement dans l'UMLS et 339 via WN.

15 EDs (3,2%) ont été trouvés uniquement dans l'UMLS, notamment `SNPs*` (Polymorphism, Single Nucleotide), `RT-PCR*` (Reverse Transcriptase Polymerase Chain Reaction) et `Micro-lesions`. Cela n'est pas surprenant dans la mesure où ces EDs sont très spécifiques du domaine biomédical.

Parallèlement, 55 EDs (11,6%) ont été trouvés uniquement dans WN, et n'ont donc pas pu être mis en correspondance dans l'UMLS. Des exemples sont `Homology`, `Lineage`, `Products`, `Pathways`, `Transcripts` ou encore `Motifs`. En opposition à la remarque ci-dessus, ces EDs sont plutôt génériques et pour cette raison, il n'est pas étonnant qu'ils soient absents de l'UMLS.

Finalement, 30 EDs (6,3%) n'ont été trouvés ni dans l'UMLS ni dans WN. C'est le cas notamment de `Paralogs`, `Ortholog` et `TaxID`.

5.3.3.2 Apport de l'approche directe

Pour 89 cas de correspondances uniques dans l'UMLS, l'approche directe est plus pertinente. Celles-ci n'ont pas pu être validées par les associations identifiées via WN car ces EDs ont été mis en correspondances avec plusieurs synsets WN. Par exemple, l'ED `All beta proteins` est associé au concept ***Beta Protein (C1158818)*** de manière unique mais à quatre synsets WN.

5.3.3.3 Apport de l'approche indirecte

Nous avons vu précédemment que **l'approche indirecte a permis d'identifier 36 nouvelles correspondances** d'EDs dans l'UMLS (cf 5.3.2.2 page précédente).

D'autre part, **l'approche indirecte a permis de désambiguïser 95 des 200 correspondances multiples** d'EDs trouvées dans l'UMLS avec l'approche directe. Presque toutes ces correspondances ont été désambiguïsées au travers de définitions similaires entre le synset et l'un des concepts candidats sauf une qui a été désambiguïsée grâce à l'existence d'ancêtres communs. Par exemple, l'ED `Protein` résulte en trois concepts UMLS (***Protein***, ***Protein measurement*** et ***Protein location***) avec l'approche directe. Grâce à l'association de ce même ED avec le

synset ***protein#n#1***, nous avons pu sélectionner de manière automatique le concept UMLS ***Protein*** puisque sa définition est similaire à celle du synset.

74 correspondances multiples n'ont pas pu être désambiguïsées parce qu'aucune paire (concept, synset) n'a pu être déterminée comme étant la meilleure. Les 31 correspondances restantes n'ont pas pu être désambiguïsées parce que les EDs en question n'existaient pas dans WN.

Enfin, **l'approche indirecte a permis de valider 98 des 187 correspondances uniques** d'EDs obtenues dans l'UMLS avec l'approche directe. Par exemple, la mise en correspondance de l'ED `mRNA sequence` avec le concept ***RNA, Messenger (C0035696)*** est validée par le synset ***mrna#n#1*** parce que leurs définitions sont similaires. Des éléments communs dans les définitions incluent « template for protein synthesis », « nucleus » et « RNA ».

5.3.3.4 Validation

L'approche indirecte permet de valider de manière automatique 177 correspondances[3] sur les 423 trouvées au total dans l'UMLS. Par exemple, la mise en correspondance de l'ED `Length` avec le concept ***Lengths (C1444754)*** est validée par le synset ***length#n#4*** parce qu'ils partagent des ancêtres communs : « Size » (***C0456389***) et « Attribute » (***C0449234***).

Ainsi, 246 correspondances[4] nécessitent une intervention humaine. Nous avons vérifié manuellement leur validité et compte tenu de la redondance existant entre les EDs issus de sources distinctes, nous avons eu à valider uniquement 52 uniques et 92 multiples[5].

Au total, nous sommes parvenus à mettre en correspondance 394 EDs dans l'UMLS (seules 23 correspondances étaient incorrectes). Les 80 EDs restants n'ont pu être trouvés dans l'UMLS pour différentes raisons :

- 30 EDs ne sont trouvés ni dans l'UMLS ni dans WN. Certains de ces EDs sont intéressants (par exemple, `Paralogs`) et les autres correspondent à des références croisées que notre méthode d'extraction n'a pu identifier (par exemple, `TaxID`) ;
- 50 EDs sont incorrects (par exemple, `See Also`).

5.3.3.5 Exemple

Pour illustrer l'ensemble du processus de contribution de WN, nous décrivons la mise en correspondance dans l'UMLS de l'ED `Transcription data`, extrait de la source GeneCards (Figure 5.6 page suivante). Dans l'UMLS, une correspondance partielle est trouvée avec le concept ***Transcription, Genetic (C0040649)***. Dans WN, deux correspondances partielles sont identifiées : cinq synsets sont trouvés pour « transcription » et deux pour « data ». La désambiguïsation de WN pour « transcription » est instantanée : le deuxième synset est du domaine *Genetics* donc c'est celui-ci qui est gardé (cf 4.2.5.2.1 page 120). Pour « data », c'est le

[3]Ce nombre correspond aux cas suivants : 1) correspondances multiples désambiguïsées par l'approche indirecte ; 2) correspondances uniques validées par l'approche indirecte avec au moins un des critères de ressemblance vérifié entre concepts et synsets

[4]Ce nombre correspond aux cas suivants : 1) correspondances uniques où l'approche directe est plus pertinente (WN ne peut donc pas etre utilisé pour la validation) ; 2) correspondances uniques pour lesquelles aucun critère de ressemblance n'est vérifié ; 3) correspondances multiples qui n'ont pu être désambiguïsés ; 4) correspondances trouvées par l'approche indirecte (parfois multiples, voir l'exemple de l'ED `Contributor` - cf 5.3.2.2 page 141)

[5]Cette validation a été faite grâce à une interface Web que nous avons développée

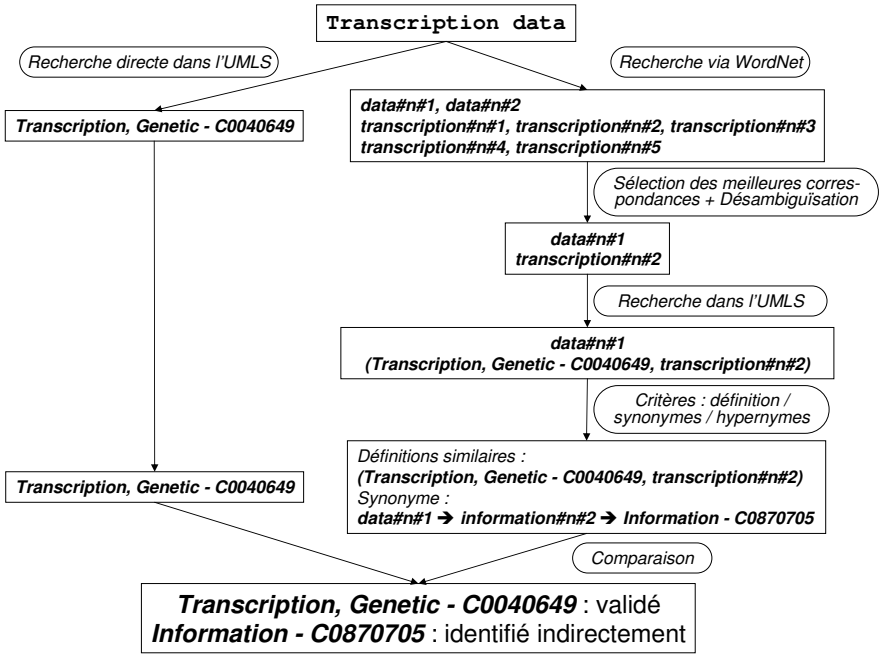


FIG. 5.6 – **Exemple du processus de mise en correspondance des éléments de données dans l'UMLS et WordNet pour l'ED** `Transcription Data`**.**

synset ***data#n#1*** qui est choisi car il a pour synonyme « information » qui appartient à l'ensemble des EDs que nous avons extraits, contrairement aux synonymes de l'autre synset associé à « data » (cf 4.2.5.2.2 page 121). À partir des deux correspondances trouvées directement dans l'UMLS et via WN, il est possible de :

- confirmer que la correspondance avec le concept ***Transcription, Genetic*** est correcte étant donnée la similarité de sa définition avec celle du synset ***transcription#n#2***[6] ;
- proposer une correspondance indirecte du mot « data » avec le concept UMLS ***Information (C0870705)***, au travers d'un synonyme du synset ***data#n#1*** qui avait été associé à l'ED directement.

5.3.4 Mise en correspondance des éléments de données au niveau *instances*

Nous avons calculé le coefficient de Jaccard pour toutes les paires d'EDs issus de sources distinctes. La similarité au niveau des valeurs des EDs nous permet de compléter et valider les correspondances des EDs dans l'UMLS.

11 paires d'EDs ont un indice de Jaccard de plus de 0,5. Celles-ci ont **permis d'identifier de nouvelles correspondances dans l'UMLS**. Par exemple, les valeurs de l'ED `Official Symbol` de la source Entrez Gene sont similaires à celles de l'ED `Gene Symbol`, extrait de HPRD

[6]La définition de ***Transcription, Genetic*** est « *The biosynthesis of RNA carried out on a template of DNA. The biosynthesis of DNA from an RNA template is called REVERSE TRANSCRIPTION* » et celle de ***transcription#n#2*** est « *(genetics) the organic process whereby the DNA sequence in a gene is copied into mRNA ; the process whereby a base sequence of messenger RNA is synthesized on a template of complementary DNA* »

(indice de 0,55). Cela indique que l'ED `Official Symbol` contient des symboles de **gènes** officiels (et non de protéines ou autres). Une nouvelle correspondance est ainsi identifiée entre cet ED et le concept UMLS ***Genes (C0017337)***.

Avec cet exemple, on constate également que l'approche au niveau *instances* **permet de valider des correspondances existantes**. Les correspondances des deux EDs avec le concept ***Symbols (C0679214)*** sont validées car la similarité des valeurs confirme qu'il s'agit bien là de symboles.

Par ailleurs, nous avons constaté que cette approche permet d'**éliminer des correspondances** identifiées de manière terminologique. En effet, l'ED `Gene Name`, issu de Entrez Gene, et l'ED `Approved Symbol`, extrait de HGNC, ont une similarité de 0,92[7]. Le nom du premier ED indique qu'il concerne des noms de gènes alors que le second contient des symboles. Il y a donc une incohérence dans le nom d'un des EDs, ce qui peut poser problème aux utilisateurs qui s'attendent à accéder à une information mais en obtiennent une autre. Or, l'ED `Approved Symbol` est également mis en correspondance avec les EDs `Official Symbol` de Entrez Gene et `Gene Symbol` de HGMD et HPRD, ce qui signifie que c'est cet ED qui est correct. C'est donc `Gene Name` qui porte un nom inadapté à son contenu. Dans ce cas, l'approche basée *instances* permet deux choses : éliminer la correspondance entre l'ED `Gene Name` et le concept ***Names (C0027365)*** et ajouter une correspondance entre ce même ED et le concept ***Symbols***.

11 correspondances dont l'indice de Jaccard est compris entre 0,2 et 0,5 ont été identifiées. Ces résultats indiquent que la similarité des valeurs des EDs concernés est assez basse mais ils peuvent malgré tout être utiles. En effet, ils permettent notamment d'identifier une correspondance supplémentaire entre l'ED `Chromosome` de HGNC et le concept ***Location (C0450429)***. Les valeurs de cet ED coïncident, avec un indice compris entre 0,23 et 0,30, avec celles de l'ED `Location` extrait de la source Entrez Gene et `Chromosomal location` issu de HGMD ainsi que de l'ED `Gene map locus` issu de HPRD.

Le détail des 21 correspondances ayant un indice de Jaccard supérieur à 0,2 est donné dans le tableau 5.7 page suivante. Pour les 11 cas supérieurs à 0,5, les correspondances sont validées automatiquement car leur pourcentage de valeurs communes est haut. Pour les 11 autres correspondances, une validation par un expert a été jugée nécessaire. Cette approche, basée *instances*, permet donc de découvrir de nouvelles correspondances dans l'UMLS, de valider des correspondances existantes et même d'en éliminer des incorrectes.

[7]Sur les 100 pages obtenues lors de l'interrogation de ces sources, ces EDs contenaient chacun 96 valeurs non vides et 92 sont communes aux deux EDs. Leur indice de Jaccard est donc de 0,92.

TAB. 5.7 – Paires d'éléments de données mis en correspondance au travers de leurs valeurs. Tous les résultats pour lesquels l'indice de Jaccard supérieur à 0,2 sont donnés.

Élément de données 1	**Source 1**	**Élément de données 2**	**Source 2**	**Indice de Jaccard**
`Gene Name`	Entrez Gene	`Approved Symbol`	HGNC	0,92
`Approved Symbol`	HGNC	`Official Symbol`	Entrez Gene	0,81
`Approved Symbol`	HGNC	`Gene Symbol`	HGMD	0,8
`Gene Name`	Entrez Gene	`Gene Symbol`	HGMD	0,75
`Gene Symbol`	HPRD	`Approved Symbol`	HGNC	0,64
`Gene Symbol`	HGMD	`Official Symbol`	Entrez Gene	0,64
`Approved Name`	HGNC	`Gene Description`	Entrez Gene	0,63
`Gene Name`	Entrez Gene	`Gene Symbol`	HPRD	0,61
`Gene Symbol`	HPRD	`Gene Symbol`	HGMD	0,59
`Official Symbol`	Entrez Gene	`Gene Symbol`	HPRD	0,55
`Genetic Association Database`	GeneCards	`Gene Symbol`	HGMD	0,52
`Gene Name`	HGMD	`Approved Symbol`	HGNC	0,36
`Official Symbol`	Entrez Gene	`Gene Name`	HGMD	0,36
`Gene Name`	HGMD	`Gene Name`	Entrez Gene	0,35
`Gene Symbol`	HGMD	`Gene Name`	HGMD	0,35
`Gene Symbol`	HPRD	`Gene Name`	HGMD	0,31
`Location`	Entrez Gene	`Chromosome`	HGNC	0,3
`Gene map locus`	HPRD	`Chromosome`	HGNC	0,3
`Location`	Entrez Gene	`Gene map locus`	HPRD	0,29
`Chromosomal location`	HGMD	`Chromosome`	HGNC	0,23
`Gene map locus`	HPRD	`Chromosomal location`	HGMD	0,21
`Chromosomal location`	HGMD	`Location`	Entrez Gene	0,21

En conclusion, nous avons montré que les méthodes développées pour faciliter la conception de notre système sont efficaces. Plus précisément, nous acquérons automatiquement les schémas locaux et nous avons créé un schéma global cohérent décrit dans un langage du Web sémantique et ré-utilisant une ressource terminologique existante. Enfin, les mises en correspondance entre les schémas locaux et le schéma global peuvent être réalisées de manière semi-automatique.

Chapitre 6

Le système

Dans ce chapitre, nous présentons le prototype du système que nous avons conçu à partir des méthodes décrites dans les chapitres précédents. Ses composants principaux, son architecture globale et le processus de requêtes mis en œuvre sont tout d'abord décrits. Puis quelques exemples illustrent le type de requêtes qu'il est possible de réaliser avec notre système. Nous montrons ensuite comment l'évolution de notre système est gérée au travers de l'intégration d'une nouvelle source mais aussi lors d'éventuelles modifications des sources déjà intégrées. Le détail des étapes est donné en précisant celles qui sont automatiques et celles qui nécessitent l'intervention humaine. Enfin, nous synthétisons ces différents points en précisant le positionnement de notre travail par rapport à l'approche LAV (*Local-As-View*) introduite dans l'état de l'art (cf 2.2.3.3.1 page 54).

6.1 Description du système

6.1.1 Composants

6.1.1.1 Médiateur

Le **médiateur** est constitué de deux éléments : le schéma global et l'ensemble de correspondances identifiées entre le schéma global et les schémas locaux.

Tout d'abord, **le schéma global** a été enrichi avec des informations issues de WN. Pour chaque concept mis en correspondance avec un ED au travers de WN, les propriétés du synset associé sont ajoutées dans le schéma global si au moins un des trois critères de similarité (définitions similaires, synonymes et/ou ancêtres communs) est vérifié. Par exemple, la correspondance de l'ED `mRNA variant` avec le concept ***RNA, Messenger (C0035696)*** est validée par l'approche indirecte car la définition de ce concept est similaire à celle du synset ***mrna#n#1***. Le concept ***C0035696*** est complété avec les propriétés suivantes du synset :

- sa définition : *the template for protein synthesis ; the form of RNA that carries information from DNA in the nucleus to the ribosome sites of protein synthesis in the cell* ;
- ses synonymes « mRNA », « template RNA » et « informational RNA » ;
- ses ancêtres, c'est-à-dire ***ribonucleic_acid#n#1***, ***polymer#n#1***, ***compound#n#1***, ***substance#n#1***, ***physical_entity#n#1*** et ***entity#n#1***.

Grâce à ces propriétés additionnelles associées aux différents concepts UMLS ayant été mis en correspondance avec un synset, le schéma global contient des informations supplémentaires susceptibles d'être utiles lors du processus de requêtes (voir plus loin 6.1.3 page 152).

L'intégration des propriétés des synsets WN est faite de manière automatique dans le cas où des correspondances de cardinalité 1-1 (un concept pour un synset) ont pu être trouvées et validées par au moins un des critères pré-cités. Comme nous l'avons indiqué précédemment lors de la description du schéma global au format OWL (cf 5.2.2 page 137), nous avons décrit un sous-ensemble de l'UMLS dans ce format puisque cette ressource est très large et seule une partie de son contenu nous intéresse concrètement ; celle permettant de représenter les EDs que nous avons extraits des sources. Dans cet ensemble, 106 concepts UMLS sont enrichis au moyen des propriétés suivantes :

- *has_WN_definition* ;
- *has_WN_synonyms* ;
- *has_WN_hypernyms*.

Ces trois propriétés sont de type *DataTypeProperty* avec pour domaine la classe *Root_Concept*, et pour co-domaine un élément de type *String*. Le schéma ainsi complété est accessible à l'adresse http ://medcin.med.univ-rennes1.fr :81/˜mougin/ onto/schema_global_with_wn.owl. La figure 6.1 illustre la même portion du schéma global que celle de la figure 5.5 page 138 à laquelle ont été ajoutées les propriétés du synset WN ***length#n#4*** associé au concept UMLS ***Lengths (C1444754)***. Sur cette petite partie du schéma global, seul ce concept est enrichi car il a été mis en correspondance avec un ED, `Length`, contrairement aux autres concepts.

L'autre composant du médiateur est l'**ensemble des correspondances existant entre les EDs et les concepts du schéma global**. Grâce à ces informations, le médiateur va identifier les EDs qui sont pertinents par rapport aux termes constituant la requête. Il envoie ensuite ces EDs aux adaptateurs. Dans un deuxième temps, il se charge d'unifier les résultats que les adaptateurs des sources contenant les EDs pertinents lui retournent.

6.1.1.2 Adaptateurs

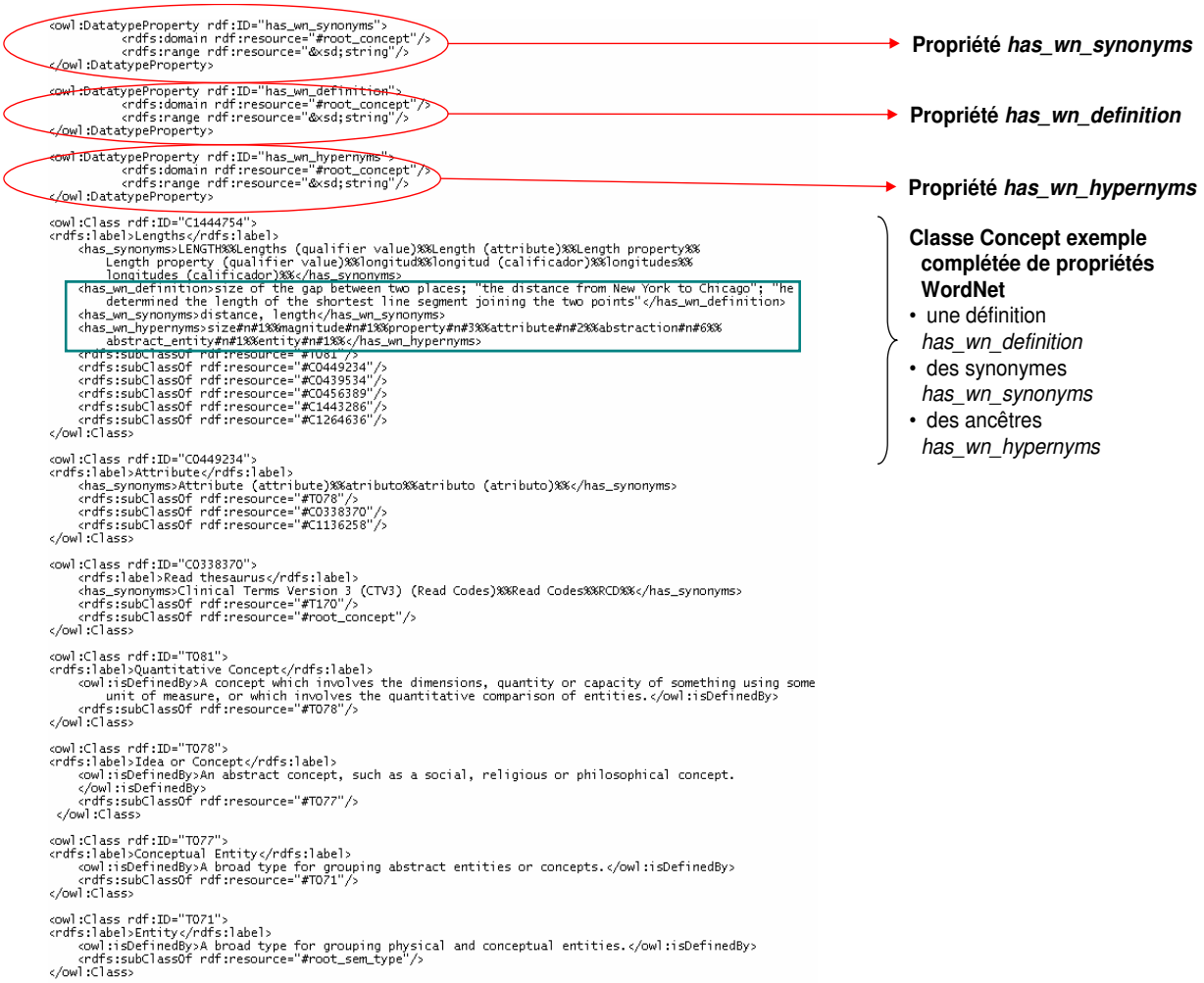
Les adaptateurs sont les composants faisant le lien entre les sources et le médiateur. Ils se chargent de récupérer les données correspondant aux requêtes posées par les utilisateurs et les fournissent au médiateur. Notre système est constitué d'un adaptateur par source et chacun d'eux est basé sur les deux éléments suivants :

- le schéma local de la source, recensant les informations importantes la concernant ;
- le programme permettant d'interroger dynamiquement la source à laquelle il est associé. Celui-ci existe déjà car c'est le même qui a permis de récupérer les valeurs associées aux éléments de données.

L'adaptateur doit récupérer les valeurs associées aux EDs que le médiateur lui a transmis. Pour cela, l'adaptateur appelle le programme interrogeant la source qu'il gère à partir d'informations se trouvant dans son schéma local et récupère les données associées à chaque ED pertinent ainsi que les éventuelles références croisées présentes sur la page résultat correspondant à l'entité (ou les) intéressant les utilisateurs. L'information obtenue est donc très précise puisque, au lieu de fournir au médiateur la page Web entière répondant à la requête des utilisateurs, seules les valeurs des EDs identifiés comme pertinents pour y répondre sont rendues en résultat.

6.1.2 Architecture globale

L'architecture globale est constituée des éléments traditionnels présents dans un système basé sur l'approche médiateur : les adaptateurs contenant les schémas locaux (qui ont déjà été présentés dans le chapitre précédent - 5.1.4 page 133) et le médiateur 6.1.1.1 page précédente. Son schéma global ré-utilise des ressources terminologiques existantes, et plus précisément a comme noyau l'UMLS complété par WN.




```
<owl:DatatypeProperty rdf:ID="has_wn_synonyms">
          <rdfs:domain rdf:resource="#root_concept"/>
          <rdfs:range rdf:resource="&xsd;string"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="has_wn_definition">
          <rdfs:domain rdf:resource="#root_concept"/>
          <rdfs:range rdf:resource="&xsd;string"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="has_wn_hypernyms">
          <rdfs:domain rdf:resource="#root_concept"/>
          <rdfs:range rdf:resource="&xsd;string"/>
</owl:DatatypeProperty>

<owl:Class rdf:ID="C1444754">
<rdfs:label>Lengths</rdfs:label>
    <has_synonyms>LENGTH%%Lengths (qualifier value)%%Length (attribute)%%Length property%%
        Length property (qualifier value)%%longitud%%longitud (calificador)%%longitudes%%
        longitudes (calificador)%%</has_synonyms>
    <has_wn_definition>size of the gap between two places; "the distance from New York to Chicago"; "he
        determined the length of the shortest line segment joining the two points"</has_wn_definition>
    <has_wn_synonyms>distance, length</has_wn_synonyms>
    <has_wn_hypernyms>size#n#1%%magnitude#n#1%%property#n#3%%attribute#n#2%%abstraction#n#6%%
        abstract_entity#n#1%%entity#n#1%%</has_wn_hypernyms>
    <rdfs:subClassOf rdf:resource="#T081"/>
    <rdfs:subClassOf rdf:resource="#C0449234"/>
    <rdfs:subClassOf rdf:resource="#C0439534"/>
    <rdfs:subClassOf rdf:resource="#C0456389"/>
    <rdfs:subClassOf rdf:resource="#C1443286"/>
    <rdfs:subClassOf rdf:resource="#C1264636"/>
</owl:Class>

<owl:Class rdf:ID="C0449234">
<rdfs:label>Attribute</rdfs:label>
    <has_synonyms>Attribute (attribute)%%atributo%%atributo (atributo)%%</has_synonyms>
    <rdfs:subClassOf rdf:resource="#T078"/>
    <rdfs:subClassOf rdf:resource="#C0338370"/>
    <rdfs:subClassOf rdf:resource="#C1136258"/>
</owl:Class>

<owl:Class rdf:ID="C0338370">
    <rdfs:label>Read thesaurus</rdfs:label>
    <has_synonyms>Clinical Terms Version 3 (CTV3) (Read Codes)%%Read Codes%%RCD%%</has_synonyms>
    <rdfs:subClassOf rdf:resource="#T170"/>
    <rdfs:subClassOf rdf:resource="#root_concept"/>
</owl:Class>

<owl:Class rdf:ID="T081">
<rdfs:label>Quantitative Concept</rdfs:label>
    <owl:isDefinedBy>A concept which involves the dimensions, quantity or capacity of something using some
        unit of measure, or which involves the quantitative comparison of entities.</owl:isDefinedBy>
    <rdfs:subClassOf rdf:resource="#T078"/>
</owl:Class>

<owl:Class rdf:ID="T078">
<rdfs:label>Idea or Concept</rdfs:label>
    <owl:isDefinedBy>An abstract concept, such as a social, religious or philosophical concept.
    </owl:isDefinedBy>
    <rdfs:subClassOf rdf:resource="#T077"/>
 </owl:Class>

<owl:Class rdf:ID="T077">
<rdfs:label>Conceptual Entity</rdfs:label>
    <owl:isDefinedBy>A broad type for grouping abstract entities or concepts.</owl:isDefinedBy>
    <rdfs:subClassOf rdf:resource="#T071"/>
</owl:Class>

<owl:Class rdf:ID="T071">
<rdfs:label>Entity</rdfs:label>
    <owl:isDefinedBy>A broad type for grouping physical and conceptual entities.</owl:isDefinedBy>
    <rdfs:subClassOf rdf:resource="#root_sem_type"/>
</owl:Class>
```

FIG. 6.1 – **Portion du schéma global à partir du concept UMLS *Lengths (C1444754)*.** À la représentation stricte de l'UMLS s'ajoutent les trois propriétés : `has_WN_definition`, `has_WN_synonyms` et `has_WN_hypernyms`. Le concept ***C1444754*** est ainsi complété par les propriétés du synset ***length#n#4***. En pratique, le concept, pour lequel aucune définition n'existe dans l'UMLS, est donc enrichi avec la définition du synset WordNet associé et ses synonymes et ascendants sont complétés respectivement par les synonymes et les hypernymes propres à WordNet.

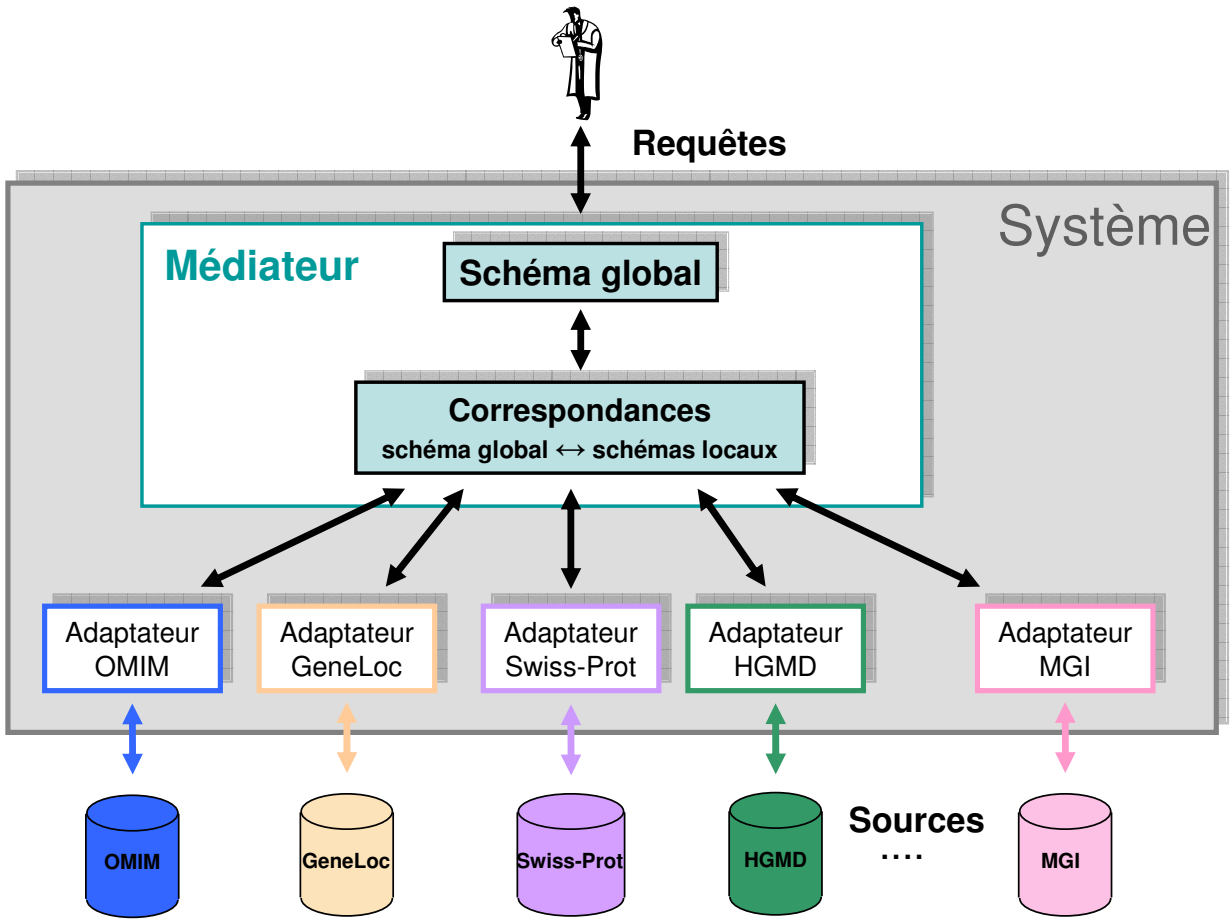


FIG. 6.2 – **Architecture de notre système d'intégration.** Les composants sont les suivants : le médiateur qui comprend le schéma global et l'ensemble de correspondances existant entre les concepts du schéma global et les éléments de données des sources et les adaptateurs qui contiennent les schémas locaux.

Le système est accessible sur Internet à l'URL suivante : http ://medcin.med.univ-rennes1.fr : 81/cgi-bin/mougin/These/main2.pl. Son interface d'interrogation est simple, elle consiste en trois champs que les utilisateurs doivent remplir (Figure 6.3 page ci-contre) :

- une liste de choix entre deux entités biologiques prédéfinies ; à savoir un gène (son nom ou son symbole) ou une maladie ;
- un champ de texte libre où les utilisateurs indiquent la ou les entités précises qui les intéressent (par exemple : les symboles des gènes « BRCA1 », « TNXB » et « HFE » ou leur nom respectif « Breast Cancer 1, early onset », « tenascin XB » et « hemochromatosis » ou encore la maladie « Epilepsy ») ;
- un autre champ de texte libre où les utilisateurs entrent le reste de leur requête, c'est-à-dire le type d'informations qu'ils veulent obtenir concernant la ou les entités données.

6.1.3 Stratégie de requêtes

Nous décrivons le processus de requêtes implémenté dans notre système. Pour illustrer l'intérêt des méthodes développées pour implémenter un tel système, nous proposons un prototype permettant de résoudre des requêtes simples mais impliquant tout de même plusieurs sources de données. En pratique, les requêtes sont traitées comme suit.

Tout d'abord, les utilisateurs posent leurs requêtes au système au travers de l'interface d'interrogation.

La deuxième phase est gérée par le médiateur ; il se charge de repérer les éléments du schéma global qui sont jugés pertinents par rapport à la requête posée. Ainsi, il détermine les concepts pouvant être rattachés aux mots constituant la requête. Plus précisément, le principe est basé sur les propriétés des concepts du schéma global de la manière suivante :

- recherche dans les termes préférés des concepts (les libellés de ceux-ci) ;
- recherche dans les synonymes des concepts ;
- recherche dans les synonymes et hypernymes du synset correspondant, quand ils sont disponibles.

Une fois ces concepts pertinents identifiés, une expansion de requêtes exploitant la hiérarchie est effectuée [Efthimiadis 96]. L'idée est la suivante : si les concepts constituant la requête sont utiles pour répondre à celle-ci, cela peut également être le cas de ses descendants. Ces derniers peuvent effectivement apporter une information plus spécifique (puisque d'un niveau plus fin) et donc possiblement intéressante. Par exemple, si on recherche les dimensions d'un gène donné, le concept UMLS ***Dimensions (C0439534)*** sera recherché parmi les correspondances existantes et même si aucun ED n'est relié à celui-ci, il sera malgré tout possible d'identifier des EDs reliés à ses descendants, comme l'ED `Length`, issu de PDB, car il est associé au concept ***Lengths (C1444754)***, enfant direct de ***C0439534***. C'est pour cette même raison que les hypernymes issus de WN sont utilisés par le médiateur. En effet, si l'un des termes constituant la requête est présent parmi les hypernymes associés à un concept de l'UMLS, alors ce concept est susceptible d'apporter des données intéressantes aux utilisateurs puisqu'il correspond à un descendant d'un terme de la requête. En pratique, cette expansion est effectuée en calculant la fermeture transitive des concepts préalablement identifiés comme pertinents vis à vis de la requête posée

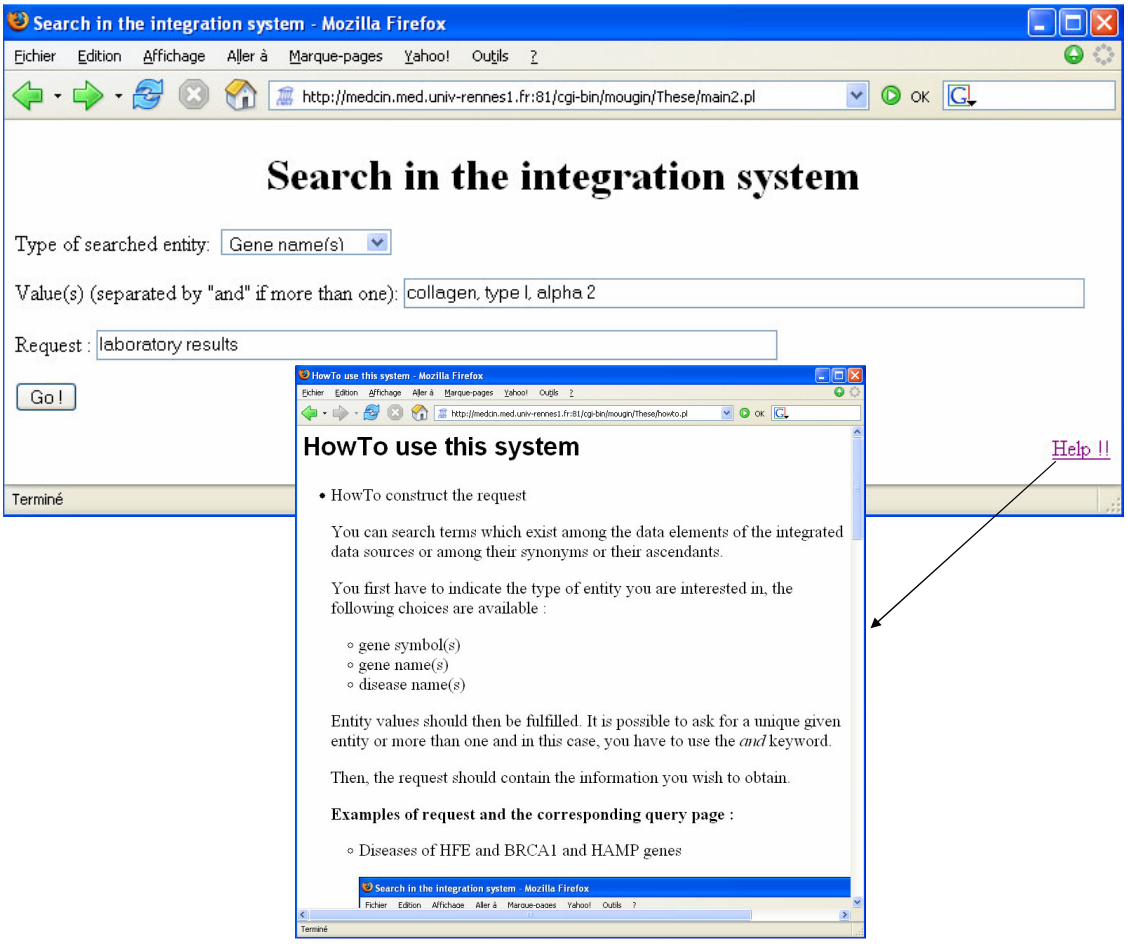


FIG. 6.3 – **Interface d'interrogation de notre système d'intégration.** La copie d'écran supérieure correspond à cette interface avec pour champs à remplir : le type d'entité considérée (au travers d'une liste de choix prédéfinis), valeur(s) associée(s) à cette entité et requête à proprement parler, c'est-à-dire les mots-clés pour lesquels les utilisateurs veulent obtenir des informations. Le lien intitulé « Help ! ! » mène à une page (page du bas) contenant une description succincte de la façon dont les utilisateurs doivent interroger le système, quelques exemples pour les guider ainsi que la liste des éléments de données extraits de chacune des sources intégrées à notre système.

pour obtenir l'ensemble de ses descendants. Ce mécanisme est une méthode classique utilisée en recherche d'informations [Voorhees 94], [Baziz 03]. Une fois les concepts identifiés (y compris les descendants), le médiateur exploite les correspondances entre ces concepts et les EDs. Il envoie les EDs sélectionnés vers les adaptateurs associés aux sources dont ils sont issus. Par ailleurs, il transmet à l'ensemble des adaptateurs les concepts identifiés comme pertinents pour qu'ils puissent vérifier au sein du schéma de la source qu'ils gèrent si le type d'un ED peut être associé à l'un de ces concepts.

Lors de la phase suivante, les adaptateurs vont donc préalablement consulter le type associé aux EDs de la source qu'ils gèrent. Si l'un des types est le même qu'un des concepts fournis par le médiateur alors il peut être intéressant de sélectionner l'ED correspondant. Par exemple, si on recherche des identifiants d'un gène donné, le mot-clé « Identifier » fera a priori partie de la requête. Les EDs ayant été typés comme *Identifier* (par exemple, l'ED `Other accession IDs`, extrait de la source MGI) présentent potentiellement un intérêt pour les utilisateurs.

L'ensemble des EDs identifiés comme pertinents par le médiateur ou l'adaptateur sont ensuite proposés aux utilisateurs (en précisant la source dont ils proviennent). Ils doivent sélectionner ceux qui les intéressent. Une fois le choix effectué, les adaptateurs exécutent les requêtes dans chacune des sources pouvant répondre en partie à la requête. Les résultats obtenus ainsi que l'ensemble des références croisées présentes parmi les valeurs des EDs sélectionnés sont fournis au médiateur. Celui-ci effectue l'intégration verticale des données récoltées en exploitant les correspondances trouvées au niveau *instances* entre les EDs de sources distinctes. Si les informations de deux EDs dont on sait que le contenu est le même sont utilisés pour répondre à la requête, seules les valeurs associées à l'un des deux EDs sont présentées aux utilisateurs (en leur précisant que l'information correspondante est issue des deux sources en question). Enfin, le médiateur fournit les résultats aux utilisateurs, incluant les valeurs associées aux EDs pertinents et les références croisées obtenues dans les sources.

6.1.4 Exemples

Il est rare de disposer d'une évaluation exhaustive de systèmes d'intégration basée médiateur. Dans la littérature, on trouve plutôt des exemples illustrant les fonctionnalités offertes par les systèmes considérés (par exemple, BACIIS [Ben Miled 05] et BioMediator [Mork 01]). Une proposition alternative a été faite pour évaluer le système ONTOFUSION [Alonso-Calvo 06]. Les concepteurs ont tout d'abord réalisé des requêtes « tests » dans chacune des sources qu'ils ont intégrées. Ensuite, ils ont utilisé leur système pour répondre aux mêmes requêtes et ont observé que les informations récupérées étaient bien les mêmes, à la différence près que dans le second cas, une unique interrogation est faite. Il est logique que les résultats obtenus au travers du système soient les mêmes que lorsque l'on interroge directement les sources. Cette proposition n'est pas vraiment utile puisqu'elle montre surtout que la fusion finale des résultats, et en particulier l'intégration verticale, n'est pas gérée par ce système.

En conséquence, nous montrons l'intérêt de notre système au travers d'exemples de requêtes qui impliquent plusieurs des sources intégrées.

6.1.4.1 Synonymie

Le premier exemple illustre l'exploitation de la synonymie pour répondre à une requête, et en particulier celle apportée par WN en complément de l'UMLS. La requête porte sur la maladie nommée « hémochromatose » et on souhaite obtenir des citations sur celle-ci. En utilisant le terme « citation », le concept directement sélectionné dans le schéma global est ***Citation (C0552371)*** auquel est associé l'ED `Primary Citation`, existant dans la source PDB. Dans l'UMLS uniquement, aucun synonyme n'existe pour ce concept. Cependant, au travers du synset ***citation#n#3*** apparié au concept UMLS précité, des synonymes ont été ajoutés à la description du concept dans le schéma global, et notamment « reference ». Grâce à ce dernier, on parvient à identifier d'autres EDs de nom `References` présents dans les sources OMIM, MGD et Swiss-Prot. Ainsi, les résultats fournis aux utilisateurs proviennent de (Figure 6.4 page suivante) :

- OMIM : 229 références concernant l'hémochromatose. Un lien vers la page OMIM associée est donné sur le mot « References » et après chaque citation, l'identifiant PubMed correspondant permet d'accéder directement à cette entrée dans PubMed (au travers d'une référence croisée) ;
- PDB : une référence (ainsi qu'un lien interne vers l'abstract) concernant la structure de la protéine impliquée dans l'hémochromatose ;
- MGD : la publication la plus ancienne sur l'hémochromatose chez la souris ainsi que la plus récente. Un lien permet d'accéder à l'ensemble des 49 références sur ce même sujet ;
- Swiss-Prot : les 23 citations proposées dans cette source concernant la protéine impliquée dans l'hémochromatose avec un lien vers la source dont elles sont issues. Un lien vers les références croisées de cette maladie dans d'autres sources est aussi fourni.

Dans la figure 6.5 page 157, nous avons suivi les liens proposés sur la page résultat de notre système afin d'illustrer l'ensemble des informations auxquelles il est possible d'accéder au travers des références croisées. De plus, les liens internes à une source donnent des précisions par rapport aux informations succinctes généralement présentées sur la page principale concernant une entité donnée.

6.1.4.2 Hiérarchie

Cet exemple illustre l'intérêt d'utiliser la hiérarchie au cours du processus de requêtes. On cherche à obtenir les résultats de procédures de laboratoire effectués sur le gène nommé « collagen, type I, alpha 2 (symbole : COL1A2) » pouvant guider les biologistes dans la réalisation ultérieure d'expériences (Figure 6.6 page 159). La requête posée est celle illustrée dans la figure 6.3 page 153. Aucun ED ne correspond à cette notion de manière directe mais parmi les éléments du schéma global, le type sémantique Laboratory Procedure (T059) est identifié comme pertinent par rapport à la requête. Celui-ci est l'ancêtre de nombreux concepts UMLS (tous ceux catégorisés par ce type sémantique dans l'UMLS). Le médiateur détermine (au travers des correspondances existantes) que les EDs suivants sont associés à certains de ces concepts :

- `Gene Map Locus` et `Linkage mapping` extraits respectivement de HPRD et GeneLoc : ces EDs fournissent des informations concernant la localisation chromosomique du gène ;

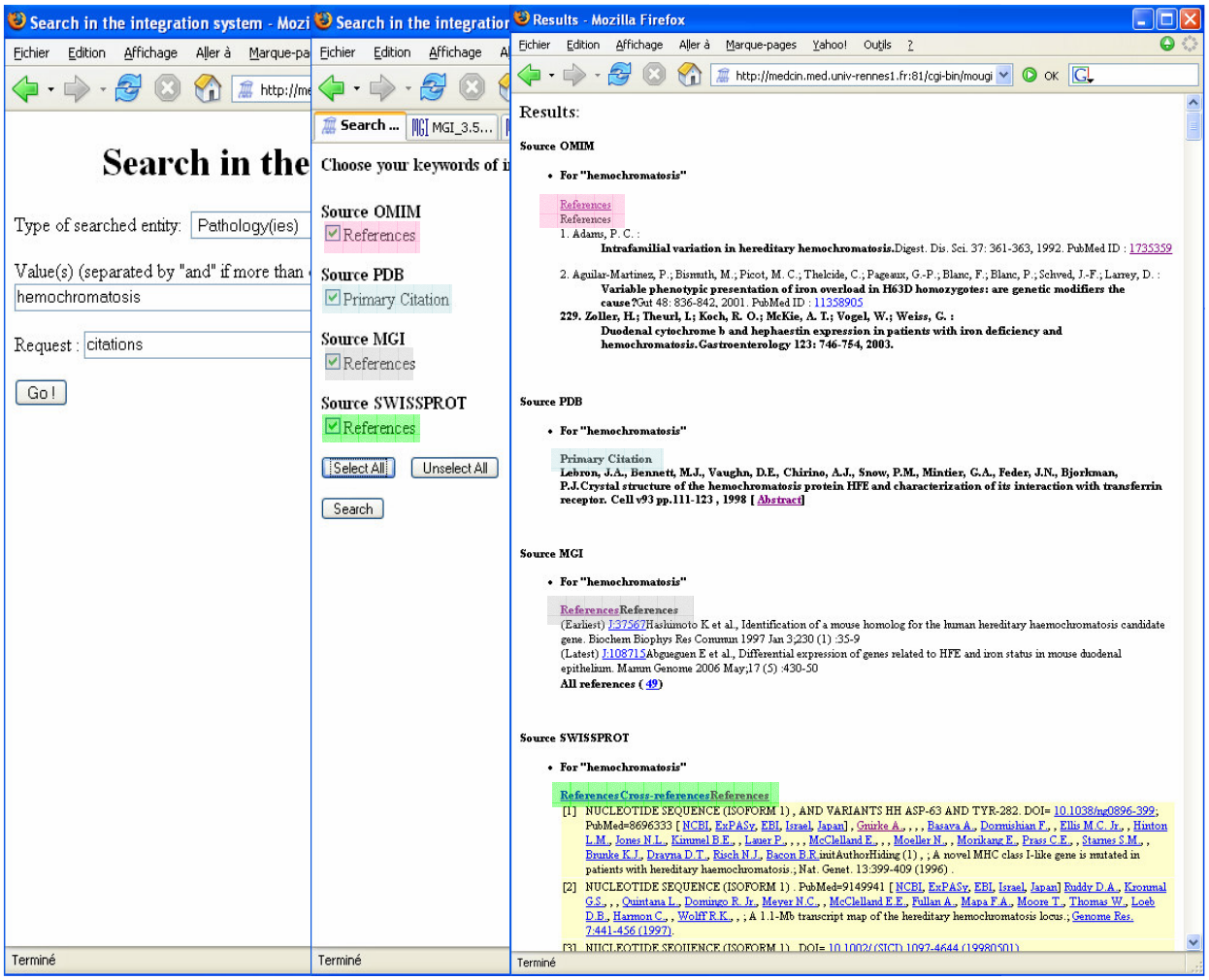


FIG. 6.4 – **Exemple de requête de la forme : *Recherche de citations concernant la maladie nommée hémochromatose dans les sources intégrées.*** La page de gauche correspond à l'interface d'interrogation où les utilisateurs remplissent les différents champs pour poser leur requête. La seconde copie d'écran montre les différents éléments de données que notre système identifie comme pertinents vis à vis de la requête posée ainsi que la source dont ils proviennent. Enfin, la troisième page présente les résultats obtenus dans les sources pour lesquelles certains éléments de données ont été choisis (cochés). Ainsi, on obtient différents types de publications issues de quatre sources distinctes : OMIM, PDB, MGI et Swiss-Prot.

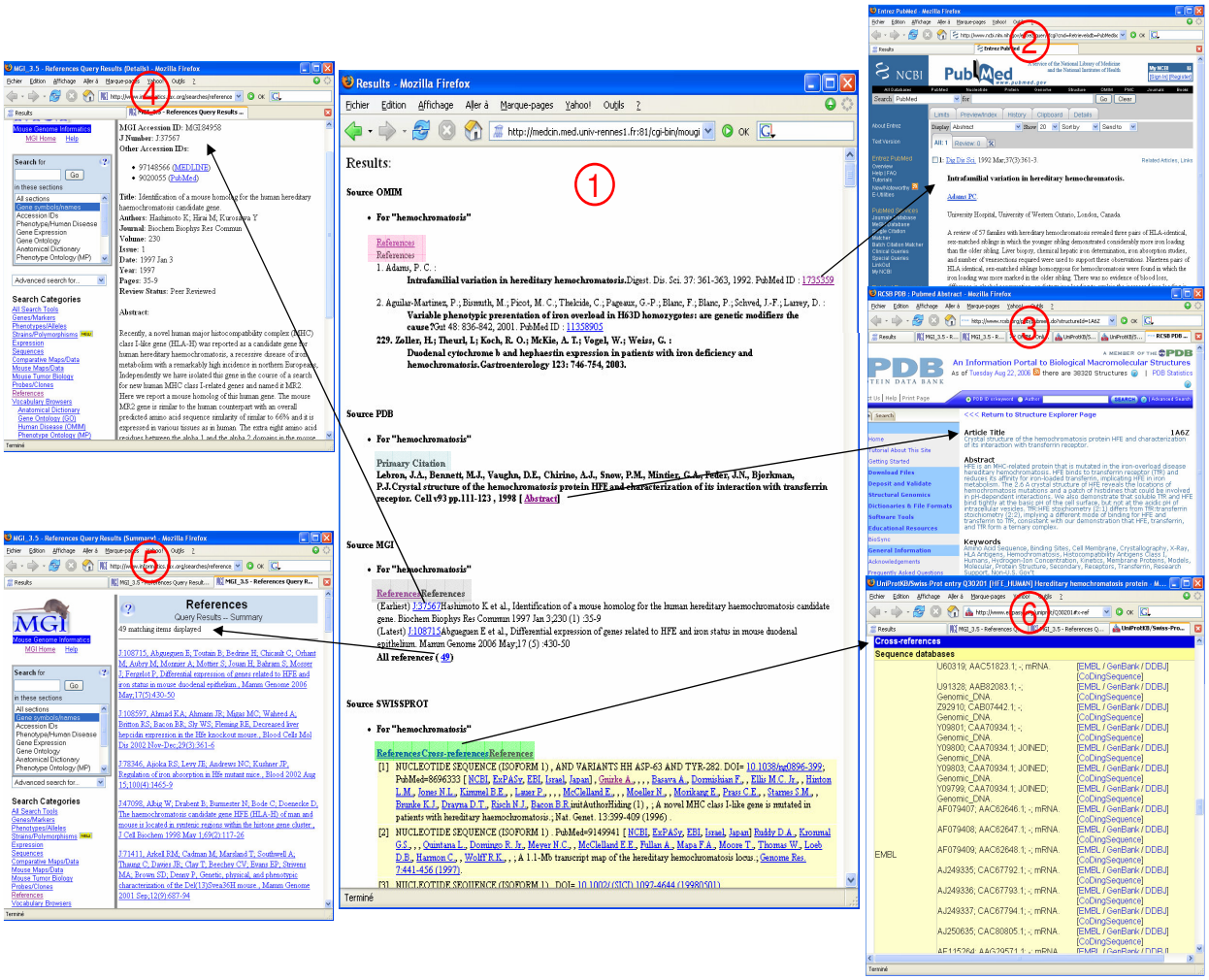


FIG. 6.5 – **Références croisées et liens internes proposés par notre système en réponse à la requête *Recherche de citations concernant la maladie nommée hémochromatose.*** La page 1 correspond à la page résultat de la figure précédente. Dans la source OMIM, un premier lien (interne) permet de visualiser les publications recensées pour cette maladie dans la page résultat d'OMIM. Cela correspond aux 229 citations que notre système récupère (seules trois d'entre elles sont visibles sur notre figure parce que nous avons choisi de tronquer le détail des 226 autres afin de disposer sur une même figure de l'ensemble des résultats issus des différentes sources). Une référence croisée vers l'entrée PubMed correspondante est également associée à chaque citation OMIM, la page 2 en donne un exemple pour la première. La page 3 illustre une référence interne à PDB où l'on peut accéder à l'abstract de la citation correspondante. MGD propose quatre liens internes ; un vers une page générale d'où on peut interroger MGD sur des publications particulières (lien « References »), deux autres permettant de visualiser l'abstract et d'autres informations concernant les articles les plus récent et ancien concernant cette maladie chez la souris (page 4) et enfin un lien vers l'ensemble des articles recensés comme pertinents pour cette interrogation (page 5). Finalement, la page 6 est obtenue en suivant le lien « Cross-references » proposé par Swiss-Prot, un autre lien interne (« References ») pointe sur la page Swiss-Prot résultat pour l'hémochromatose, information récupérée et fournie en bas de la page 1 (et en dessous) par notre système.

- `Mouse, Rat` extrait de Entrez Gene : cet ED est récupéré de manière erronée parce qu'un synonyme du concept associé contient le mot « Laboratory ». On choisit donc de ne pas récupérer les valeurs qui lui sont associées en ne le sélectionnant pas parmi les EDs proposés aux utilisateurs ;
- les EDs `Assay Type`, `Assays`, `Northern Blot`* et `Results` extraits de MGD : ces EDs contiennent des données obtenues par des procédures appliquées sur le gène donné (ici « COL1A2 ») chez la souris. En particulier, le nombre de résultats obtenus pour les « Northern Blot »*, une technique de quantification, est indiqué (et un lien vers leur détail). Sont également fournis le nombre de et un lien vers l'ensemble des résultats des différentes procédures (incluant RT-PCR* et RNA in situ), donnant ainsi accès aux amorces et sondes déjà utilisées et validées ;
- `Antibodies and Assays for symb` issu de GeneCards : il donne des références croisées vers deux sources, Invitrogen[1] qui concerne des procédures de laboratoire et Abcam® donnant des informations sur les anticorps d'intérêt potentiel pour des études protéomiques ;
- `Get Region on 2D PAGE` extrait de Swiss-Prot : cet ED donne un accès aux résultats obtenus par électrophorèse bidimensionnelle au travers d'un gel de polyacrylamide sur la protéine associée au gène considéré. C'est une référence croisée vers la source SWISS-2DPAGE.

6.1.4.3 Instances

Le troisième exemple montre comment notre système exploite, au moment d'exécuter les requêtes posées par les utilisateurs, les informations que nous avons obtenues grâce à l'utilisation des instances. Pour cela, la requête suivante est posée : *chercher les séquences associées au gène « BRCA1 »* (Figure 6.7 page 161). Les deux EDs `Primer 1` et `Primer 2` ayant été typés comme *Sequence*, ils sont ajoutés à la liste des EDs possiblement intéressants pour répondre à la requête. C'est grâce au typage des EDs effectué au niveau *instances* que ces deux EDs peuvent être utilisés ici. Le nombre d'EDs identifiés comme pertinents est assez important car en plus des informations sur les séquences, les EDs contenant le mot « gene » sont listés. En fait, parmi les propriétés du concept ***Genes (C0017337)*** a été ajouté l'hypernyme direct ***sequence#n#1*** du synset ***gene#n#1*** lui-même associé au concept UMLS ***Genes***. Ce dernier étant un descendant du terme recherché, il fait partie des termes d'intérêt et les EDs lui étant associés également. Ainsi, les utilisateurs disposent de nombreux EDs candidats et parmi ceux concernant les séquences, différents types de valeurs peuvent être obtenues :

- des séquences nucléiques au travers de l'ED `Sequence`, issu de HPRD ;
- des séquences protéiques : des informations concernant la séquence de la protéine associée au gène BRCA1 peuvent être récoltées, avec l'ED `Sequence Information`, extrait de Swiss-Prot (où on aura notamment le poids moléculaire et la longueur de la séquence), et à nouveau `Sequence`, ED issu de HPRD (qui donne par exemple le nombre d'acides aminés constituant la séquence) ;

[1] http ://www.invitrogen.com/

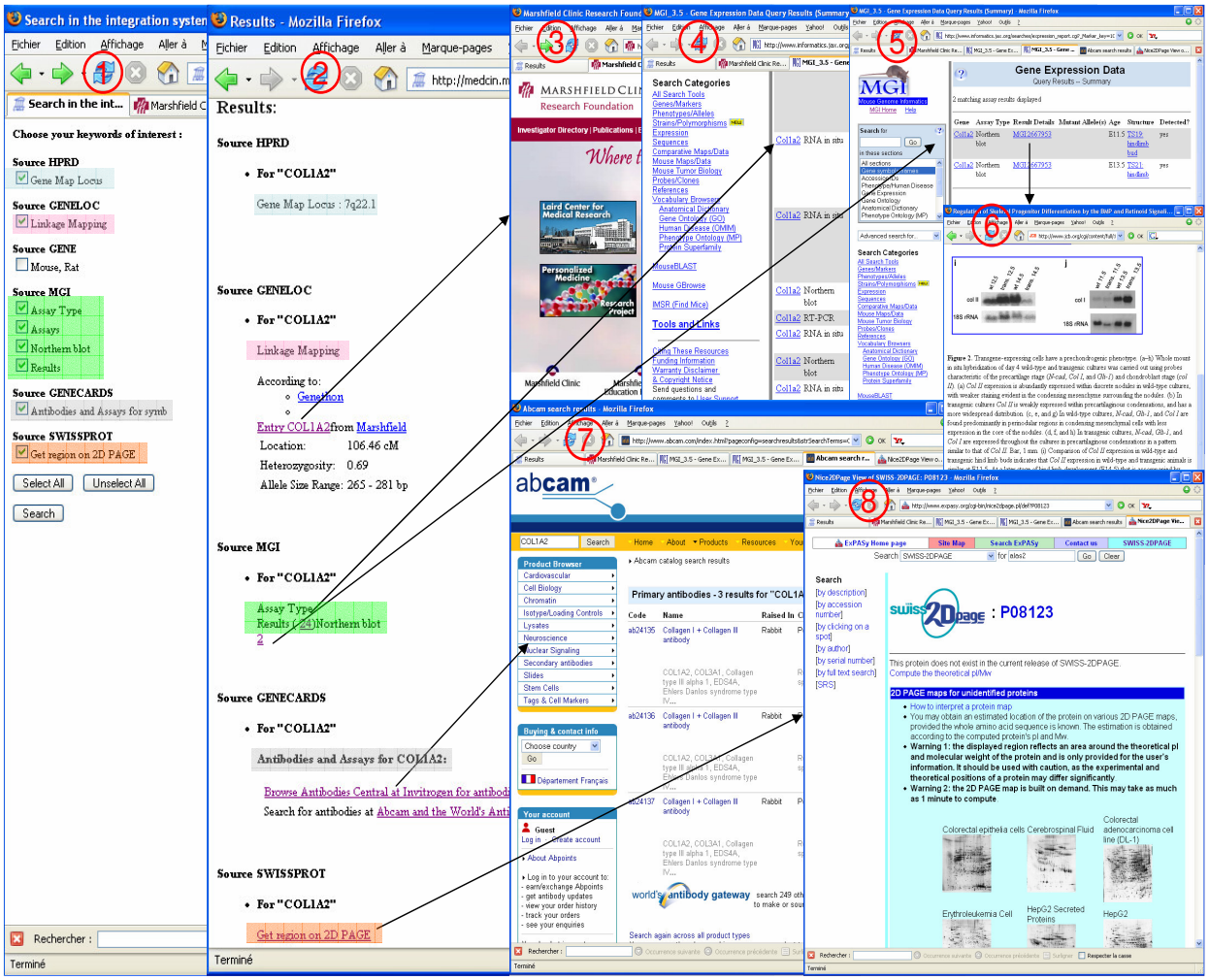


FIG. 6.6 – **Exemple de requête de la forme : *Recherche des résultats obtenus par des procédures de laboratoire pour le gène « collagen, type I, alpha 2 »*.** La page 1 correspond aux EDs identifiés par notre système comme potentiellement pertinents pour répondre à la requête. La page 2 présente les résultats récoltés dans les cinq sources HPRD, GeneLoc, MGD, GeneCards et Swiss-Prot. La page 3 est la page d'accueil du laboratoire Marshfield d'où la source GeneLoc a obtenu ses informations (référence croisée). Les pages 4 et 5 présentent, respectivement, les résultats issus de MGD pour l'ensemble des procédures effectuées et plus spécifiquement pour les « Northern Blot »*. À partir de ces pages, il est possible d'accéder aux publications d'où sont issues ces données et ainsi de visualiser les figures expérimentales correspondantes (page 6). La page 7 liste les anticorps trouvés par Abcam et la page 8 indique qu'il n'y a pas d'informations concernant la protéine associée au gène COL1A2 dans SWISS-2DPAGE.

- des séquences d'ADN complémentaire via l'ED `cDNA sequences`, issu de GeneCards. Cet ED permet d'accéder aux séquences correspondantes dans GenBank (en suivant une référence croisée) ;
- des séquences d'amorces (utiles pour la RT-PCR*) grâce aux EDs `Primer 1` et `Primer 2`, extraits de GeneLoc ;
- la structure de la séquence protéique correspondante via l'ED `Sequence Details`, issu de PDB ;
- une carte des séquences nucléique et protéique murines au travers de l'ED `Sequence Map`, extrait de MGD.

6.2 Évolution du système

6.2.1 Ajout d'une nouvelle source

Pour illustrer les différentes étapes nécessaires lors de l'ajout d'une nouvelle source et leur automatisation, nous avons réalisé l'intégration de Aceview [Thierry-Mieg 06] au sein de notre système. Cette source fournit, pour l'ensemble des génomes humain et de l'espèce *C. Elegans*, des annotations mises à jour de manière régulière au sujet de la structure des introns et exons des gènes et de leurs variants transcrits alternatifs. Son contenu est généré semi-automatiquement d'après les sources GenBank, dbEST[2] et RefSeq[3] dont les données sont résumées et pour lesquelles la redondance est éliminée avant d'être ajoutée à Aceview.

La première étape est manuelle, elle nécessite d'identifier les méta-données concernant la nouvelle source, c'est-à-dire son nom, le type d'entité pour laquelle on peut rechercher des informations et l'URL à partir de laquelle on peut l'interroger. Du descriptif précédent, on déduit immédiatement les deux premières méta-données, le nom est *Aceview* et l'entité centrale est le *gène*. Pour l'URL d'interrogation, il faut consulter le site Web de la source et y identifier l'URL que l'on peut utiliser pour poser des requêtes dynamiquement à Aceview. Une fois que l'ensemble des méta-données est connu, une interface (à l'adresse http ://medcin.med.univ-rennes1.fr :81/cgi-bin/mougin/These/nvelle_source.pl) permet de saisir ces informations indispensables.

L'étape suivante consiste à extraire les EDs de la nouvelle source ainsi que ces références croisées. Ce processus est fait automatiquement au moyen de l'algorithme décrit dans la partie 4.2.3.2.2 page 104. La liste des 57 EDs récupérés dans AceView est donnée en annexe D (page 218). Pour les références croisées identifiées par notre algorithme, dix d'entre elles existent déjà dans notre base de données. En revanche, quatre références sont nouvelles et nécessitent d'être validées ou supprimées (détail de la validation donné également en annexe D). Pour cela, une interface est disponible, il y est possible de modifier soit le nom de la source référencée (qui est étiquetée « new ref » lorsqu'il n'a pas été possible d'en identifier le nom), soit l'URL (cela peut être utile par exemple si elle n'est pas sous sa forme de base). D'autre part, il est également possible de supprimer une référence identifiée mais paraissant erronée ou non pertinente. Parmi les quatre références croisées dont le nom n'est pas connu, nous avons gardé trois d'entre elles

[2]http ://www.ncbi.nlm.nih.gov/dbEST/
[3]http ://www.ncbi.nlm.nih.gov/RefSeq/

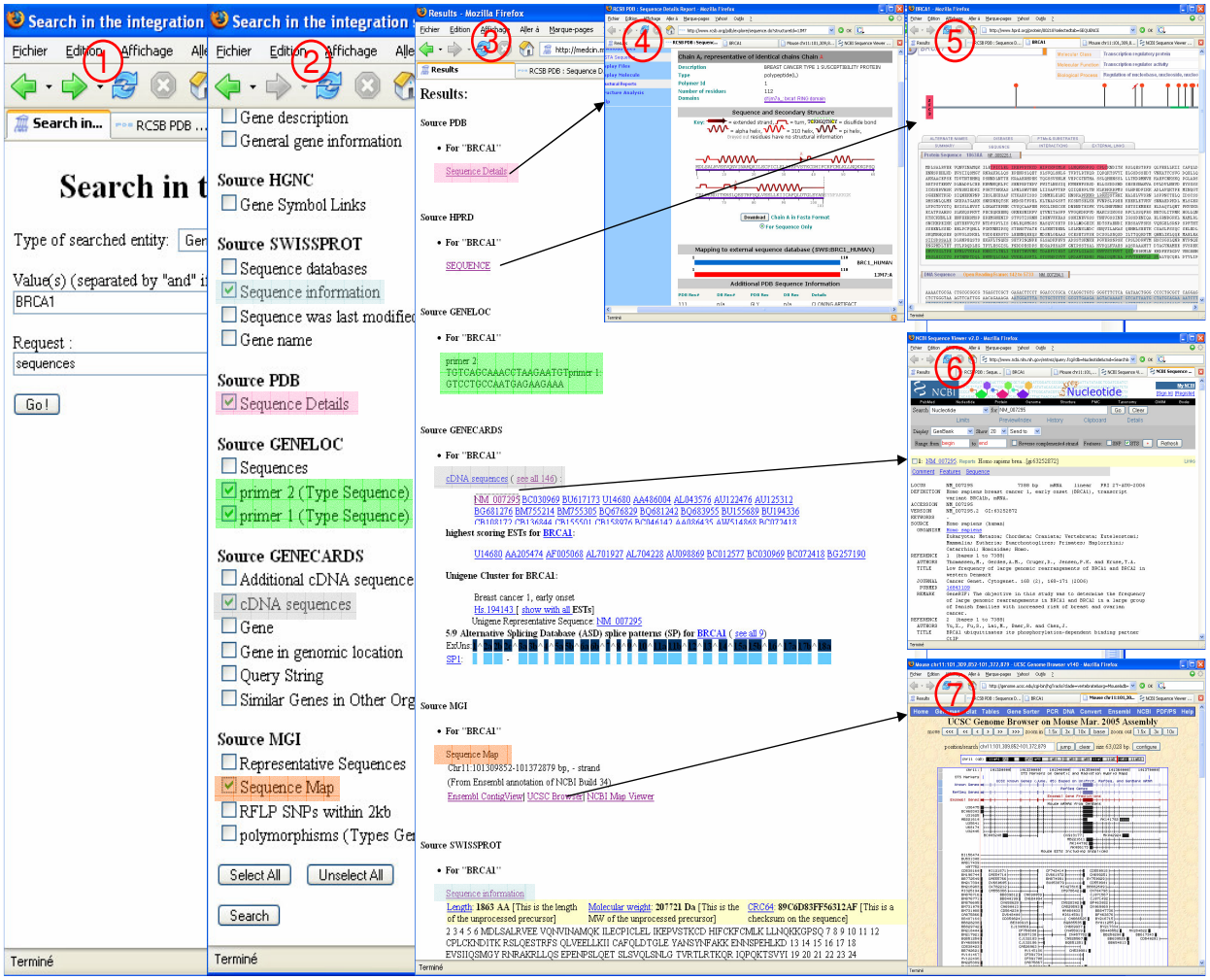


FIG. 6.7 – **Exemple de requête de la forme : *Recherche des séquences reliées au gène ayant pour symbole « BRCA1 »*.** La page de gauche (numéro 1) montre l'interface d'interrogation que les utilisateurs remplissent. La page 2 recense les différents éléments de données mis en correspondance avec le terme de la requête. La troisième page liste les résultats recueillis dans chaque source par notre système. PDB fournit un lien interne donnant des informations sur la structure de la séquence protéique associée au gène BRCA1 (page 4). HPRD offre aussi un lien interne dont la page associée (numéro 5) présente les séquences protéique et nucléique de BRCA1. Dans GeneLoc, on récupère les séquences de deux amorces tandis que dans GeneCards, on récolte des informations concernant des séquences d'ADN complémentaire, visualisables au travers d'une référence croisée vers GenBank (page 6). MGD indique le chromosome et les paires de base sur lesquels se trouve le gène chez la souris et fournit une carte des séquences nucléique et protéique (page 7). Enfin, Swiss-Prot liste diverses données concernant la séquence protéique associée à BRCA1 (longueur en acides aminés - AA, poids moléculaire en Dalton - Da et le contrôle de redondance cyclique sur 64 bits).

que nous avons renommées (et avons modifié l'URL de l'une d'elle). Une référence qui était un lien vers la description d'un algorithme utilisé par Aceview a été supprimée car elle n'est pas pertinente.

La troisième étape est manuelle. Elle nécessite d'écrire le programme permettant de récupérer les valeurs associées aux EDs extraits de la nouvelle source. Même si ce travail est pénible, il a l'avantage d'avoir une double utilité : il permet tout d'abord de récupérer des valeurs (entre 0 et 100) pour chaque ED auquel sera attribué un type (TYPE SÉMANTIQUE T, *Identifier*, *Integer*, *Sequence* ou *String*) et, dans un deuxième temps, c'est le programme qu'utilisera l'adaptateur gérant l'accès à Aceview au moment où les utilisateurs vont interroger le système. Les EDs que nous avons extraits ont tous été typés comme *String*. Cela n'est pas vraiment étonnant vu que Aceview est une source de type textuelle (comme OMIM) où on ne trouve pas d'informations structurées ou standardisées. À ce stade, le schéma de la nouvelle source est créé de manière automatique, puisque l'on dispose de toutes les informations nécessaires pour cela, c'est-à-dire ses méta-données, ses références croisées, l'ensemble des EDs que nous avons extraits ainsi que le type de chacun d'eux.

L'étape suivante détermine, de manière semi-automatique, les correspondances entre les EDs extraits de la nouvelle source et les éléments du schéma global. En pratique, cela consiste tout d'abord à mettre en correspondance les EDs avec les concepts de l'UMLS d'une part et les synsets de WordNet de l'autre. Ensuite, les concepts et synsets sont appariés pour voir s'il est possible de compléter, avec WordNet, la mise en correspondance des EDs avec les concepts de l'UMLS directement. Pour Aceview, les résultats sont les suivants :

- parmi les 22 correspondances uniques, 16 sont validées automatiquement. Un exemple est la correspondance trouvée entre l'ED `Bibliography` et le concept UMLS ***Bibliography (C0021920)*** qui est validée par le synset ***bibliography#n#1*** grâce à la similitude de leur définition respective. Les 6 correspondances uniques restantes ont été validées manuellement ;
- sur 29 correspondances multiples, 7 sont désambiguïsées. Par exemple, l'association trouvée entre l'ED `Phenotype` et les deux concepts UMLS (identifiés par correspondance exacte) : ***Phenotype (C0031437)*** et ***Phenotype determination (C1285572)*** est désambiguïsée grâce au synset ***phenotype#n#1*** qui a une définition similaire au concept ***Phenotype (C0031437)***. C'est donc ce dernier qui est choisi par rapport à l'autre. Sur les 22 correspondances multiples restantes, 10 ont été désambiguïsées manuellement (car certains EDs sont redondants) ;
- 6 nouvelles correspondances sont identifiées au travers de WordNet. Par exemple, l'ED `Functional annotation` n'est pas trouvé dans l'UMLS mais est par contre associé au synset ***annotation#n#1*** dont l'hypernyme direct est ***comment#n#1***. Celui-ci est mis en correspondance avec le concept ***Published Comment (C0282411)***. Ainsi, `Functional annotation` peut être mis en correspondance indirectement avec un concept de l'UMLS. Quatre des correspondances indirectes identifiées (synsets correspondants : ***annotation#n#1***, ***product#n#4***, ***sequence#n#1*** et ***transcript#n#2***) pour Aceview avaient déjà été trouvées pour les sources que nous avons intégrées au moment de la concep-

tion du système. Aucune validation manuelle de ces correspondances indirectes n'est nécessaire puisqu'elles l'avaient déjà été pour d'autres cas (ré-utilisation des correspondances existantes). En revanche, pour les deux EDs `Full Page` et `Sequencing Gap`, des synsets sont candidats pour identifier des concepts UMLS indirectement. Pour le premier, le synset ***full_page#n#1*** n'a pas de synonyme mais a par contre comme hypernyme direct ***page#n#1***. Le problème est que le concept UMLS associé ***PAGE (C1441680)*** est une procédure de laboratoire et constitue donc une correspondance erronée. Celle-ci n'est donc pas validée. Le deuxième cas concerne l'ED `Sequencing Gap`, le synset qui lui est associé est ***sequence#v#2*** qui n'a pas non plus de synonyme mais a pour hypernyme direct ***determine#v#1*** qui peut être mis en correspondance dans l'UMLS avec le concept ***Determined by (C0521095)***. Cette correspondance est correcte et est donc validée. Cela va entraîner l'ajout de ce concept (et ses ascendants) au schéma global, complété des propriétés WordNet du synset ***sequence#v#2***.

Parallèlement dans cette étape, on recherche également des correspondances éventuelles entre les EDs de la nouvelle source et ceux des sources déjà intégrées au système. En pratique, on compare les valeurs des EDs de la nouvelle source avec celles de chaque ED déjà existant. Si le coefficient de Jaccard entre les ensembles de valeurs de deux EDs est supérieur à 0,5, des nouvelles correspondances pourront être automatiquement ajoutées dans la base recensant les correspondances utilisées ensuite par le médiateur. Des validations et suppressions de correspondances peuvent aussi être effectuées. Si le coefficient est entre 0,2 et 0,5, les concepteurs doivent valider les actions proposées. Pour la source Aceview, étant donné le caractère textuel des informations qu'elle fournit, aucune correspondance avec les valeurs d'un ED d'une autre source n'a pu être identifiée.

La cinquième étape est entièrement automatique. Les correspondances ayant été validées (ou non), il reste à mettre à jour le schéma global en fonction des nouveaux éléments identifiés comme nécessaires à l'étape précédente pour représenter les EDs de la nouvelle source. Les nouveaux concepts permettant de représenter les EDs sont ajoutés dans le schéma global ainsi que les propriétés des synsets appariés avec ces concepts lorsqu'au moins un des critères de similarité est vérifié. Au total, 54 EDs (sur 57 au total) sont mis en correspondance avec des éléments du schéma global. Celui-ci est ainsi augmenté de 23 nouveaux concepts (ainsi que leurs ancêtres que nous ajoutons à notre schéma global), eux-mêmes enrichis par les propriétés de 8 nouveaux synsets. Le schéma global ainsi complété est disponible à l'adresse http ://www.med.univ-rennes1.fr/ mougin/onto/schema_global_with_wn_aceview_added.owl.

Le tableau 6.1 récapitule les différentes étapes (et sous-tâches) nécessaires à l'ajout d'une source au sein de notre système ainsi que le degré d'automatisation fourni pour les réaliser.

TAB. 6.1 – Étapes nécessaires à l'ajout d'une source. Certaines étapes sont divisées en sous-tâches et pour chacune d'entre elles, on indique si elle est automatique, manuelle ou semi-automatique. Enfin, lorsqu'une intervention humaine est nécessaire, nous précisons si une interface est mise à disposition pour assister le travail des concepteurs.

Étape	**Sous-tâche**	**Automatique / manuelle**	**Interface disponible**
Collecte des méta-données	-	Manuelle	Saisie des méta-données pour initier l'ajout de la source
Identification des éléments de données et références croisées	Extraction	Automatique	-
	Traitement des références croisées	Semi-automatique	Modification / suppression des nouvelles références
Écriture du programme pour interroger dynamiquement la nouvelle source	-	Manuelle	Aucune
Création du schéma de la nouvelle source	Typage des éléments de données	Automatique	-
	Création du fichier XML	Automatique	-
Mise en correspondance des éléments de données avec ceux du schéma global	Mise en correspondance directe des éléments de données dans l'UMLS	Automatique	-
	Mise en correspondance indirecte des éléments de données dans l'UMLS	Automatique	-

Étape	Sous-tâche	Automatique / manuelle	Interface disponible
	Appariement des concepts et synsets	Automatique	-
	Validation des correspondances non validées par l'approche indirecte	Manuelle	Proposition des correspondances aux concepteurs
Mise en correspondance des éléments de données au travers de leurs valeurs	Calcul du coefficient de Jaccard	Automatique	-
	Validation des correspondances	Semi-automatique	Proposition des correspondances aux concepteurs si le coefficient de Jaccard est compris entre 0,2 et 0,5
Intégration des nouveaux éléments au schéma global	-	Automatique	-

Pour clore cette partie, nous illustrons l'intérêt d'ajouter une nouvelle source à notre système au travers d'un exemple de requête. Avant d'intégrer Aceview, il n'était pas possible de récupérer directement des informations concernant les introns et exons d'un gène. Ceci est maintenant immédiat grâce à cette nouvelle source (Figure 6.8 page suivante).

6.2.2 Modification d'une source

Nous avons vu en détail les différentes étapes nécessaires à l'ajout d'une nouvelle source car c'est la tâche qui implique le plus de travail. Dans le cas d'une modification, certaines étapes ne sont pas nécessaires et il faut au préalable distinguer plusieurs types de mise à jour :

- l'URL d'interrogation de la source a changé. Pour ce cas de figure, il faut modifier cette information manuellement ;
- le format de sortie des résultats fournis par la source est modifié. Dans ce cas, le programme permettant de récupérer les valeurs associées aux EDs risque de ne plus fonctionner, ce qui empêche le système de pouvoir interroger efficacement cette source. C'est ce cas qui est le plus gênant car il nécessite de ré-écrire, ou tout du moins de modifier, le programme d'interrogation de la source, ce qui doit être fait manuellement ;
- les EDs contenus dans la source sont modifiés, certains supprimés, d'autres renommés ou encore ajoutés.

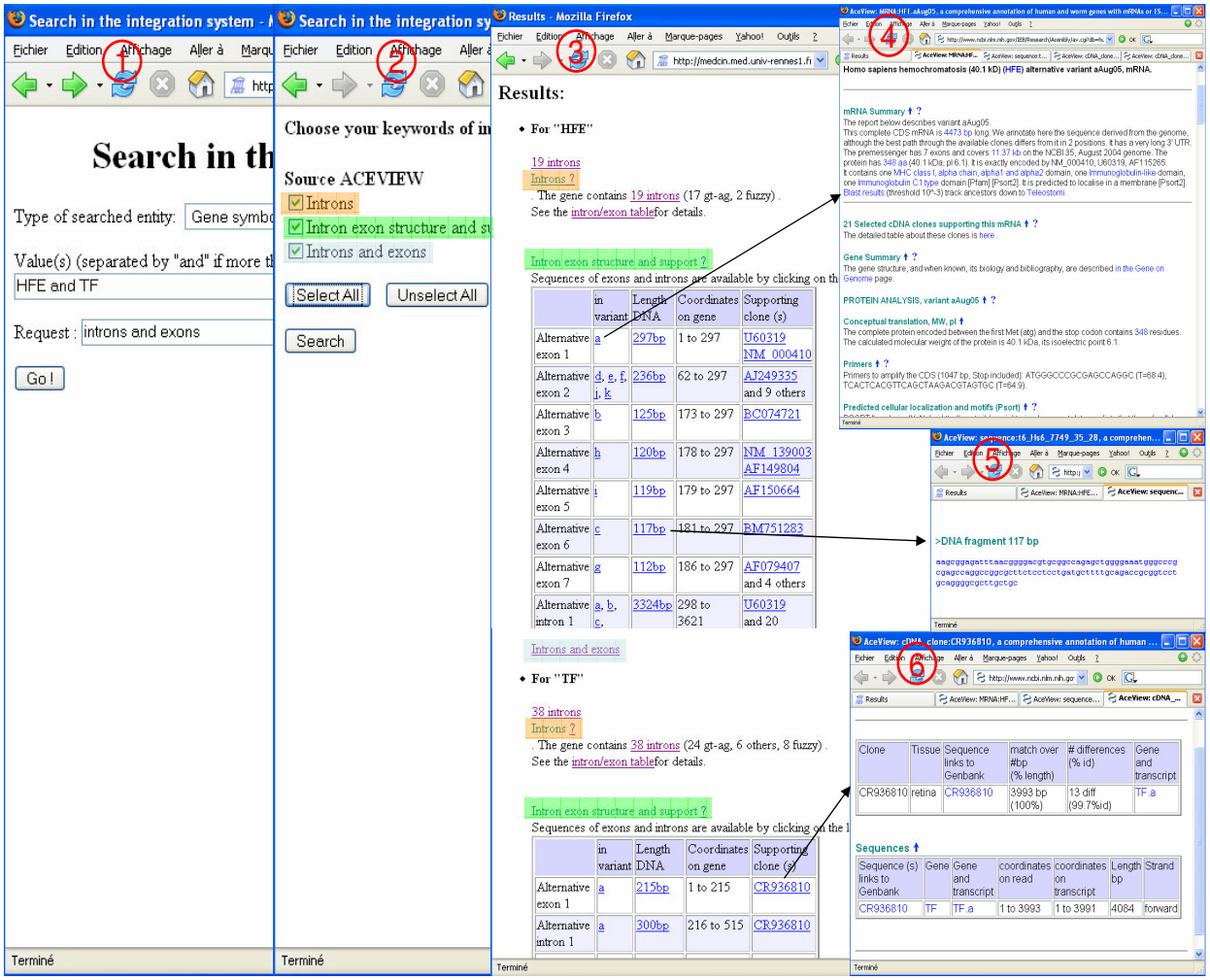


FIG. 6.8 – **Exemple de requête de la forme : *Recherche des introns et exons des gènes ayant pour symboles « HFE » et « TF »*.** Sur la page 1, on pose la requête dans l'interface utilisateur. La page 2 indique que seule la source Aceview contient des éléments de données correspondant aux mots de la requête. La page 3 donne les résultats fournis pour les trois éléments de données dont on souhaite récupérer les valeurs. Nous avons découpé une partie du tableau résultat pour le gène HFE afin de pouvoir montrer sur une figure unique les résultats obtenus pour « TF » en même temps. Pour ces éléments de données, les liens fournis par Aceview sont internes. Pour chaque intron et exon, on peut ainsi accéder au descriptif du(des) variant(s) où il se trouve (page 4), à la séquence du fragment d'ADN complémentaire lié à cet intron ou exon (page 5) ainsi qu'à la séquence d'ADN complémentaire complète représentant le transcrit alternatif (page 6).

C'est ce troisième cas de figure que nos approches permettent de faciliter. Certaines étapes sont similaires à celles qui sont nécessaires pour ajouter une source. En pratique, il faut commencer par exécuter le programme d'extraction des EDs dans la source. Comme déjà souligné, ce programme est ainsi utile non seulement au moment de l'intégration d'une nouvelle source mais aussi pour gérer plus aisément son évolution au sein de notre système. Grâce à la liste des EDs extraits et celle dont on disposait précédemment, il est possible d'identifier les différences. À ce stade, deux cas se présentent : certains EDs ont été supprimés et d'autres ont été ajoutés. Dans le premier cas, les correspondances dans lesquelles ces EDs sont impliqués sont simplement supprimées. Dans le cas où des nouveaux EDs ont été ajoutés, les étapes nécessaires sont les suivantes :

- on recherche des correspondances entre les EDs et les éléments du schéma global et parallèlement, on récupère les valeurs associées à chaque nouvel ED. Ainsi, on peut typer chaque ED et re-générer le schéma de la source en fonction des modifications identifiées ;
- ensuite, si l'ensemble des EDs en question peuvent être mis en correspondance avec des concepts du schéma global alors l'étape suivante consiste uniquement à ajouter les nouvelles correspondances identifiées (y compris celles considérant uniquement les valeurs des EDs) et le processus de mise à jour est terminé ;
- par contre, si certains EDs ne sont associés à aucun élément du schéma global alors on reprend le même processus que pour l'ajout d'une nouvelle source à partir de l'étape 4, c'est-à-dire la mise en correspondance des EDs avec WordNet et l'UMLS puis leur appariement, etc.

6.3 Synthèse

Ce chapitre présente les composants permettant d'interroger le système. Dans le chapitre précédent, le schéma global et les schémas locaux avaient déjà été décrits. Ici, ce sont le médiateur et les adaptateurs qui ont été introduits. Puis la stratégie de requêtes a été détaillée et illustrée par trois exemples. Enfin, la gestion de l'évolution du système a été abordée afin de préciser les étapes nécessaires pour l'intégration d'une nouvelle source et pour la modification de sources déjà intégrées au système.

Notre système est proche de l'approche LAV dans le sens où les EDs des schémas locaux sont représentés au moyen de concepts du schéma global [Lenzerini 02]. Par exemple, l'ED `Gene Name` de la source HGMD est associé aux deux concepts ***Genes (C0017337)*** et ***Names (C0027365)*** et peut être représenté comme suit :

$$(\texttt{Gene Name})_{HGMD} \rightarrow (\boldsymbol{Genes}) \wedge (\boldsymbol{Names})$$

En revanche, notre système ne met en œuvre qu'une version simplifiée de l'approche LAV. En particulier, au niveau du traitement des requêtes, les ré-écritures que nous effectuons consistent uniquement à récupérer les valeurs associées aux EDs mis en correspondance avec des concepts de la requête. Il n'y a pas de combinaison complexe comme des jointures. Ceci nous évite d'être confrontés aux difficultés propres à l'approche LAV pour ré-écrire les requêtes (cf 2.2.3.3.1

page 54), au prix d'une expressivité plus limitée.

Enfin, il faut souligner que malgré la limite précédente, le système que nous proposons permet l'enrichissement automatique du schéma global, ce qui n'est généralement pas le cas dans les médiateurs suivant l'approche LAV. Habituellement, les éléments présents dans le schéma de la nouvelle source sont représentés à l'aide des éléments du schéma global, ne nécessitant donc aucun changement de ce dernier. Cependant, des éléments nouveaux introduits par la nouvelle source seront ignorés si on ne peut pas les représenter avec des éléments du schéma global, ce qui constitue une limite puisqu'ils sont potentiellement pertinents. Il est bien sûr possible de faire évoluer le schéma global dans l'approche LAV, mais cela entraîne un travail complexe de mise à jour des adaptateurs.

Puisque le système que nous proposons repose sur un mécanisme simplifié de traitement des requêtes, il a été possible de réaliser automatiquement l'enrichissement du schéma global lorsque un élément présent dans le schéma d'une nouvelle source existe dans l'UMLS mais pas dans le schéma global. Le concept identifié dans l'UMLS (directement ou au travers de WN) est ainsi ajouté au schéma global et il est alors possible de représenter l'élément correspondant dans la nouvelle source.

Ainsi, notre approche s'inspire du modèle LAV puisque les schémas locaux sont définis par référence au schéma global. En revanche, nous n'avons mis en œuvre qu'un mécanisme simplifié de ré-écriture des requêtes. Cependant, cette simplification nous a permis de proposer une fonction originale d'enrichissement automatique du schéma global.

Chapitre 7

Discussion

Nous avons développé un système d'intégration basée sur la médiation pour le domaine biomédical. D'autres systèmes de ce type existent et nous positionnons ici notre travail par rapport à ceux-ci. Pour cela, nous détaillons les aspects nouveaux proposés par nos méthodes, leurs limites et les perspectives qui pourraient permettre de répondre à celles-ci.

7.1 Comparaison avec les systèmes existants

Nous avons cherché à automatiser le maximum de tâches pouvant faciliter la conception de notre système d'intégration. Les spécificités de notre système sont les suivantes : l'acquisition automatique des schémas locaux, le développement de méthodes de mise en correspondance situées au niveau *instances* afin de compléter les techniques situées au niveau *schéma* mises en œuvre pour associer les schémas locaux et le schéma global, et enfin le développement d'un schéma global à partir d'une ressource terminologique existante. Par ailleurs, nous avons montré dans la partie précédente en quoi ces différents aspects permettent de gérer plus facilement l'évolution de notre système.

Ce sont principalement les deux premières spécificités qui rendent notre système original par rapport aux systèmes d'intégration existants. En effet, nous avons vu qu'aucun d'entre eux ne permet d'acquérir automatiquement les schémas locaux. L'approche que nous proposons a été déjà comparée à d'autres travaux développés en informatique dans la section 4.2.3.2.1 page 103.

Pour le deuxième aspect, nous avons vu que, parmi les systèmes d'intégration existants, seul le système BACIIS [Ben Miled 05] a mis en œuvre une méthode située au niveau *instances* exploitant les données pour inférer des informations utiles au niveau *schéma*. Elle permet de découvrir les attributs dans des pages Web résultats fournies sur le site Web des sources biomédicales intégrées. Cependant, les concepteurs de BACIIS n'ont pas cherché à utiliser les valeurs associées aux attributs alors qu'elles permettent parfois de préciser ces attributs. Si un nouvel attribut `Name` doit être intégré et que celui-ci existe dans l'ontologie de BACIIS, ils seront alors mis en correspondance sans avoir vérifié au préalable si les valeurs qui lui sont associées correspondent bien au nom du même type de données (c'est-à-dire qu'il y aura, associés à un même concept ***Name***, un attribut `Name` issu d'une source S1 qui correspondra à un nom de gène et un autre attribut `Name` d'une source S2 qui concernera en réalité des noms de protéines). Ainsi, lorsque la sémantique des attributs est ignorée, cela risque de poser des problèmes d'incohérences. Les concepteurs des systèmes SEMEDA [Köhler 03] et INDUS [Reinoso-Castillo 03] ont souligné l'intérêt d'exploiter les valeurs associées aux attributs dans ce cadre mais n'ont pas implémenté d'approches le réalisant. Notre système pallie ce manque grâce aux méthodes que nous avons proposées au niveau *instances*. Celles-ci permettent en effet de préciser et de typer les éléments de données extraits des sources intégrées, pouvant même détecter des mauvaises correspondances identifiées avec des méthodes terminologiques (appliquées au niveau *schéma*).

Par rapport aux techniques exploitant le niveau *instances* qui ont été développées en informatique, les approches que nous avons mises en œuvre ressemblent à celles développées par Xu et Embley [Xu 03] (cf 2.3.4 page 81). L'inconvénient de ce système est qu'il nécessite la description d'une ontologie du domaine pour décrire le contenu des sources dont les concepteurs veulent aligner les schémas. Mais nous avons pu développer des méthodes similaires à celles qu'ils

proposent au niveau *instances* en remplaçant l'ontologie de domaine par l'UMLS. En effet, ce système terminologique dispose non seulement de concepts de haut niveau mais aussi de concepts dont la granularité est tellement fine que l'on peut les assimiler à des instances. Les méthodes structurelles qu'ils ont développées portant sur les contraintes définies sur les attributs ne sont pas applicables dans notre cas puisque nous ne disposons pas d'informations de ce type dans nos schémas locaux. Enfin, les techniques terminologiques exploitant WordNet sont proches des nôtres. Cependant, nous complétons celles-ci par l'exploitation des définitions pour déterminer la similarité entre deux éléments, ce qui est plus significatif sémantiquement.

Concernant l'utilisation d'une ressource déjà existante pour décrire le schéma global de notre système, notre approche peut être comparée à ce qui est fait dans le système d'intégration TAMBIS [Stevens 00]. En effet, les concepteurs utilisent l'ontologie TAO qu'ils ont créée à partir d'une ontologie pré-existante qu'ils ont adaptée afin qu'elle puisse représenter l'ensemble des éléments constituant les schémas des sources intégrées. Nous avons également choisi de ré-utiliser une ressource déjà existante pour nous affranchir du développement lourd et pénible d'un schéma global en partant de rien. L'ontologie à l'origine de TAO ayant été mise en œuvre par les mêmes concepteurs que TAMBIS, il est très probable que ses caractéristiques et son contenu étaient d'avance bien adaptés pour que l'ontologie soit aisément ré-utilisable dans TAMBIS. Dans notre cas, l'UMLS est un système terminologique entièrement indépendant de notre application, ce qui a nécessité un travail complexe pour l'adapter à nos besoins. Nous avons notamment considéré l'exploitation de deux approches distinctes pour créer un graphe orienté sans cycle à partir du Metathesaurus et ainsi concevoir notre schéma global. Nous l'avons de plus enrichi avec une autre ressource terminologique pour augmenter ses capacités. Par ailleurs, nous n'avons pas utilisé TAO comme schéma global pour différentes raisons : elle ne dispose ni d'informations terminologiques (synonymes et définitions) associées aux concepts ni d'instances, qui sont nécessaires pour nos méthodes. De plus, elle ne représente pas les connaissances médicales et recouvre moins largement le domaine biologique que l'UMLS.

Dans la suite de la discussion, nous soulignons ce qu'apportent nos méthodes de mise en correspondance et l'intérêt d'utiliser l'UMLS et WordNet pour décrire notre schéma global. Puis nous montrons en quoi la représentation de ce dernier de manière formelle permettrait de compléter notre travail. Ensuite, nous proposons deux perspectives possibles pour améliorer notre processus de requêtes avant de considérer les possibles généralisation et ré-utilisation de notre travail pour d'autres domaines ou applications.

7.2 Méthodes exploitant les niveaux *schéma* et *instances*

Pour faciliter la conception et gérer l'évolution de notre système, nous avons développé des méthodes situées aux niveaux *schéma* et *instances* afin de mettre correspondance les EDs extraits des sources dans le schéma global. Nous discutons des apports, limites et perspectives de nos approches.

7.2.1 Méthodes de mise en correspondance au niveau *schéma*

7.2.1.1 Apports

Tout d'abord, **nos méthodes permettent de traiter des correspondances de tout type de cardinalités**. La plupart des travaux existants se focalisent principalement sur les correspondances de cardinalité 1-1 ([Rahm 01]), ce qui constitue une limite. Avec nos approches, nous résolvons des correspondances de cardinalité 1-1, 1-n et 1-0 entre les EDs et les concepts de l'UMLS en utilisant WordNet comme ressource externe.

Pour mettre en correspondance les EDs dans l'UMLS, nous avons utilisé des méthodes **terminologiques** basées sur les outils lexicaux fournis par l'UMLS. Des traitements linguistiques, comme la tokenisation, la lemmatisation et l'utilisation d'une ressource supplémentaire (le Specialist Lexicon) sont appliquées aux EDs [McCray 94]. Des correspondances de cardinalités 1-1, 1-n et 1-0 ont été obtenues. Ces résultats ne sont pas satisfaisants dans les deux derniers cas et l'utilisation d'une ressource externe (WordNet) a permis de répondre en partie à ce problème.

Une fois les EDs associés à des synsets, ces derniers sont mis en correspondance dans l'UMLS. Pour comparer les différentes paires de (concept, synset) obtenues ainsi, nous avons mis en œuvre des approches terminologiques et structurelles. La similarité de leur définition, le nombre de synonymes communs et le nombre d'ancêtres communs sont calculés. Grâce à ces critères, il est possible de valider les correspondances de cardinalité 1-1 (si au moins un des critères est vérifié) et de désambiguïser les cardinalités 1-n si une paire (concept, synset) est déterminée comme étant meilleure que les autres. Dans ces deux cas, les mises en correspondances sont de plus validées automatiquement, limitant ainsi le travail des concepteurs. Les correspondances de type 1-0 sont améliorées quand un synset ayant été mis en correspondance avec un ED, non trouvé directement dans l'UMLS, a un synonyme ou un hypernyme direct qui existent dans l'UMLS.

7.2.1.2 Limites et perspectives

Notre système permet donc de valider des correspondances si au moins un des trois critères que nous avons définis est vérifié. Cependant, si la similarité entre un concept et un synset est très basse, il peut être erroné de considérer cette condition comme suffisante pour garantir qu'une correspondance est correcte. Il serait nécessaire de compléter notre travail avec des **mesures de similarité robustes en fixant un seuil** en dessous duquel les correspondances ne pourraient être acceptées, comme cela est proposé dans [Kefi 06]. De plus, après avoir appliqué nos méthodes, il reste malgré tout plus de la moitié des correspondances entre EDs et concepts UMLS à valider ou à supprimer par les concepteurs du système. Cependant, de nombreux EDs sont communs d'une source à l'autre et on a pu constater au travers de l'intégration de la source Aceview (cf 6.2.1 page 160) que les correspondances existantes sont ré-utilisables. Cela permet de limiter le travail manuel imposé lors de l'évolution du système.

Les schémas dont nous disposons pour chaque source regroupent l'ensemble des EDs que nous avons extraits ainsi que leur type. Cela **ne permet pas d'utiliser des méthodes structurelles basées sur les contraintes ni des approches sémantiques** basées sur l'interprétation de ces EDs. En effet, sans définition, il n'est pas possible de décrire ces EDs de manière formelle pour ensuite raisonner sur ces derniers. On verra cependant dans la section 7.4 page 178

comment les concepts représentant ces EDs peuvent en revanche être exploités dans ce but. Une perspective possible est d'**intégrer d'autres méthodes structurelles basées sur les graphes**. En l'occurrence, l'utilisation des descendants ou encore des relations s'inscrit dans la recherche d'un contexte commun entre deux éléments. Cependant, comparer des ensembles de descendants issus d'une ressource terminologique spécifique du domaine biomédical avec ceux d'une ressource terminologique générale ne nous paraît pas adapté, étant donné que leur niveau de granularité est très différent (cf le nombre de descendants présents dans l'UMLS - 5.2.1.1 page 134). Par contre, l'utilisation des relations est potentiellement prometteur, comme montré dans [Maedche 02] pour comparer des ontologies. L'UMLS contient différents types de relations dans le Metathesaurus, certaines d'entre elles sont même définies de manière formelle (issues de SNOMED-CT par exemple - [Schulz 05]). WordNet contient également des relations pouvant être exploitées pour être mises en correspondance avec certaines relations de l'UMLS. Par exemple, considérons la relation de composition dans WordNet (nommée meronym) qui équivaut à la relation de type *part_of* dans l'UMLS. Nos méthodes existantes établissent que le concept ***Chromosome (C0008633)*** et le synset ***chromosome#n#1*** peuvent être associés (par similarité de leur définition). Or il existe une relation issue de SNOMED-CT qui est typée *part_of* entre ***Chromosome*** et ***Cell Nucleus (C0007610)*** et parallèlement une relation de composition entre ***chromosome#n#1*** et ***nucleus#n#1***. Si une première correspondance au niveau du terme (approche terminologique avec les outils lexicaux de l'UMLS) a pu être établie entre ***Cell Nucleus (C0007610)*** et ***nucleus#n#1*** (synonyme « cell nucleus »), alors il serait possible de la valider grâce à leur environnement commun (même composant). Cette approche pourrait ainsi augmenter le nombre de correspondances validées de manière automatique.

Enfin, il y a deux raisons pour lesquelles nous n'obtenons pas plus de correspondances entre les concepts et synsets. D'une part, **les synsets WordNet ne comportent pas beaucoup de synonymes** et d'autre part, **de nombreux concepts UMLS ne disposent pas de définition**. Cela limite les correspondances identifiables au travers des deux critères correspondants. Si ces ressources améliorent ces aspects (intégration de synonymes supplémentaires aux synsets dans WordNet et ajout systématique de définitions aux concepts UMLS), nos méthodes donneront de meilleurs résultats.

7.2.2 Méthodes développées au niveau *instances*

7.2.2.1 Apports

Les correspondances identifiées au niveau *schéma* exploitent le nom des EDs. Le problème est que cela n'est pas suffisant pour certains EDs qui sont ambigus (par exemple, `Name`) ou mal nommés (par exemple, `Chromosome`[1]). Notre méthode visant à **typer les EDs en exploitant leurs données** permet de résoudre en partie ces problèmes. Pour cela, nous avons mis en correspondance les valeurs associées à chaque ED dans l'UMLS. Cela a permis, pour une trentaine d'EDs, de préciser que leur ensemble de valeurs concernaient un type connu d'informations.

[1]Comme nous l'avons déjà souligné, l'ED `Chromosome`, extrait de HGNC, indique en fait la localisation chromosomique d'un gène donné

De plus, cette approche **permet de typer des EDs qui n'ont pas forcément été mis en correspondance avec un élément du schéma global** en exploitant uniquement le niveau *schéma*. En effet, l'ED `From`, extrait de Swiss-Prot, n'est trouvé ni dans l'UMLS ni dans WordNet mais ses valeurs permettent de déterminer qu'il indique l'organisme pour lequel est définie une protéine donnée. Sur 94 valeurs non vides, 100% d'entre elles sont trouvées dans l'UMLS et les concepts correspondants sont tous catégorisés par le type sémantique ORGANISM. Cet ED est ainsi rattaché au schéma global au travers de son type.

L'autre approche implémentée au niveau *instances* permet de trouver des correspondances additionnelles entre les EDs et le schéma global, ainsi que de valider ou éliminer des correspondances identifiées au niveau *schéma*. Elle consiste à comparer les ensembles de valeurs des EDs deux à deux. Le coefficient de Jaccard [Van Rijsbergen 79] détermine un pourcentage de similarité et lorsqu'il est suffisamment haut, les trois cas de figure suivants se présentent. Si les concepts associés avec ces EDs sont les mêmes, **les correspondances sont validées**. Si les concepts associés avec ces EDs sont différents, deux possibilités se présentent. Si l'un des EDs est associé à un ou des concepts supplémentaires par rapport à l'autre ED, il est possible d'**ajouter une nouvelle correspondance** entre ce deuxième ED et le ou les concepts auxquels est associé le premier. Si les concepts sont incompatibles entre les deux EDs alors une incohérence existe pour l'un des EDs et la **correspondance mise en jeu est éliminée**.

Nous avons abordé la notion d'**intégration *verticale* qui correspond à l'agrégation de données sémantiquement similaires** [Sujansky 01]. La deuxième approche présentée permet de la gérer en partie dans notre système. Cet aspect est très important puisque la plupart des systèmes ne gèrent que l'**intégration *horizontale* qui réalise une composition de données complémentaires**. Cela pose problème car ils ne tiennent pas compte du possible recouvrement des sources. Au travers de correspondances identifiées entre des EDs de même contenu, notre méthode parvient à identifier des données identiques dans des sources distinctes. Ainsi, le médiateur peut filtrer aisément les données redondantes avant de les fournir en résultat aux utilisateurs du système d'intégration.

7.2.2.2 Limites et perspectives

Une limite de nos approches situées au niveau *instances* est qu'elles ne fournissent pour l'instant que peu de résultats. En effet, le typage des EDs n'est réussi que pour un peu plus de 11% des EDs extraits des sources et seules 22 paires d'EDs ont pu être mis en correspondance au travers des valeurs. Cela peut s'expliquer par le fait que les sources biomédicales fournissent souvent des données peu structurées ou standardisées. Il est nécessaire de compléter le typage des EDs dans le cas où les valeurs ne sont pas présentes dans l'UMLS. Pour cela, il faudrait **définir des patrons pouvant identifier un type complexe mais connu d'informations**, telles que des dates ou encore des références bibliographiques dont le format est généralement le même. Cela permettrait, par exemple, de typer plus précisément l'ED `Bibliography`, extrait de Entrez Gene, qui contient des informations bibliographiques comme les EDs `Primary Citation` ou `References`, dont les valeurs sont du même style. Un typage commun pour ces EDs permettraient de les proposer aux utilisateurs comme EDs candidats à une requête telle que

celle présentée précédemment : *Main citations of the hemochomatosis pathology*. Sans cela, l'ED `Bibliography` n'apparaît pas parmi les EDs candidats pouvant fournir des résultats intéressants aux utilisateurs, alors qu'il devrait être présent (cf 6.1.4.1 page 155). Une autre possibilité serait d'utiliser des techniques d'apprentissage, comme dans [Doan 03].

L'intégration verticale reste un point important à approfondir. Notre travail permet de la résoudre pour les cas où une correspondance au niveau des valeurs de deux EDs distincts a été identifiée. Cependant, pour les situations impliquant des EDs pour lesquels nous ne disposons pas de ce type d'informations, le médiateur n'effectue pour l'instant pas de contrôle quant à la redondance des données recueillies. Il y a donc probablement des informations identiques répétées parmi les résultats que fournit notre système. Une perspective importante est ainsi de développer des méthodes permettant au médiateur d'analyser en détail les résultats collectés dans les sources afin de les unifier et donc d'éliminer des possibles redondances.

En conclusion, nous avons illustré l'intérêt de pouvoir combiner des approches situées aux niveaux *schéma* et *instances* afin d'obtenir de meilleurs résultats, moins ambigus et plus cohérents pour les mises en correspondance entre les EDs extraits des sources intégrées et les éléments du schéma global. L'articulation de nos méthodes de différents types (terminologiques et structurelles) étant fixe, notre approche globale de mise en correspondance est qualifiée d'**hybride**.

7.3 Schéma global

Notre schéma global est constitué principalement de l'UMLS, et plus précisément d'une partie de cette ressource. Pour compléter sa couverture, nous avons enrichi certains de ses concepts par des propriétés de synsets WordNet mis en correspondance avec les mêmes EDs. Ces deux ressources terminologiques présentent des avantages et inconvénients et nous soulignons en quoi l'utilisation de l'une permet de compenser certaines des limites de l'autre.

7.3.1 L'UMLS

7.3.1.1 Intégration terminologique

Nous avons transformé l'UMLS sous la forme d'un graphe orienté sans cycle afin de pouvoir effectuer des parcours du graphe du Metathesaurus. Cela est indispensable lors du processus de requêtes qui réalise une expansion de celles-ci. Cependant, l'approche formelle utilisée pour cela ne résout pas tous les problèmes posés par l'intégration terminologique réalisée pour construire l'UMLS.

Parmi les avantages qu'elle a sur l'approche naïve, nous avons démontré que l'approche formelle filtre de nombreux descendants illégitimes. Toutefois, la **compatibilité sémantique des descendants restants est incomplète**. En effet, 59% des descendants obtenus par l'approche formelle pour les 26 584 concepts considérés en détail dans cette étude sont incompatibles au niveau des types sémantiques avec leur concept source respectif. Par exemple, les descendants

de ***Accidents (C0000924)***, catégorisé comme PHENOMENON OR PROCESS (groupe sémantique PHENOMENA), incluent le concept ***Accident prevention (C0000918)***, catégorisé comme THERAPEUTIC OR PREVENTIVE PROCEDURE (groupe sémantique PROCEDURES) et qui est donc sémantiquement incompatible avec le type sémantique du concept source. Cette relation non hiérarchique dans son vocabulaire d'origine n'est pas filtrée dans la mesure où elle n'introduit pas d'incohérence structurelle dans le graphe. En d'autres termes, notre méthode est efficace pour s'assurer que les relations hiérarchiques impliquées dans des cycles sont supprimées du Metathesaurus de manière cohérente. Cependant, pour que le Metathesaurus soit cohérent non seulement structurellement, mais aussi sémantiquement, une analyse sémantique de toutes les relations hiérarchiques serait nécessaire [McCray 02].

7.3.1.2 Choix de représentation

Nous avons représenté la relation de catégorisation liant les concepts aux types sémantiques comme une relation de type *is_a* dans notre schéma global. Cela a pour effet de confondre la relation hiérarchique liant les concepts entre eux et celle qui, habituellement, les assigne à un type sémantique de haut niveau. Ainsi, des redondances vont apparaître pour les concepts catégorisés par le même type sémantique que leurs parents. Par exemple, le concept ***Lengths (C1444754)*** a pour type sémantique QUANTITATIVE CONCEPT (T081) alors que l'un de ses pères, ***Dimensions (C0439534)*** est également catégorisé sous ce type sémantique (cf figure 5.5 page 138). Cette redondance peut cependant être aisément éliminée par un raisonneur (commande *Classify Taxonomy* sous Protégé, par exemple).

Cette représentation permet de résoudre une partie de l'incohérence sémantique introduite dans la partie précédente. En effet, le concept ***Variant (C0205419)***, par exemple, a pour type sémantique QUALITATIVE CONCEPT (T080), alors que son père ***Normality findings (C0456197)*** est catégorisé par FINDING (T033). ***Variant*** devrait avoir parmi ses types sémantiques au moins celui de son père ou un descendant de ce dernier, sinon une incohérence existerait. Avec notre représentation, par héritage de son père, ***Variant*** a également pour type sémantique FINDING, et n'est donc plus incohérent sémantiquement avec son concept père.

7.3.1.3 Connaissance supplémentaire

En plus de servir de schéma global, l'UMLS **pourrait permettre de découvrir des nouvelles connaissances au cours du processus de requêtes**. Nous avons vu que des informations situées au niveau *instances* se trouvent également dans l'UMLS, comme par exemple des noms de gènes donnés comme « HFE », « BRCA1 » et « TNXB ». Il est donc envisageable de considérer l'UMLS comme une source biomédicale au même titre que celles que nous avons intégrées. Comme les termes permettant de décrire la requête des utilisateurs sont associés à des concepts UMLS, il serait possible de proposer en résultat les descendants de ceux-ci ayant une association avec le gène ou la maladie intéressant les utilisateurs. Pour mettre en œuvre cette approche, il faudrait exploiter les relations inter-conceptuelles autres que hiérarchiques (par exemple, *can_be_qualified_by*) ou même les co-occurrences (termes présents dans un même article indexé dans PubMed) [Burgun 01c]. Par exemple, si l'on recherche des informa-

tions concernant les antigènes de la transferrine, le concept UMLS ***Antigens (C0003320)*** sera sélectionné et, sous celui-ci, existe un concept, ***HLA Antigens (C0019721)***, qui co-occure avec le concept **Transferrin (C0040679)**. On pourrait ainsi intégrer le concept ***HLA Antigens*** aux résultats de la requête.

7.3.2 WordNet

7.3.2.1 Couverture d'ordre général.

L'avantage principal apporté par WordNet est qu'il **complète la couverture des termes généraux**. Des correspondances indirectes ont ainsi été identifiées au travers de WordNet entre des EDs et l'UMLS. Il permet ainsi d'**enrichir le schéma global** en complétant les connaissances présentes dans l'UMLS. De plus, WordNet apporte des informations supplémentaires aux concepts dont les correspondances avec des EDs ont pu être validées et désambiguïsées grâce à des synsets. Quand une paire (concept, synset) a été déterminée comme étant la meilleure par rapport aux autres candidates ou qu'une nouvelle correspondance a pu être identifiée indirectement, la description du concept UMLS au sein du schéma global a été complétée avec les propriétés du synset associé. Cela revient donc à ajouter des synonymes, des hypernymes et une définition dans le schéma global, ce qui s'avère particulièrement utile au cours du processus de requêtes comme nous l'avons vu dans le chapitre précédent (cf 6.1.4.1 page 155). Par cette approche, les utilisateurs obtiennent effectivement des résultats plus complets.

7.3.2.2 Ambiguïté

Une limite résultant du caractère général de WordNet est son **ambiguïté**. En effet, cette ressource offre généralement plusieurs interprétations (et donc autant de synsets candidats) d'un même terme, comme le mot « Species » sur la figure 4.2 page 100. Pour en extraire une connaissance plus spécifique d'un certain domaine, différentes possibilités existent. Tout d'abord, on peut exploiter des définitions dans lesquelles il est possible d'identifier des termes issus d'un certain domaine et donc de décider de ne garder que les synsets correspondants. De plus, dans la description de WordNet, des noms de domaine ont été ajoutés à certains synsets, précisant le domaine dans lequel s'exprime le mot représenté. Cependant, les auteurs de MedicalWordNet ont souligné en quoi ces méthodes n'étaient pas suffisamment avancées [Fellbaum 06]. D'abord, les définitions peuvent se révéler problématiques car elles ont été générées par des linguistes qui ne sont pas spécialistes des domaines auxquels appartiennent les mots les constituant. En ce qui concerne les noms de domaine associés à des synsets, en dehors des relations hiérarchiques, il n'y a pas de relations entre des domaines qui sont pourtant connexes. C'est le cas notamment des domaines *Biology* et *Medicine*. D'autre part, les noms de domaine n'ont pas été assignés aux synsets de manière systématique. Parallèlement, de nombreux synsets dont le sens appartient clairement à un domaine prédéfini ne sont pas étiquetés comme tels. Par exemple, le synset ***organic_process#n#1*** (synonyme « biological process » et définition « a process occurring in living organisms ») touche de manière évidente au domaine *Biology* et ne contient pourtant pas ce nom de domaine dans sa définition. La méthode spécifique que nous proposons pour désambiguïser WordNet résout ce type de problème puisqu'elle vérifie la présence de mots du

domaine biomédical au sein des définitions et utilise plusieurs domaines concernant (de près ou de loin) le domaine biomédical.

7.3.2.3 Perspective

Il serait intéressant d'intégrer EuroWordNet à notre travail. C'est une base de données multilingue créée à partir de WordNets de nombreux pays européens (italien, français, espagnol, etc) [Vossen 98]. En plus d'être structurée de la même manière que WordNet, cette ressource permet d'interconnecter les langues pour pouvoir passer de mots dans une langue aux mots similaires dans n'importe quelle autre langue. Cet aspect est particulièrement intéressant car l'UMLS, de son côté, contient également des termes traduits dans plusieurs langues (notamment au travers du MeSH défini dans de nombreuses langues). Pour déterminer une des paires (concept, synset) comme meilleure, il faudrait comparer non plus uniquement les définitions et le nombre de synonymes communs de langue anglaise mais aussi ceux des autres langues. Par exemple, le terme `Reason` en anglais est associé à deux concepts UMLS : ***Reason for (C0392360)*** et ***Reasonning (C0684328)*** et à six synsets WordNet. En utilisant par exemple la partie espagnole d'EuroWordNet (accessible à l'adresse http ://nipadio.lsi.upc.edu/cgi-bin/wei4/public/wei.consult.perl), il est possible d'identifier trois synonymes « causa », « motivo » et « razón » entre le concept ***Reason for*** et le synset ***reason#n#5***. Ainsi, alors que WordNet ne permet pas de désambiguïser la correspondance multiple obtenue par l'UMLS, EuroWordNet le peut.

7.3.3 Conclusion

L'utilisation de l'UMLS et WordNet présente des avantages et des inconvénients et le fait de combiner ces deux ressources compense souvent les problèmes posés par l'un au travers des bénéfices apportés par l'autre. L'UMLS est disponible gratuitement à condition de disposer d'une licence. Pour notre travail, nous avons récupéré en local l'UMLS et WordNet (qui est libre d'accès) afin de pouvoir faire toutes les recherches et manipulations dont nous avions besoin. L'avantage est que l'on peut ainsi manipuler ces ressources de manière plus flexible mais présente l'inconvénient de devoir les mettre à jour quand de nouvelles versions sont disponibles. En ce qui concerne l'UMLS, il faut tout d'abord en éliminer les cycles. Puis, pour l'UMLS comme pour WordNet, il faut recommencer la phase de mises en correspondance des EDs avec les concepts et synsets respectivement. Pour les résultats identiques, aucune modification n'est requise, en revanche si un concept ou un synset ont été supprimés ou modifiés, il faut à nouveau les apparier en comparant les nouvelles propriétés.

7.4 Apport de méthodes formelles pour le schéma global

Le schéma global de notre système est bien formé structurellement puisqu'il est construit sous la forme d'un graphe orienté sans cycle mais n'est pas formel. Kashyap a décrit une méthode pour représenter l'ensemble du Réseau Sémantique au format OWL (relations et types sémantiques) [Kashyap 03] mais cela n'est pas suffisant pour nos besoins. En effet, notre schéma global contient une grande partie du Metathesaurus pour lequel aucune représentation formelle n'a été proposée.

Cela est limitatif pour réaliser certaines tâches plus avancées comme l'amélioration du processus de requêtes, la classification automatique de nouveaux concepts au sein du schéma global ou encore l'ajout d'une ontologie de haut niveau au-dessus de celui-ci. Pour illustrer la possibilité d'intégrer ces différents points dans ce travail, nous utilisons une partie restreinte et simplifiée de notre schéma global se limitant à quelques concepts.

WordNet identifie des correspondances avec des EDs mais qui ne sont pas forcément intégrées telles quelles dans notre système. Si la correspondance d'un ED dans l'UMLS est identifiée au travers d'un synonyme ou hypernyme direct d'un synset, alors les propriétés de ce synset sont ajoutées à la description du concept identifié. Par contre, si le synset n'est associé à aucun concept, aucune information n'est ajoutée dans le schéma global. Dans le premier cas, il peut s'avérer utile d'ajouter un concept au schéma global qui serait plus précis que l'hypernyme direct. Dans le deuxième cas, cet ajout est indispensable si l'on veut pouvoir représenter l'ED initialement trouvé dans WordNet. Nous illustrons notre propos au travers de l'ajout du concept ***Locus*** auquel est attribué l'identifiant ***CN000001***. Ce terme est normalement associé au concept ***Associated topography (C0205145)*** avec notre méthode d'identification de correspondances indirectes. Pour cette section, nous introduisons ***Locus*** comme nouveau concept, fils de ***Associated topography***. L'intégration d'un nouveau concept au sein de notre schéma global permet tout d'abord d'effectuer des recherches plus précises. Une requête de la forme *Locus of HAMP gene* proposera les EDs `Gene Map Locus`, extrait de HPRD, et `Locus`, présent dans GeneCards, alors que jusqu'ici notre système présentait en plus l'ED `Site of Expression`, issu de HPRD (car « Locus » et « Site » appartenaient à un même concept). Cela est intéressant dans la mesure où ce dernier ED ne fournit pas d'informations concernant le locus d'un gène mais les endroits où s'exprime le gène considéré (cf annexe E - page 220). On parviendrait ainsi à fournir des données plus pertinentes et correspondant exactement aux attentes des utilisateurs.

7.4.1 Amélioration du processus de requêtes

Pour améliorer le processus de requêtes, il faudrait décrire les concepts du schéma global de manière formelle, et ce notamment à l'aide de propriétés. En effet, ces dernières pourraient être exploitées au moment de reformuler les requêtes des utilisateurs. Plus précisément, le médiateur se chargerait de rechercher parmi les propriétés des concepts constituant la requête si ceux-ci ont des restrictions concernant un type d'entité biologique particulier. Par exemple, considérons le concept ***Locus (CN000001)*** tout juste introduit. Si on lui ajoute comme propriété le fait d'être un emplacement associé à un gène, on pourra interpréter une requête de la forme *Topography of HAMP gene* comme si l'on voulait implicitement récupérer les informations concernant le **locus** de ce gène. On pourra ainsi inférer que la requête équivaut en réalité à *Locus of HAMP gene*. En pratique, cela implique tout d'abord d'introduire une propriété `associated_with` de type *ObjectProperty* (au sens OWL) avec pour domaine ***Associated topography*** et co-domaine le concept ***Biological Products (C0005522)*** (père du concept ***Genes (C0017337)***). Dans un deuxième temps, il faut préciser le concept ***Locus*** de la manière suivante : une restriction est appliquée sur la propriété `associated_with` indiquant que ce concept est associé à un gène, c'est-à-dire ayant pour co-domaine le concept ***Genes*** (Figure 7.1 page suivante).

```
<owl:Class rdf:about="#CN000001">
  <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
        Locus</rdfs:label>
  <owl:isDefinedBy rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
  the specific site of a particular gene on its chromosome</owl:isDefinedBy>
  <owl:equivalentClass>
   <owl:Class>
    <owl:intersectionOf rdf:parseType="Collection">
     <owl:Restriction>
      <owl:onProperty>
       <owl:ObjectProperty rdf:about="#associated_with"/>
      </owl:onProperty>
      <owl:allValuesFrom rdf:resource="#C0017337"/>
     </owl:Restriction>
     <owl:Class rdf:about="#C0205145"/>
    </owl:intersectionOf>
   </owl:Class>
  </owl:equivalentClass>
</owl:Class>
```

```
<owl:ObjectProperty rdf:about="#associated_with">
  <rdfs:range rdf:resource="#C0005522"/>
  <rdfs:domain rdf:resource="#C0205145"/>
</owl:ObjectProperty>
```

FIG. 7.1 – **Définition du concept *Locus (CN000001)* en OWL ainsi que de la propriété `associated_with`.** L'identifiant du concept est créé arbitrairement, le libellé et la définition (en rose) sont ceux du synset utilisé pour créer ce nouveau concept. Le reste du code correspond à la restriction définie sur la propriété *`associated_with`* (définie dans le cadre en bas à droite, en orange) et le fait que le père de ce concept est ***C0205145***. Les concepts ***C0017337***, ***C0005522*** et ***C0205145*** sont respectivement les concepts ***Genes***, ***Biological Products*** et ***Associated topography***.

```
<owl:Class rdf:ID="CN000002">
  <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
        LocusAllele</rdfs:label>
  <owl:equivalentClass>
   <owl:Class>
    <owl:intersectionOf rdf:parseType="Collection">
     <owl:Class rdf:ID="C0205145"/>
     <owl:Restriction>
      <owl:allValuesFrom>
       <owl:Class rdf:ID="C0002085"/>
      </owl:allValuesFrom>
      <owl:onProperty>
       <owl:ObjectProperty rdf:ID="associated_with"/>
      </owl:onProperty>
     </owl:Restriction>
    </owl:intersectionOf>
   </owl:Class>
  </owl:equivalentClass>
</owl:Class>
```

FIG. 7.2 – **Définition initiale du concept *Locus of Allele (CN000002)* en OWL.** L'identifiant et le libellé (en turquoise) du concept sont créés arbitrairement. Le reste du code correspond à la restriction définie sur la propriété `associated_with` et le fait que ce concept est fils de ***C0205145***. Les concepts ***C0205145*** et ***C0002085*** sont respectivement les concepts ***Associated topography*** et ***Alleles***. Cela signifie que ***Locus of Allele*** est équivalent à l'intersection de ***Associated topography*** et de l'ensemble des instances n'étant reliées qu'à des allèles par la relation `associated_with`.

7.4.2 Classification de nouveaux concepts

Une fois que l'on a introduit des propriétés associées aux concepts du schéma global, il devient possible d'effectuer du raisonnement sur ceux-ci afin de classer automatiquement des nouveaux concepts considérés comme utiles pour représenter les EDs extraits des sources biomédicales intégrées ou à intégrer. Cela diminuerait le travail manuel à réaliser par les concepteurs lors de l'intégration de ces nouveaux concepts. Reprenons l'exemple du concept ***Locus***, supposons que l'on souhaite introduire un nouveau concept, plus spécifique : ***Locus of Allele*** (auquel on attribue l'identifiant ***CN000002***). On définit ce concept comme une topographie concernant les allèles. Pour cela, il faut d'abord définir la classe ***CN000002*** à la racine de notre schéma global, indiquer que c'est un fils du concept ***Associated topography*** puis y restreindre la propriété `associated_with` en précisant qu'elle doit s'appliquer à des allèles (concept ***Alleles (C0002085)***, fils de ***Genes***) (Figure 7.2).

À partir de cette description du nouveau concept ***Locus of Allele*** (que nous avons réalisée dans Protégé avec le plugin OWL [Knublauch 04]), on appelle un raisonneur (ici Pellet [Parsia 04]) au moyen de la commande *ClassifyTaxonomy*. Le concept est automatiquement classé sous le concept ***Locus*** (Figure 7.3 page suivante), ce qui était attendu. En effet, comme ce nouveau concept correspond à la topographie d'un allèle et que celui-ci est un gène, on infère que ***Locus of Allele*** est un locus. Le code OWL définissant le nouveau concept ***Locus of Allele*** est ainsi enrichi du fait que ce nouveau concept est en réalité un fils de ***Locus*** (Figure 7.4 page 183).

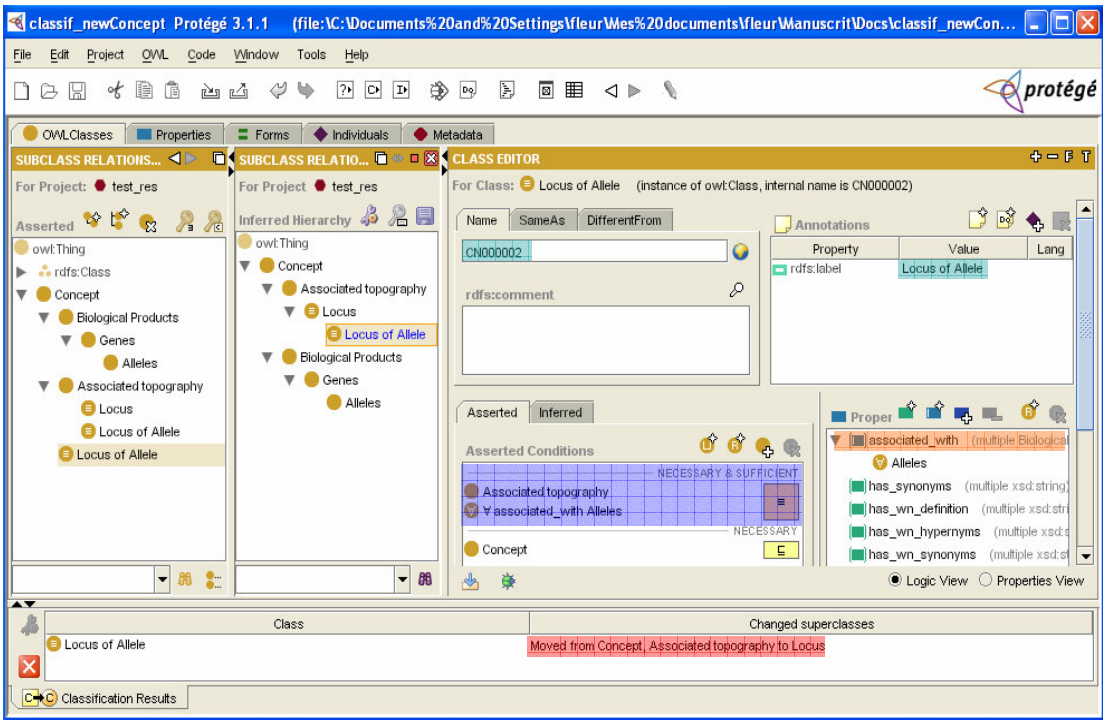


FIG. 7.3 – **Classification du concept *Locus of Allele (CN000002)* sous Protégé avec le plugin OWL.** Au sein d'une description de cinq concepts faisant partie de notre schéma global (excepté le concept ***Locus*** dont nous simulons l'ajout dans cette discussion), on cherche à introduire le nouveau concept ***Locus of Allele***. En vert sont surlignés l'identifiant et le libellé que nous avons ajoutés manuellement. La restriction sur la propriété `associated_with` permet d'indiquer que les produits biologiques auquel est associé ce nouveau concept sont les allèles (définition surlignée en bleu). La fenêtre de gauche présente la hiérarchie de concepts, ***Locus of Allele*** y est présent deux fois car Protégé le déplace directement sous ***Associated topography*** quand on précise que c'est un fils de ce dernier. Une fois cette description effectuée, nous avons exécuté la commande *ClassifyTaxonomy* (Menu *OWL*) qui appelle le raisonneur Pellet. Celui-ci permet de déduire que le nouveau concept est en fait un fils du concept ***Locus***. Les étapes réalisées par le raisonneur sont surlignées en rouge en bas de la fenêtre. Il est possible d'accepter ou de refuser les modifications proposées et si l'on accepte, la fenêtre du milieu où la hiérarchie bien formée est créée apparaît. Le concept ***Locus of Allele*** y est écrit en bleu, ce qui indique qu'il a pu être classé.

```
<owl:Class rdf:ID="CN000002">
  <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
        LocusAllele</rdfs:label>
  <owl:equivalentClass>
   <owl:Class>
    <owl:intersectionOf rdf:parseType="Collection">
     <owl:Class rdf:ID="C0205145"/>
     <owl:Restriction>
      <owl:allValuesFrom>
       <owl:Class rdf:ID="C0002085"/>
      </owl:allValuesFrom>
      <owl:onProperty>
       <owl:ObjectProperty rdf:ID="associated_with"/>
      </owl:onProperty>
     </owl:Restriction>
    </owl:intersectionOf>
   </owl:Class>
  </owl:equivalentClass>
  <rdfs:subClassOf>
   <owl:Class rdf:ID="CN000001"/>
  </rdfs:subClassOf>
</owl:Class>
```

FIG. 7.4 – **Définition du concept *Locus of Allele (CN000002)* en OWL après classification automatique.** Le code OWL est complété par la relation du subsomption entre ***Locus of Allele*** et ***Locus (CN000001)***.

7.4.3 Ontologie de haut niveau

Des travaux assez récents ont considéré l'utilisation d'une ontologie de haut niveau (cf 2.1.2.3.3 page 27), DOLCE, pour rendre WordNet plus rigoureuse et donc plus efficacement exploitable par les applications la nécessitant [Gangemi 03]. Dans ce cadre, il pourrait être utile d'étendre l'utilisation de WordNet dans notre système. En effet, au lieu de compléter uniquement les concepts permettant de représenter les EDs, il faudrait le faire pour l'ensemble des concepts du schéma global et associer, dans un second temps, les synsets les plus hauts dans la hiérarchie avec DOLCE (comme cela a été décrit par Gangemi et al.) pour disposer d'une ontologie mise en correspondance avec une ontologie de haut niveau. Un avantage concerne l'**interopérabilité**. En effet, si notre système peut utiliser une ontologie de haut niveau également intégrée dans d'autres applications, notre travail pourra être relié avec ces autres applications au travers de DOLCE [Noy 04]. Notre expansion de requêtes pourra ainsi être complétée par des connaissances externes. Si la requête est constituée de termes n'existant pas dans notre schéma mais qui sont présents dans d'autres applications utilisant DOLCE, leurs synonymes pourront être récupérés et être utiles s'ils sont présents dans notre schéma global. De plus, les **techniques de mises en correspondance pourraient également être améliorées** par la présence d'une ontologie de haut niveau. Il serait possible d'identifier des correspondances indirectes supplémentaires entre des termes d'ontologies ou de systèmes terminologiques exploitant DOLCE et les concepts du schéma global. Des techniques sémantiques pourraient par ailleurs être développées pour détecter des mises en correspondance se basant sur l'analyse des interprétations que l'on peut faire des concepts se trouvant dans l'ontologie de haut niveau. Ces aspects s'inscrivent dans des perspectives du Web sémantique qui ont, jusqu'ici, été peu investies [Shvaiko 05].

7.5 Processus de requêtes

Le principe de notre processus de requêtes est simple : à partir des termes constituant la requête, le médiateur recherche des concepts correspondants dans le schéma global, en exploitant, s'il y en a, les synonymes et hypernymes associés qui sont issus de WordNet. Une expansion de requêtes est par ailleurs implémentée afin de calculer les descendants des concepts constituant la requête, auxquels sont possiblement associés des EDs. Ce processus permet de répondre à des requêtes impliquant plusieurs sources en récupérant dans les différentes sources toutes les informations correspondant aux EDs associés aux concepts de la requête posée. Cependant, cela ne suffit pas pour traiter des requêtes plus complexes, notamment impliquant des relations entre les concepts constituant la requête ou encore nécessitant de traverser des références croisées entre sources. Un exemple pour chaque perspective est donné en annexe F (page 221).

Ajouter des propriétés aux concepts pour les exploiter au moment où le médiateur reformule la requête permet d'améliorer le traitement des requêtes (cf 7.4.1 page 179). Pour l'instant, il n'y a pas de propriétés associées aux concepts de notre schéma global et les ajouter une par une manuellement constituerait une tâche pénible. Une solution serait d'utiliser les relations associatives existant au niveau du Réseau Sémantique de l'UMLS, qui permettent de relier les types sémantiques. Comme chaque concept est catégorisé par au moins un type sémantique, il est envisageable de récupérer les relations impliquant les types sémantiques associés pour relier les concepts catégorisés par ces derniers. Des auteurs ont cependant souligné que cela peut parfois poser problème [McCray 02], il faudrait donc les vérifier manuellement. Cela permettrait de traiter les requêtes de manière plus spécifique et fournir des résultats plus précis aux utilisateurs. Pour la création de ces propriétés, le travail pré-cité de Kashyap pourrait être exploité [Kashyap 03].

Dans notre travail, nous récupérons les **références croisées** mais il faudrait les intégrer au processus de requêtes. Les systèmes d'intégration basés sur la navigation traversent automatiquement les différents points d'accès entre les sources qu'ils intègrent afin de proposer des chemins possibles répondant aux requêtes complexes posées par les utilisateurs [Cohen-Boulakia 05a]. Il serait intéressant de compléter notre processus de requêtes dans ce sens. Ainsi, il serait possible d'effectuer des requêtes plus complexes en cherchant notamment des informations concernant d'autres entités biologiques que les gènes et les maladies.

7.6 Généralisation - Ré-utilisation

La **généralisation de notre système** est un point intéressant. Nous avons développé ce système afin de faciliter les travaux de recherche et de collecte d'informations des biologistes et des médecins. Cependant, ces besoins ne sont pas spécifiques au domaine biomédical et d'autres domaines nécessitent ce type de système d'intégration, par exemple la recherche d'informations pour les chercheurs en général et la planification de voyages. Les méthodes développées dans notre système imposent un certain nombre de pré-requis pour être transposables à un autre domaine d'application. Tout d'abord, il faut pouvoir interroger les sources à intégrer de ma-

nière dynamique au travers de leur site Web (par exemple, par des GCIs). Cet aspect n'est pas très contraignant car c'est généralement le cas. Il faut aussi constituer un corpus de termes qui seront utilisés pour interroger les sources afin de récupérer un échantillon de pages Web nous permettant d'extraire les EDs pertinents. Ensuite et c'est ce point le plus important, il faut disposer pour ce domaine spécifique d'une ontologie ou au moins d'une ressource terminologique existante suffisamment complète. Plus précisément, il est nécessaire qu'elle contienne des concepts organisés en hiérarchie et pour lesquels des synonymes et définitions existent, si possible. Il faudra personnaliser WordNet pour le domaine étudié en suivant le même modèle réalisé pour le domaine biomédical (c'est-à-dire recenser les domaines WordNet concernant ce domaine et constituer un corpus de termes pertinents présents dans les définitions WordNet). Sous ces conditions, qui requièrent l'existence une ressource terminologique du domaine étudié, notre travail serait applicable à d'autres domaines.

Finalement, **certaines approches développées dans ce travail auraient pu être utilisées par d'autres types d'approches d'intégration**. D'une part, la méthode d'acquisition des schémas locaux est intéressante pour les trois approches d'intégration détaillées dans l'état de l'art. En effet, pour mettre en œuvre un système intégrant de multiples sources, il faut préalablement connaître les sources choisies et disposer d'un minimum d'informations les concernant, en particulier de son schéma. Même si notre méthode récupère uniquement une liste d'éléments de données (et leur type) informant en partie du contenu des sources, cela peut donner une idée aux concepteurs sur l'intérêt potentiel d'une source et constituer un bon point de départ pour savoir quel type d'informations il sera nécessaire de représenter pour l'intégrer.

D'autre part, les techniques de mise en correspondance peuvent s'avérer utiles pour tout type d'approches d'intégration nécessitant la définition d'un schéma global. En effet, une fois ce schéma décrit, les informations présentes dans les schémas locaux doivent être mises en correspondance avec les éléments du schéma global. Dans ce cadre, notre travail propose des solutions intéressantes. Cela concerne non seulement l'approche basée sur la médiation mais également l'approche entrepôt pour lesquelles des mises en correspondance entre le schéma global et les schémas locaux peuvent être utiles si ces deux composants sont définis indépendamment. Dans ce cas, il faut pouvoir traduire les schémas locaux pour qu'ils s'intègrent dans le schéma global. Des méthodes automatisant cette tâche peuvent donc aussi être bénéfiques aux entrepôts de données pour leur conception et leur maintenance. Dans le cas de l'approche basée sur la navigation, comme il n'y a pas de schéma global, ce genre de tâches n'existe pas. Cependant, dans les systèmes BioNavigation et BioGuide où les concepteurs ont complété l'architecture classique au moyen d'un schéma global, ces mises en correspondance sont également à effectuer et nos méthodes peuvent les faciliter. Enfin, l'approche *pair à pair*, où le système ne contient plus uniquement un schéma global mais plusieurs (par type de domaine, par exemple), peut aussi tirer profit de nos méthodes. En particulier, la problématique de mise en correspondance y est capitale et la ré-utilisation de techniques adaptées aux approches d'intégration centralisées est indispensable [Hacid 04].

Chapitre 8

Conclusion générale

En conclusion, nous proposons un système d'intégration basée sur la médiation pour le domaine biomédical. L'objectif est de faciliter la recherche et la collecte d'informations aux biologistes et médecins. Au cours de leurs travaux, ces derniers doivent disposer de données concernant l'existant et ces informations sont généralement accessibles sur Internet. Le problème est qu'elles sont réparties dans des sources distribuées, autonomes et hétérogènes à de multiples niveaux. C'est pour cela que le développement d'un système d'intégration est incontournable.

Pour concevoir notre système, nous nous sommes focalisés sur l'automatisation de différentes étapes. Ainsi, nous avons développé une méthode automatique pour extraire des sources de données à intégrer leurs éléments de données reflétant leur schéma, c'est-à-dire le type d'informations qu'elles contiennent. Ensuite, pour mettre en correspondance ces éléments de données avec les éléments du schéma global (basé sur une ressource déjà existante), nous avons mis en œuvre des techniques terminologiques et structurelles au niveau *schéma*. À partir d'une première correspondance des éléments de données directement dans le schéma global, nous avons validé et désambiguïsé ces correspondances grâce à l'utilisation d'une ressource externe. De plus, quand aucune correspondance n'a pu être trouvée directement dans le schéma global, la ressource externe permet parfois de proposer une correspondance indirecte. De cette façon, nous avons pu enrichir le schéma global.

Parallèlement, nous avons proposé des méthodes terminologiques situées au niveau *instances*. Celles-ci ont permis d'une part de typer les éléments de données extraits des sources afin de compléter leur schéma local. D'autre part, nous avons identifié des correspondances entre éléments de données au travers de leurs valeurs afin de vérifier et de compléter les correspondances obtenues au niveau *schéma*.

Nous avons réalisé un système d'intégration utilisant ces différentes méthodes. Celui-ci permet aux utilisateurs de poser des requêtes concernant certaines entités biologiques. Le processus de requêtes se base sur le schéma global, et exploite au travers de celui-ci la synonymie et la hiérarchie pour étendre les requêtes des utilisateurs. Les adaptateurs peuvent également exploiter le type des éléments de données associés aux éléments du schéma global considérés comme pertinents vis à vis de la requête. Les utilisateurs obtiennent finalement les valeurs spécifiques associées aux éléments de données qu'ils ont sélectionnés.

L'automatisation de certaines tâches de conception permet de faciliter la gestion de l'évolution du système. Typiquement, l'intégration d'une nouvelle source implique de nombreuses étapes similaires à celles nécessaires au moment de la conception, comme l'acquisition de son schéma local et la mise en correspondance de ses éléments de données avec les éléments du schéma global. Comme lors de la conception, la plupart des tâches sont automatiques même s'il reste des cas où les administrateurs du système doivent intervenir. L'automatisation partielle des tâches de conception et d'évolution est l'élément clé de ce travail. Les systèmes d'intégration existants qui adoptent la même approche ne se focalisent généralement pas sur ces aspects alors qu'ils impliquent des tâches particulièrement lourdes pour les concepteurs. Il est de plus déterminant de garantir aux biologistes et aux médecins que les informations accessibles au travers de ce type de système sont à jour qu'il est possible d'intégrer à tout moment une nouvelle source qui leur semblerait pertinente. Nos approches facilitent les tâches nécessaires pour cela.

Enfin, nous avons souligné les possibles perspectives que l'on pourrait considérer pour compléter notre système. En particulier, la représentation de notre schéma global dans un langage formel n'est pas suffisant. Il faudrait en tirer profit, notamment pour enrichir notre schéma global et ainsi fournir des informations plus précises aux biologistes et médecins. De plus, le processus de requêtes de notre prototype nécessite d'être approfondi, par exemple en introduisant des propriétés aux éléments du schéma global qui pourraient être exploitées pour traiter des requêtes plus complexes.

Bibliographie

[Alonso-Calvo 06] R. Alonso-Calvo, V. Maojo, H. Billhardt, F. Martin-Sanchez, M. Garcia-Remesal & D. Perez-Rey. *An agent- and ontology-based system for integrating public gene, protein, and disease databases.* Journal of Biomedical Informatics, vol. In Press, Corrected Proof, 2006.

[Antoniou 04] G. Antoniou & F. van Harmelen. A semantic web primer (cooperative information systems). The MIT Press, April 2004.

[Aronson 01] A. Aronson. *Effective Mapping of Biomedical Text to the UMLS Metathesaurus : The MetaMap Program.* In Proc AMIA 2001, pages 17–21, 2001.

[Ashburner 00] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin & G. Sherlock. *Gene Ontology : tool for the unification of biology.* Nature Genetics, vol. 25, pages 25–29, May 2000.

[Aussenac-Gilles 04] N. Aussenac-Gilles & J. Mothe. *Ontologies as Background Knowledge to Explore Document Collections.* In RIAO 2004 , Avignon, pages 129–142, 26-28 avril 2004.

[Baader 91] F. Baader & B. Hollunder. *A Terminological Knowledge Representation System with Complete Inference Algorithms.* In PDK '91 : Proceedings of the International Workshop on Processing Declarative Knowledge, pages 67–86, London, UK, 1991. Springer-Verlag.

[Baader 03] F. Baader, D. Calvanese, DL. McGuinness, D. Nardi & PF. Patel-Schneider, editeurs. The description logic handbook : theory, implementation, and applications. Cambridge University Press, New York, NY, USA, 2003.

[Bachimont 00] B. Bachimont. Engagement sémantique et engagement ontologique : Conception et réalisation d'ontologies en ingénierie des connaissances, chapitre 19, pages 305–324. Eyrolles, 2000.

[Baker 99] PG Baker, CA Goble, S Bechhofer, NW Paton, R Stevens & A Brass. *An ontology for bioinformatics applications.* Bioinformatics, vol. 15, no. 6, pages 510–520, 1999.

[Bard 04] JBL Bard & SY Rhee. *Ontologies in biology : design, applications and future challenges.* Nat Rev Genet, vol. 5, no. 3, pages 213–222, Mar 2004.

[Baziz 03] M. Baziz, N. Aussenac-Gilles & M. Boughanem. *Exploitation des Liens Sémantiques pour l¿Expansion de Requêtes dans un Système de Recherche d'Information.* In XXIème Congrès INFORSID 2003, pages 121–134, Inforsid, 20 rue Axel Duboul - 31000 Toulouse, janvier 2003. INFORSID.

[Ben-Miled 04] Z. Ben-Miled, N. Li, Y. Liu, Y. He, Lynch E. & O. Bukhres. *On the Integration of a Large Number of Life Science Web Databases.* In DILS, pages 172–186, 2004.

[Ben Miled 05] Z. Ben Miled, N. Li & O. Bukhres. *BACIIS : Biological and Chemical Information Integration SYstems.* Journal of Database Management, vol. 16(3), pages 73–85, 2005.

[Benson 06] DA. Benson, I. Karsch-Mizrachi, DJ. Lipman, J. Ostell & DL. Wheeler. *GenBank.* Nucl. Acids Res., vol. 34, no. Database Issue, pages D16–20, 2006.

[Berners-Lee 01] T. Berners-Lee, J. Hendler & O. Lassila. *The Semantic Web.* In Scientific American, May 2001.

[Birkland 06] A. Birkland & G. Yona. *BIOZON : a system for unification, management and analysis of heterogeneous biological data.* BMC Bioinformatics, vol. 7, page 70, 2006.

[Blake 06] JA. Blake, JT. Eppig, CJ. Bult, JA. Kadin, JE. Richardson & Mouse Genome Database Group. *The Mouse Genome Database (MGD) : updates and enhancements.* Nucl. Acids Res., vol. 34, no. Database Issue, pages D562–567, 2006.

[Bodenreider 01] O. Bodenreider. *Circular Hierarchical Relationships in the UMLS : Etiology.* Proc AMIA Symp, pages 57–61, 2001.

[Bodenreider 04] O. Bodenreider. *The Unified Medical Language System (UMLS) : integrating biomedical terminology.* Nucl. Acids Res., vol. 32, no. DataBase Issue, pages D267–270, 2004.

[Boeckmann 03] B. Boeckmann, A. Bairoch, R. Apweiler, M.C. Blatter, A. Estreicher, E. Gasteiger, M.J. Martin, K. Michoud, C. O'Donovan, I. Phan, S. Pilbout & M. Schneider. *The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003.* Nucleic Acids Res, vol. 31, pages 365–370, 2003.

[Borst 97] P. Borst, H. Akkermans & J. Top. *Engineering ontologies.* International Journal of Human-Computer Studies, vol. 46, no. 2-3, pages 365–406, February 1997.

[Bray 00] T. Bray, J. Paoli, CM. Sperberg-McQueen & E. Maler. *Extensible Markup Language (XML) 1.0.* W3C, 1.1 edition, October 2000. URL : http ://www.w3c.org/TR/REC-xml.

[Brickley 00] D. Brickley & R. Guha. *Resource Description Framework (RDF) Schema Specification 1.0.* W3C, 1.0 edition, 2000. URL : http ://www.w3.org/TR/2000.

[Bry 03] F. Bry & P. Kröger. *A Computational Biology Database Digest : Data, Data Analysis, and Data Management.* Distrib. Parallel Databases, vol. 13, no. 1, pages 7–42, 2003.

[Buneman 95] P. Buneman, SB. Davidson, K. Hart, GC. Overton & L. Wong. *A Data Transformation System for Biological Data Sources.* In VLDB, pages 158–169, 1995.

[Burgun 01a] A. Burgun & O. Bodenreider. *Comparing terms, concepts and semantic classes in WordNet and the Unified Medical Language System.* In NAACL01 Workshop on WordNet and Other Lexical Resources : Applications, Extensions and Cuomizations, pages 77–82, 2001.

[Burgun 01b] A. Burgun & O. Bodenreider. *Mapping the UMLS Semantic Network into general ontologies.* Proc AMIA Symp, pages 81–85, 2001.

[Burgun 01c] A. Burgun & O. Bodenreider. *Methods for exploring the semantics of the relationships between cooccurring UMLS concepts*, 2001.

[Busse 99] S Busse, RD Kutsche, U Leser & H Weber. *Federated Information Systems : Concepts, Terminology and Architectures.* Rapport technique, TU Berlin, 1999.

[Buttler 02] D. Buttler, M. Coleman, T. Critchlow, R. Fileto, W. Han, C. Pu, D. Rocco & L. Xiong. *Querying multiple bioinformatics information sources : can semantic web research help ?* SIGMOD Rec., vol. 31, no. 4, pages 59–64, 2002.

[Cannata 05] N. Cannata, E. Merelli & RB. Altman. *Time to Organize the Bioinformatics Resourceome.* PLoS Computational Biology, vol. 1(7), page e76, 2005.

[Chandrasekaran 99] B Chandrasekaran, JR Josephson & VR Benjamins. *What are ontologies and why do we need them ?* IEEE Intelligent Systems, vol. 14, no. 1, pages 20–26, 1999.

[Chen 76] PP. Chen. *The Entity-Relationship Model - Toward a Unified View of Data.* Transactions on Database Systems, vol. 1, no. 1, pages 9–36, 1976.

[Cimino 98] JJ. Cimino. *Auditing the Unified Medical Language System with Semantic Methods.* J Am Med Inform Assoc, vol. 5, no. 1, pages 41–51, 1998.

[Codd 70] EF. Codd. *A Relational Model of Data for Large Shared Data Banks.* Commun. ACM, vol. 13, no. 6, pages 377–387, 1970.

[Cohen Boulakia 04] S. Cohen Boulakia, S. Lair, N. Stransky, S. Graziani, F. Radvanyi, E. Barillot & C. Froidevaux. *Selecting biomedical data sources according to user preferences.* Bioinformatics, vol. 20 Suppl 1, pages I86–I93, Aug 2004.

[Cohen-Boulakia 05a] S. Cohen-Boulakia. *Intégration de données biologiques : Sélection de sources centrée sur l'utilisateur.* PhD thesis, Université de Paris Sud XI, 2005.

[Cohen Boulakia 05b] S. Cohen Boulakia, SB. Davidson & C. Froidevaux. *A User-Centric Framework for Accessing Biological Sources and Tools.* In DILS, pages 3–18, 2005.

[Consortium 06] Gene Ontology Consortium. *The Gene Ontology (GO) project in 2006.* Nucl. Acids Res., vol. 34, no. Database Issue, pages D322–326, 2006.

[Cooper 98] DN Cooper, EV Ball & M Krawczak. *The human gene mutation database.* Nucl. Acids Res., vol. 26, no. 1, pages 285–287, 1998.

[Corcho 03] O. Corcho, M. Fernandez-Lopez & A. Gomez-Perez. *Methodologies, tools and languages for building ontologies : where is their meeting point ?* Data Knowl. Eng., vol. 46, no. 1, pages 41–64, 2003.

[Covitz 03] PA. Covitz, F. Hartel, C. Schaefer, S. De Coronado, G. Fragoso, H. Sahni, S. Gustafson & KH. Buetow. *caCORE : A common infrastructure for cancer informatics.* Bioinformatics, vol. 19, no. 18, pages 2404–2412, 2003.

[Crescenzi 01] V. Crescenzi, G. Mecca & P. Merialdo. *RoadRunner : Towards Automatic Data Extraction from Large Web Sites.* In VLDB, pages 109–118, 2001.

[Darmoni 00] SJ. Darmoni, JP. Leroy, F. Baudic, M. Douyère, J. Piot & B. Thirion. *CISMeF : a structured health resource guide.* Methods Inf Med, vol. 39, no. 1, pages 30–35, Mar 2000.

[Davidson 95] SB. Davidson, C. Overton & P. Buneman. *Challenges in integrating biological data sources.* J Comput Biol, vol. 2, no. 4, pages 557–572, 1995.

[Davidson 97] SB. Davidson, GC. Overton, V. Tannen & L. Wong. *BioKleisli : A Digital Library for Biomedical Researchers.* Int. J. on Digital Libraries, vol. 1, no. 1, pages 36–53, 1997.

[Davidson 01] SB. Davidson, J. Crabtree, BP. Brunk, J. Schug, V. Tannen, GC. Overton & CJ. Stoeckert Jr. *K2/Kleisli and GUS : experiments in integrated access to genomic data sources.* IBM Syst. J., vol. 40, no. 2, pages 512–531, 2001.

[de Keizer 00] NF. de Keizer, A. Abu-Hanna & JH. Zwetsloot-Schonk. *Understanding terminological systems I : Terminology and typology.* Methods Inf Med, vol. 39, no. 1, pages 16–21, Mar 2000.

[Deshpande 05] N. Deshpande, KJ. Addess, WF. Bluhm, JC. Merino-Ott, W Townsend-Merino, Q Zhang, C. Knezevich, L. Xie, L. Chen, Z. Feng, R. Kramer Green, JL. Flippen-Anderson, J. Westbrook, HM. Berman & PE. Bourne. *The RCSB Protein Data Bank : a redesigned query system and relational database based on the mmCIF schema.* Nucl. Acids Res., vol. 33, no. Database Issue, pages D233–237, 2005.

[Do 02] HH. Do & E. Rahm. *COMA - A System for Flexible Combination of Schema Matching Approaches.* In VLDB, pages 610–621, 2002.

[Doan 03] A. Doan, J. Madhavan, P. Domingos & A. Halevy. *Ontology matching : A machine learning approach*, 2003.

[Efthimiadis 96] EN. Efthimiadis. *Query expansion.* Annual review of information science and technology, vol. 31, pages 121–187, 1996.

[Ehrig 04] M. Ehrig & Y. Sure. *Ontology Mapping - An Integrated Approach.* In ESWS, pages 76–91, 2004.

[Eikvil 99] L. Eikvil. *Information Extraction from World Wide Web - A Survey.* Rapport technique 945, Norweigan Computing Center, 1999.

[Etzold 93] T. Etzold & P. Argos. *SRS–an indexing and retrieval tool for flat file data libraries.* Comput Appl Biosci, vol. 9, no. 1, pages 49–57, Feb 1993.

[Euzenat 04] J. Euzenat, T. Le Bach, J. Barrasa, P. Bouquet, J. De Bo, R. Dieng, M. Ehrig, M. Hauswirth, M. Jarrar, R. Lara, D. Maynard, A. Napoli, G. Stamou, H. Stuckenschmidt, P. Shvaiko, S. Tessaris, S. Van Acker & I. Zaihrayeu. *State of the Art on Ontology Alignment.* Knowledge Web Deliverable #D2.2.3, INRIA, Saint Ismier, 2004.

[Eyre 06] TA. Eyre, F. Ducluzeau, TP. Sneddon, S. Povey, EA. Bruford & MJ. Lush. *The HUGO Gene Nomenclature Database, 2006 updates.* Nucl. Acids Res., vol. 34, no. Database Issue, pages D319–321, 2006.

[Fellbaum 06] C. Fellbaum, U. Hahn & B. Smith. *Towards new information resources for public health-From WordNet to MedicalWordNet.* J Biomed Inform, vol. 39(3), pages 321–332, 2006.

[Freier 02] A. Freier, R. Hofestädt, M. Lange, U. Scholz & A. Stephanik. *BioDataServer : a SQL-based service for the online integration of life science data.* In Silico Biol, vol. 2, no. 2, pages 37–57, 2002.

[Friedman 99] M. Friedman, AY. Levy & TD. Millstein. *Navigational Plans for Data Integration.* In Intelligent Information Integration, 1999.

[Froidevaux 02] C. Froidevaux & S. Cohen Boulakia. *Intégration de Sources de Données Génomiques du Web.* In Journées scientifiques du Web Sémantique, 2002. http ://www.lalic.paris4.sorbonne.fr/stic/octobre/programme0209.html.

[Galperin 06] MY. Galperin. *The Molecular Biology Database Collection : 2006 update.* Nucl. Acids Res., vol. 34, no. Database Issue, pages D3–5, 2006.

[Gangemi 03] A. Gangemi, N. Guarino, C. Masolo & A. Oltramari. *Sweetening WORDNET with DOLCE.* AI Mag., vol. 24, no. 3, pages 13–24, 2003.

[Giunchiglia 05] F. Giunchiglia, P. Shvaiko & M. Yatskevich. *S-Match : an algorithm and an implementation of semantic matching.* In Y. Kalfoglou, M. Schorlemmer, A. Sheth, S. Staab & M. Uschold, editeurs, Semantic Interoperability and Integration, numéro 04391 in Dagstuhl Seminar Proceedings. Internationales Begegnungs- und Forschungszentrum fuer Informatik (IBFI), Schloss Dagstuhl, Germany, 2005.

[Gruber 93] TR. Gruber. *A translation approach to portable ontology specifications.* Knowledge Acquisition, vol. 5, no. 2, pages 199–220, 1993.

[Gu 00] H. Gu, Y. Perl, J. Geller, M. Halper, L. Liu & J. Cimino. *Representing the UMLS as an OODB : Modeling issues and advantages.* Journal of the American Medical Informatics Association, vol. 7(1), pages 66–80, 2000.

[Guarino 94] N. Guarino, M. Carrara & P. Giaretta. *Formalizing Ontological Commitment.* In AAAI, pages 560–567, 1994.

[Guarino 95] N. Guarino & R. Poli. *Formal Ontology in Conceptual Analysis and Knowledge Representation.* Special issue of the International Journal of Human and Computer Studies, vol. 43(5/6), pages 625–640, 1995.

[Guarino 97a] N. Guarino. *Some Organizing Principles for a Unified Top-Level Ontology.* Proceedings of AAAI Spring Symposium on Ontological Engineering. Stanford, CA, AAAI Press., pages 57–63, 1997.

[Guarino 97b] N. Guarino. *Understanding, building and using ontologies.* Int. J. Hum.-Comput. Stud., vol. 46, no. 2-3, pages 293–310, 1997.

[Guérin 05] E. Guérin, G. Marquet, A. Burgun, O. Loréal, L. Berti-Equille, U. Leser & F. Moussouni. *Integrating and Warehousing Liver Gene Expression Data and Related Biomedical Resources in GEDAW.* In DILS, pages 158–174, 2005.

[Hacid 04] MS. Hacid & C. Reynaud. *L'intégration de sources de données.* Revue Information - Interaction - Intelligence (R I3), vol. 4, no. 2, 2004.

[Hahn 04] U Hahn & S Schulz. *Boosting the Medical Knowledge Infrastructure - A Feasibility Study on Very Large Terminological Knowledge Bases.* Proc Symp on Engineering of Intelligent Systems, 2004.

[Halevy 03] AY. Halevy, ZG. Ives, D. Suciu & I. Tatarinov. *Schema Mediation in Peer Data Management Systems.* In ICDE, pages 505–516, 2003.

[Hamosh 05] A. Hamosh, AF. Scott, JS. Amberger, CA. Bocchini & VA. McKusick. *Online Mendelian Inheritance in Man (OMIM), a knowledgebase of human genes and genetic disorders.* Nucl. Acids Res., vol. 33, no. Database issue, pages D514–D517, Jan 2005.

[Hendler 01] J Hendler. *Agents and the Semantic Web.* IEEE Intelligent Systems, vol. 16, no. 2, 2001.

[Hernandez 04] T. Hernandez & S Kambhampati. *Integration of biological sources : current systems and challenges ahead.* SIGMOD Rec., vol. 33, no. 3, pages 51–60, 2004.

[Horrocks 02] I. Horrocks. *DAML+OIL : a Description Logic for the Semantic Web.* IEEE Data Engineering Bulletin, vol. 25, no. 1, pages 4–9, 2002.

[Kalfoglou 03] Y. Kalfoglou & M. Schorlemmer. *Ontology Mapping : The State of the Art.* The Knowledge Engineering Review, vol. 18(1), pages 1–31, 2003.

[Kanehisa 06] M. Kanehisa, S. Goto, M. Hattori, KF. Aoki-Kinoshita, M. Itoh, S. Kawashima, T. Katayama, M. Araki & M. Hirakawa. *From genomics to chemical genomics : new developments in KEGG.* Nucl. Acids Res., vol. 34, no. Database Issue, pages D354–357, 2006.

[Kang 03] J. Kang & JF. Naughton. *On schema matching with opaque column names and data values.* In SIGMOD '03 : Proceedings of the 2003 ACM SIGMOD international conference on Management of data, pages 205–216, New York, NY, USA, 2003. ACM Press.

[Karp 96] PD. Karp. *A Strategy for Database Interoperation.* Journal of Computational Biology, vol. 2, no. 4, pages 573–583, 1996.

[Kashyap 98] V. Kashyap & A. Sheth. *Semantic Heterogeneity in Global Information Systems : The Role of Metadata, Context and Ontologies.* In M. Papazoglou and G. Schlageter, editors, Cooperative Information Systems : Current Trends and Applications., pages 139–178, 1998.

[Kashyap 03] V. Kashyap & A. Borgida. *Representing the UMLS Semantic Network Using OWL : (Or « What's in a Semantic Web Link ? »).* In International Semantic Web Conference, pages 1–16, 2003.

[Kasprzyk 04] A. Kasprzyk, D. Keefe, D. Smedley, D. London, W. Spooner, C. Melsopp, M. Hammond, P. Rocca-Serra & E. Cox T.and Birney. *EnsMart : A Generic System for Fast and Flexible Access to Biological Data.* Genome Res., vol. 14, no. 1, pages 160–169, 2004.

[Kefi 06] H. Kefi, B. Safar & C. Reynaud. *Alignement de taxonomies pour l'interrogation de sources d'information hétérogènes.* In Actes du congrès francophone Reconnaissance des Formes et Intelligence Artificielle (RFIA), 2006.

[Knight 94] K. Knight & SK. Luk. *Building a large-scale knowledge base for machine translation.* In AAAI '94 : Proceedings of the twelfth national conference on Artificial intelligence (vol. 1), pages 773–778, Menlo Park, CA, USA, 1994. American Association for Artificial Intelligence.

[Knublauch 04] H. Knublauch, RW. Fergerson, NF. Noy & MA. Musen. *The Protégé OWL Plugin : An Open Development Environment for Semantic Web Applications.* In Proceeding of the Third International Semantic Web Conference (ISWC2004), volume 3298 of *Lecture Notes in Computer Science*, pages 229–243. Springer Berlin Heidelberg, 2004.

[Köhler 02] J. Köhler & S. Schulze-Kremer. *The Semantic Metadatabase (SEMEDA) : Ontology Based Integration of Federated Molecular Biological Data Sources.* In Silico Biology, vol. 2, pages 219–31, 2002.

[Köhler 03] J. Köhler, S. Philippi & M. Lange. *SEMEDA : ontology based semantic integration of biological databases.* Bioinformatics, vol. 19, no. 18, pages 2420–2427, 2003.

[Kouranov 06] A. Kouranov, L. Xie, J. de la Cruz, L. Chen, J. Westbrook, PE. Bourne & HM. Berman. *The RCSB PDB information portal for structural genomics.* Nucl. Acids Res., vol. 34, no. Database Issue, pages D302–305, 2006.

[Kumar 03] A. Kumar & B. Smith. *The Unified Medical Language System and the Gene Ontology : Some Critical Reflections.* In KI2003 : Advances in AI, pages 135–148. A. Günter and R. Kruse and B. Neumann, 2003.

[Kushmerick 97] N. Kushmerick, DS. Weld & RB. Doorenbos. *Wrapper Induction for Information Extraction.* In Intl. Joint Conference on Artificial Intelligence (IJCAI), pages 729–737, 1997.

[Lacroix 05] Z. Lacroix, K. Parekh, ME. Vidal, M. Cardenas & N. Marquez. *BioNavigation : Selecting Optimum Paths Through Biological Resources to Evaluate Ontological Navigational Queries.* In DILS, pages 275–283, 2005.

[Larson 89] JA. Larson, SB. Navathe & R. Elmasri. *A Theory of Attributed Equivalence in Databases with Application to Schema Integration.* IEEE Trans. Softw. Eng., vol. 15, no. 4, pages 449–463, 1989.

[Lassila 98] O. Lassila & R. Swick. *Resource Description Framework (RDF) model and syntax specification.* W3C Working Draft WD-rdf-syntax-19981008., 1998.

[Laublet 02] P Laublet, C Reynaud & Charlet J. *Sur quelques aspects du Web Sémantique.* In Assises du GDR I3, Eds Cépadues, 2002.

[Lee 06] TJ. Lee, Y. Pouliot, V. Wagner, P. Gupta, DWJ. Stringer-Calvert, Tenenbaum JD. & PD. Karp. *BioWarehouse : a bioinformatics database warehouse toolkit.* BMC Bioinformatics, vol. 7, page 170, 2006.

[Lenzerini 02] M. Lenzerini. *Data Integration : A Theoretical Perspective.* In PODS, pages 233–246, 2002.

[Letovsky 98] SI Letovsky, RW Cottingham, CJ Porter & PWD Li. *GDB : the Human Genome Database.* Nucl. Acids Res., vol. 26, no. 1, pages 94–99, 1998.

[Lin 02] SH. Lin & JM. Ho. *Discovering informative content blocks from Web documents.* In KDD '02 : Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, pages 588–593, New York, NY, USA, 2002. ACM Press.

[Lindberg 93] DA. Lindberg, BL. Humphreys & AT. McCray. *The Unified Medical Language System.* Methods Inf Med, vol. 32, no. 4, pages 281–291, Aug 1993.

[Louie 06] B Louie, P Mork, F Martin-Sanchez, A Halevy & P Tarczy-Hornoch. *Data integration and genomic medicine.* J Biomed Inform, vol. In Press, 2006.

[Maedche 02] A. Maedche & S. Staab. *Measuring Similarity between Ontologies.* In EKAW, pages 251–263, 2002.

[Maglott 05] D. Maglott, J. Ostell, KD. Pruitt & T. Tatusova. *Entrez Gene : gene-centered information at NCBI.* Nucl. Acids Res., vol. 33, no. Database Issue, pages D54–58, 2005.

[Mahoui 05] M. Mahoui, H. Kulkarni, N. Li, Z. Ben-Miled & K. Börner. *Semantic Correspondence in Federated Life Science Data Integration Systems.* In DILS, pages 137–144, 2005.

[Markowitz 97] VM. Markowitz, IM. Chen, AS. Kosky & E. Szeto. *Facilities for exploring molecular biology databases on the Web : a comparative study.* Pac Symp Biocomput, pages 256–267, 1997.

[Marquet 05] G Marquet, E Guérin, F Moussouni, O Loréal & A Burgun. *UMLS-based biomedical annotation of functional genomic data.* In Actes de Journées Ouvertes Biologie Informatique Mathématiques (JOBIM-2005), pages 45–54, 2005.

[Masolo 02] C. Masolo, S. Borgo, A. Gangemi, N. Guarino, A. Oltramari & L. Schneider. *WonderWeb Deliverable D17. The WonderWeb Library of Foundational Ontologies and the DOLCE ontology*, 2002.

[McCray 94] AT. McCray, S. Srinivasan & AC. Browne. *Lexical methods for managing variation in biomedical terminologies.* Proc AMIA Symp, pages 235–239, 1994.

[McCray 01] AT McCray, A Burgun & O Bodenreider. *Aggregating UMLS semantic types for reducing conceptual complexity.* Proceedings of Medinfo, vol. 10, no. Pt 1, pages 216–220, 2001.

[McCray 02] AT McCray & O Bodenreider. A conceptual framework for the biomedical domain, chapitre The semantics of relationships : an interdisciplinary perspective. Editors Green R, Bean CA, Myaeng SH, pages 181–198. Boston : Kluwer Academic Publishers, 2002.

[Mendonca 98] EA. Mendonca, JJ. Cimino, KE. Campbell & KA. Spackman. *Reproducibility of interpreting « and » and « or » in termi-nology systems.* Proc AMIA Symp, pages 790–794, 1998.

[Miller 98] G. Miller, editeur. Wordnet : An electronic lexical database (language, speech, and communication). The MIT Press, May 1998.

[Miller 01] RJ. Miller, MA. Hernandez, LM. Haas, L. Yan, CT. Howard Ho, R. Fagin & L. Popa. *The Clio project : managing heterogeneity.* SIGMOD Rec., vol. 30, no. 1, pages 78–83, 2001.

[Minsky 75] M. Minsky. *A framework for Representing Knowledge.* In P.M Winston, editeur, The Psychology of Computer Vision, pages 211–277. McGraw Hill, New York, 1975.

[Mishra 06] Gopa R. Mishra, M. Suresh, K. Kumaran, N. Kannabiran, Shubha Suresh, P. Bala, K. Shivakumar, N. Anuradha, Raghunath Reddy, T. Madhan Raghavan, Shalini Menon, G. Hanumanthu, Malvika Gupta, Sapna Upendran, Shweta Gupta, M. Mahesh, Bincy Jacob, Pinky Mathew, Pritam Chatterjee, K. S. Arun, Salil Sharma, K. N. Chandrika, Nandan Deshpande, Kshitish Palvankar, R. Raghavnath, R. Krishnakanth, Hiren Karathia, B. Rekha, Rashmi Nayak, G. Vishnupriya, H. G. Mohan Kumar,

M. Nagini, G. S. Sameer Kumar, Rojan Jose, P. Deepthi, S. Sujatha Mohan, T. K. B. Gandhi, H. C. Harsha, Krishna S. Deshpande, Malabika Sarker, T. S. Keshava Prasad & Akhilesh Pandey. *Human protein reference database–2006 update.* Nucl. Acids Res., vol. 34, no. Database Issue, pages D411–414, 2006.

[Mork 01] P. Mork, A. Halevy & P. Tarczy-Hornoch. *A model for data integration systems of biomedical data applied to online genetic databases.* Proc AMIA Symp, pages 473–477, 2001.

[Mork 05] P. Mork, R. Shaker & P. Tarczy-Hornoch. *The Multiple Roles of Ontologies in the BioMediator Data Integration System.* In DILS, pages 96–104, 2005.

[Mougin 04] F. Mougin, A. Burgun, O. Bodenreider & P. Le Beux. *Towards the automatic generation of biomedical sources schema.* Proceedings of Medinfo, vol. 11, no. Pt 2, pages 783–787, 2004.

[Mougin 05] F. Mougin & O. Bodenreider. *Approaches to eliminating cycles in the UMLS Metathesaurus : naive vs. formal.* Proc AMIA Symp, pages 550–554, 2005.

[Mougin 06a] F. Mougin & O. Bodenreider. *Éliminer les cycles dans les systèmes terminologiques : comparaison de deux approches.* In Actes du congrès francophone Reconnaissance des Formes et Intelligence Artificielle (RFIA), 2006.

[Mougin 06b] F. Mougin, A. Burgun & O. Bodenreider. *Data integration through data elements : Mapping data elements to terminological resources.* Proc Second International Symposium on Semantic Mining in Biomedicine, pages 52–59, 2006.

[Mougin 06c] F. Mougin, A. Burgun & O. Bodenreider. *Mapping data elements to terminological resources for integrating biomedical data sources.* BMC BioInformatics, vol. 7, no. Suppl 3 :S6, Nov 24 2006.

[Mougin 06d] F. Mougin, A. Burgun & O. Bodenreider. *Using WordNet to Improve the Mapping of Data Elements to UMLS for Data Sources Integration.* Proc AMIA Symp, pages 574–578, 2006.

[Mulder 05] Nicola J. Mulder, Rolf Apweiler, Teresa K. Attwood, Amos Bairoch, Alex Bateman, David Binns, Paul Bradley, Peer Bork, Phillip Bucher, Lorenzo Cerutti, Richard Copley, Emmanuel Courcelle, Ujjwal Das, Richard Durbin, Wolfgang Fleischmann, Julian Gough, Daniel Haft, Nicola Harte, Nicolas Hulo, Daniel Kahn, Alexander Kanapin, Maria Krestyaninova, David Lonsdale, Rodrigo Lopez, Ivica Letunic, Martin Madera, John Maslen, Jennifer McDowall, Alex Mitchell, Anastasia N. Nikolskaya, Sandra Orchard, Marco Pagni, Chris P. Ponting, Emmanuel Quevillon, Jeremy Selengut, Christian J. A. Sigrist, Ville Silventoinen, David J. Studholme, Robert Vaughan & Cathy H. Wu. *InterPro, progress and status in 2005.* Nucl. Acids Res., vol. 33, no. Database Issue, pages D201–205, 2005.

[Napoli 97] A. Napoli. *Une introduction aux logiques de descriptions.* rr RR-3314, inria, 1997.

[Naumann 02] F. Naumann, CT. Ho, X. Tian, L. Haas & N. Megiddo. *Attribute Classification Using Feature Analysis.* In ICDE '02 : Proceedings of the 18th International Conference on Data Engineering, page 271, Washington, DC, USA, 2002. IEEE Computer Society.

[Nilsson 02] M Nilsson, M Palmèr & A Naeve. *Semantic Web Meta-data for e-Learning, Some Architectural Guidelines.* In Proceedings of the 11th World Wide Web Conference, 2002.

[Noy 04] NF Noy. *Semantic Integration : A Survey Of Ontology-Based Approaches.* SIGMOD Record, vol. 33, no. 4, pages 65–70, 2004.

[Okubo 06] K. Okubo, H. Sugawara, T. Gojobori & Y. Tateno. *DDBJ in preparation for overview of research activities behind data submissions.* Nucl. Acids Res., vol. 34, no. Database Issue, pages D6–9, 2006.

[Palpanas 00] T. Palpanas. *Knowledge discovery in data warehouses.* SIGMOD Rec., vol. 29, no. 3, pages 88–100, 2000.

[Parsia 04] B. Parsia & E. Sirin. *Pellet : An OWL DL Reasoner.* In 3rd International Semantic Web Conference (ISWC2004), 2004.

[Pease 02] A. Pease, I. Niles & J. Li. *The Suggested Upper Merged Ontology : A Large Ontology for the Semantic Web and its Applications.* In A. Pease, editeur, Ontologies and the Semantic Web, Papers from the AAAI Workshop. AAAI Press, 2002.

[Pisanelli 98] DM Pisanelli, A Gangemi & G Steve. *An ontological analysis of the UMLS Methathesaurus.* Proc AMIA Symp, pages 810–814, 1998.

[Rahm 01] E. Rahm & PA. Bernstein. *A survey of approaches to automatic schema matching.* VLDB J., vol. 10, no. 4, pages 334–350, 2001.

[Rahm 04] E. Rahm, HH. Do & S. Massmann. *Matching Large XML Schemas.* SIGMOD Record, vol. 33, no. 4, pages 26–31, 2004.

[Rajpathak 01] D. Rajpathak, E. Motta & R Roy. *A Generic Task Ontology for Scheduling Applications.* In Proceedings of the International Conference on Artificial Intelligence'2001 (IC-AI'2001), pages 1037–1043, 2001.

[Rasmussen 92] E. Rasmussen. *Clustering algorithms.* Information retrieval : data structures and algorithms, pages 419–442, 1992.

[Rebhan 98] M. Rebhan, V. Chalifa-Caspi, J. Prilusky & D. Lancet. *GeneCards : a novel functional genomics compendium with automated data mining and query reformulation support.* Bioinformatics, vol. 14, no. 8, pages 656–664, 1998.

[Reinoso-Castillo 03] J. Reinoso-Castillo, A. Silvescu, D. Caragea, J. Pathak & V. Honavar. *Information Extraction and Integration from Heterogeneous, Distributed, Autonomous Information Sources : A Federated, Query-Centric Ap-*

proach. In IEEE International Conference on Information Integration and Reuse, pages 183–191, 2003.

[Rosen 03] N. Rosen, V. Chalifa-Caspi, O. Shmueli, A. Adato, M. Lapidot, J. Stampnitzky, M. Safran & D. Lancet. *GeneLoc : exon-based integration of human genome maps.* Bioinformatics, vol. 19, no. suppl_1, pages i222–224, 2003.

[Rosse 03] C. Rosse & JLV. Mejino Jr. *A reference ontology for biomedical informatics : the foundational model of anatomy.* J. of Biomedical Informatics, vol. 36, no. 6, pages 478–500, 2003.

[Rousset 02] MC. Rousset, A. Bidault, C. Froidevaux, H. Gagliardi, F. Goasdoué, C. Reynaud & B. Safar. *Construction de Médiateurs pour Intégrer des Sources d'Information Multiples et Hétérogènes : le projet PICSEL.* Revue I3, vol. 2, pages 09–59, 2002.

[Safran 02] M. Safran, I. Solomon, O. Shmueli, M. Lapidot, S. Shen-Orr, A. Adato, U. Ben-Dor, N. Esterman, N. Rosen, I. Peter, T. Olender, V. Chalifa-Caspi & D. Lancet. *GeneCards 2002 : towards a complete, object-oriented, human gene compendium.* Bioinformatics, vol. 18, no. 11, pages 1542–1543, Nov 2002.

[Safran 03] M. Safran, V. Chalifa-Caspi, O. Shmueli, T. Olender, M. Lapidot, N. Rosen, M. Shmoish, Y. Peter, G. Glusman, E. Feldmesser, A. Adato, I. Peter, M. Khen, T. Atarot, Y. Groner & D. Lancet. *Human Gene-Centric Databases at the Weizmann Institute of Science : GeneCards, UDB, CroW 21 and HORDE.* Nucl. Acids Res., vol. 31, no. 1, pages 142–146, 2003.

[Schonbach 00] C. Schonbach, P. Kowalski-Saunders & V. Brusic. *Data warehousing in molecular biology.* Brief Bioinform, vol. 1, no. 2, pages 190–198, 2000.

[Schuler 96] GD. Schuler, JA. Epstein, H. Ohkawa & JA. Kans. *Entrez : molecular biology database and retrieval system.* Methods Enzymol, vol. 266, pages 141–162, 1996.

[Schulz 05] S. Schulz & U. Hahn. *Part-whole representation and reasoning in formal biomedical ontologies.* Artificial Intelligence in Medicine, vol. 34, no. 3, pages 179–200, 2005.

[Schulz 06] S. Schulz, A. Kumar & T. Bittner. *Biomedical ontologies : What part-of is and isn't.* J Biomed Inform, vol. 3(9), pages 350–361, 2006.

[Schulze-Kremer 02] S. Schulze-Kremer. *Ontologies for molecular biology and bioinformatics.* In Silico Biol, vol. 2, no. 3, pages 179–193, 2002.

[Shaker 02] R. Shaker, P. Mork, M. Barclay & P. Tarczy-Hornoch. *A rule driven bidirectional translation system remapping queries and result sets between a mediated schema and heterogeneous data sources.* Proc AMIA Symp, pages 692–696, 2002.

[Shvaiko 05] P. Shvaiko & J. Euzenat. *A Survey of Schema-Based Matching Approaches.* J. Data Semantics IV, pages 146–171, 2005.

[Smith 04] B. Smith, J. Köhler & A. Kumar. *On the Application of Formal Principles to Life Science Data : a Case Study in the Gene Ontology.* In DILS, pages 79–94, 2004.

[Smith 05] B Smith, W Ceusters, B Klagges, J Köhler, A Kumar, J Lomax, C Mungall, F Neuhaus, AL Rector & C Rosse. *Relations in biomedical ontologies.* Genome Biol, vol. 6, no. 5, page R46, 2005.

[Soderland 97] S. Soderland. *Learning to Extract Text-Based Information from the World Wide Web.* In Knowledge Discovery and Data Mining, pages 251–254, 1997.

[Soldatova 05] LN. Soldatova & RD. King. *Are the current ontologies in biology good ontologies ?* Nature Biotechnology, vol. 23, no. 9, pages 1095–1098, 2005.

[Stein 03] LD Stein. *Integrating biological databases.* Nat Rev Genet, vol. 4, no. 5, pages 337–345, May 2003.

[Stevens 00] R. Stevens, PG. Baker, S. Bechhofer, G. Ng, A. Jacoby, NW. Paton, CA. Goble & A. Brass. *TAMBIS : Transparent Access to Multiple Bioinformatics Information Sources.* Bioinformatics, vol. 16, no. 2, pages 184–186, 2000.

[Stuckenschmidt 01] H. Stuckenschmidt & F. van Harmelen. *Ontology-based metadata generation from semi-structured information.* In K-CAP '01 : Proceedings of the 1st international conference on Knowledge capture, pages 163–170, New York, NY, USA, 2001. ACM Press.

[Sujansky 01] W. Sujansky. *Heterogeneous database integration in biomedicine.* J Biomed Inform, vol. 34, no. 4, pages 285–298, August 2001.

[Teusan 03] R Teusan, A Bihouee, N Le Meur, G Ramstein & J. Leger. *MADTOOLS : management tools for the mining of microarray data. 'Details on MADSENSE, a gene comprehension support system'.* In European Conference on Computational Biology, Paris, 2003.

[Thierry-Mieg 06] D. Thierry-Mieg & J. Thierry-Mieg. *AceView : a comprehensive cDNA-supported gene and transcripts annotation.* Genome Biology, vol. 7, no. Suppl1, page S12, 2006.

[Van Rijsbergen 79] CJ. Van Rijsbergen. Information retrieval. Butterworth-Heinemann, Newton, MA, USA, 1979.

[Vargas Solar 02] G. Vargas Solar & A. Doucet. *Médiation de données : solutions et problèmes ouverts.* In Actes des 2èmes Assises nationales du GdR I3, Nancy, France, décembre 2002.

[Voorhees 94] EM. Voorhees. *Query Expansion Using Lexical-Semantic Relations.* In SIGIR, pages 61–69, 1994.

[Vossen 98] Piek Vossen, editeur. Eurowordnet : a multilingual database with lexical semantic networks. Kluwer Academic Publishers, Norwell, MA, USA, 1998.

[Wheeler 06] David L. Wheeler, Tanya Barrett, Dennis A. Benson, Stephen H. Bryant, Kathi Canese, Vyacheslav Chetvernin, Deanna M. Church, Michael DiCuccio, Ron Edgar, Scott Federhen, Lewis Y. Geer, Wolfgang Helmberg, Yuri Kapustin, David L. Kenton, Oleg Khovayko, David J. Lipman, Thomas L. Madden, Donna R. Maglott, James Ostell, Kim D. Pruitt, Gregory D. Schuler, Lynn M. Schriml, Edwin Sequeira, Stephen T. Sherry, Karl Sirotkin, Alexandre Souvorov, Grigory Starchenko, Tugba O. Suzek, Roman Tatusov, Tatiana A. Tatusova, Lukas Wagner & Eugene Yaschenko. *Database resources of the National Center for Biotechnology Information.* Nucl. Acids Res., vol. 34, no. Database Issue, pages D173–180, 2006.

[Whetzel 06] PL. Whetzel, H. Parkinson, HC. Causton, L. Fan, J. Fostel, G. Fragoso, L. Game, M. Heiskanen, N. Morrison, P. Rocca-Serra, SA. Sansone, C. Taylor, J. White & Jr Stoeckert CJ. *The MGED Ontology : a resource for semantics-based description of microarray experiments.* Bioinformatics, vol. 22, no. 7, pages 866–873, 2006.

[Widom 95] J. Widom. *Research Problems in Data Warehousing.* In 4th International Conference on Information and Knowledge Management, pages 25–30, Baltimore, Maryland, 1995.

[Wiederhold 92] G. Wiederhold. *Mediators in the Architecture of Future Information Systems.* IEEE Computer, vol. 25, no. 3, pages 38–49, 1992.

[Wroe 03] C. Wroe, R. Stevens, CA. Goble & M. Ashburner. *A Methodology to Migrate the Gene Ontology to a Description Logic Environment Using DAML+OIL.* In Pacific Symposium on Biocomputing, pages 624–635, 2003.

[Xu 03] L. Xu & DW. Embley. *Discovering Direct and Indirect Matches for Schema Elements.* In DASFAA, pages 39–46, 2003.

[Xyleme 01] L. Xyleme. *A dynamic warehouse for XML Data of the Web.* IEEE Data Eng. Bull., vol. 24, no. 2, pages 40–47, 2001.

[Zhang 04] S. Zhang & O. Bodenreider. *Comparing associative relationships among equivalent concepts across ontologies.* Medinfo, vol. 11, no. Pt 1, pages 459–466, 2004.

[Zhang 05] S. Zhang & O. Bodenreider. *Alignment of multiple ontologies of anatomy : Deriving indirect mappings from direct mappings to a reference.* Proc AMIA Symp, pages 864–868, 2005.

Liste des publications personnelles

Revues internationales

Mougin F., Burgun A., Bodenreider O. Mapping data elements to terminological resources for integrating biomedical data sources. BMC Bioinformatics. 2006 Nov 24 ;7 Suppl 3 :S6

Pouliquen B., Le Duff F., Delamarre D., Cuggia M., Mougin F., Le Beux P. (2005). Managing educational resource in medicine : system design and integration. Int J Med Inform. 2005 Mar ;74(2-4) :201-207

Cuggia M., Mougin F., Le Beux P. Indexing method of digital audiovisual medical resources with semantic Web integration. Int J Med Inform. 2005 Mar ;74(2-4) :169-77.

Conférences internationales avec comité de lecture et publication des actes

Mougin F., Burgun A., Bodenreider O. Using WordNet to Improve the Mapping of Data Elements to UMLS for Data Sources Integration, AMIA Annu Symp Proc. 2006 :574-578

Mougin F., Burgun A., Bodenreider O. Data integration through data elements : Mapping data elements to terminological resources, Proceedings of the Second International Symposium on Semantic Mining in Biomedicine (SMBM-2006), 2006 ;52-59

Alecu I., Bousquet C., Mougin F., Jaulent MC. Mapping of the WHO-ART terminology on Snomed CT to improve grouping of related adverse drug reactions. Stud Health Technol Inform, 2006 ;124 :833-838

Garcelon N., Mougin F., Bousquet C., Burgun A. Evidence in pharmacovigilance : extracting Adverse Drug Reactions articles from MEDLINE to link them to case databases. Stud Health Technol Inform, 2006 ;124 :528-533

Mougin F., Bodenreider O., Approaches to eliminating cycles in the UMLS Metathesaurus : Naïve vs. formal. AMIA Annu Symp Proc. 2005 ;550-554

Burgun A., Bodenreider O., Mougin F. Classifying diseases with respect to anatomy : a study in SNOMED CT. AMIA Annu Symp Proc. 2005 ;91-95

Bertaud V., Lasbleiz J., Mougin F., Marin F., Burgun A., Duvauferrier R. Toward a unified representation of findings in clinical radiology. Stud Health Technol Inform. 2005 ;116 :671-6

Mougin F., Burgun A., Loréal O., Le Beux P. Towards the automatic generation of biomedical sources schema, Medinfo, 2004. 2004 ;783-787

Mougin F., Cuggia M., Le Beux P. Development of an indexing search engine for the UMVF : Proposal for an indexing method based on Dublin Core and XML, Stud Health Technol Inform, 2003 ;95 :727-731

Conférences nationales

Mougin F., Burgun A., Bodenreider O., Méthodes pour résoudre lhétérogénéité sémantique des « data elements » de sources de données biomédicales pour leur intégration, Workshop Ontologie, Grille et intégration Sémantique pour la Biologie (OGSB), JOBIM, 4 juillet 2006, Bordeaux

Mougin F., Bodenreider O. Éliminer les cycles dans les systèmes terminologiques : comparaison de deux approches, Actes du congrès francophone Reconnaissance des Formes et Intelligence Artificielle (RFIA), 2006

Mougin F., Golbreich C., Burgun A., Le Beux P. Vers une génération automatique du mapping de sources biomédicales, Deuxième Journée Web Sémantique Médical, 9 mars 2004, Rouen

Mougin F., Le Duff F., Le Beux P., Optimisation des Recherches Bibliographiques MedLine Pour le Non Expert à partir d'un Outil d'aide à la Navigation dans l'Indexation MeSH, Internet et Pédagogie Médicale, Marseille, 11-12 dec 2003

Mougin F., Burgun A., Le Beux P. Les métadonnées dans le cadre du Web sémantique : applications au domaine biomédical, Première Journée Web Sémantique Médical, 17 mars 2003, Rennes

Mougin F., Cuggia M., Le Duff F., Kamendje B., Le Beux P. Proposition de méthode pour l'indexation des ressources documentaires de l'UMVF, Internet et Pédagogie Médicale, Lille 28-29 nov 2002

Posters

Mougin F., Burgun A., Loréal O., Le Beux P. A UMLS-based approach to resolve biomedical sources heterogeneities, Actes de JOBIM 2005, Lyon

Mougin F., Marquet G., Burgun A., Guérin E., Moussouni F. and Loréal O. Use of metadata for biomedical heterogeneous data sources integration, in Proceedings of the 2nd European Conference on Computational Biology, September 27-30, 2003, Paris, FRANCE

Glossaire

ADN	Acide désoxyribonucléique, 7
API	Application Program Interface ou Interface de programmation, 58
ARN	Acide ribonuléique, 7
DTD	Document Type Definition - grammaire permettant de vérifier la conformité du document XML qui lui est rattaché, 39
ED	Élément de données, unité d'information de base construite sur des structures standard ayant un sens unique et des valeurs distinctes, 98
Génomique fonctionnelle	Étude et analyse directe du transcriptome et du protéome : elle vise à déterminer la fonction des gènes à partir de leurs produits d'expression (ARN et protéines), ainsi qu'à étudier leur mode de régulation et leurs interactions, 7
Northern Blot	Technique employée en biologie moléculaire pour étudier l'expression de gènes, 118
OWL	Web Ontology Language, 17
Parseur	Algorithme qui permet d'analyser un texte et d'en déterminer sa structure syntaxique afin d'effectuer divers traitements, comme par exemple une compilation, 45

PCR	Polymerase Chain Reaction - Technique d'amplification d'un segment d'ADN in vitro par la Taq polymerase (enzyme travaillant à haute température) en présence de deux amorces spécifiques et de nucléotides, 138
Protéome	Ensemble des protéines exprimé par le génome d'une espèce donnée. Il assure le développement, la croissance et le fonctionnement de la cellule (donc de l'organisme), 7
RDF	Resource Description Framework, 17
Ressource	Entité informatique (document électronique, image, service, collection d'autres ressources, etc) ayant une identité, 9
RT-PCR	Reverse Transcriptase Polymerase Chain Reaction - Technique d'amplification d'un segment d'ARN par adjonction de la transcriptase revers à la PCR, 138
SNP	Polymorphism, Single Nucleotide - il désigne des variations ponctuelles de 1 paire de base sur 1000, 138
Transcriptome	Ensemble des ARN messagers transcrits à partir du génome, 7
UMLS	Unified Medical Language System, 86
URL	Uniform Resource Locator - chaîne de caractères indiquant l'emplacement d'une ressource sur Internet et la méthode permettant d'y accéder, 12
WN	WordNet, 95
XML	eXtensible Markup Language, 8

Annexes

Annexe A : Les deux hiérarchies des types sémantiques de l'UMLS

Entity
- Physical Object
  - Organism
    - Plant
      - Alga
    - Fungus
    - Virus
    - Rickettsia or Chlamydia
    - Bacterium
    - Archaeon
    - Animal
      - Invertebrate
      - Vertebrate
        - Amphibian
        - Bird
        - Fish
        - Reptile
        - Mammal
          - Human
  - Anatomical Structure
    - Embryonic Structure
    - Anatomical Abnormality
      - Congenital Abnormality
      - Acquired Abnormality
    - Fully Formed Anatomical Structure
      - Body Part, Organ, or Organ Component
      - Tissue
      - Cell
      - Cell Component
      - Gene or Genome
  - Manufactured Object
    - Medical Device
    - Research Device
    - Clinical Drug

[Entity] (continued)
- [Physical Object] (continued)
  - Substance
    - Chemical
      - Chemical Viewed Functionally
        - Pharmacologic Substance
          - Antibiotic
        - Biomedical or Dental Material
        - Biologically Active Substance
          - Neuroreactive Substance or Biogenic Amine
          - Hormone
          - Enzyme
          - Vitamin
          - Immunologic Factor
          - Receptor
        - Indicator, Reagent, or Diagnostic Aid
        - Hazardous or Poisonous Substance
      - Chemical Viewed Structurally
        - Organic Chemical
          - Nucleic Acid, Nucleoside, or Nucleotide
          - Organophosphorus Compound
          - Amino Acid, Peptide, or Protein
          - Carbohydrate
          - Lipid
            - Steroid
            - Eicosanoid
        - Inorganic Chemical
        - Element, Ion, or Isotope
    - Body Substance
    - Food

[Entity] (continued)
- Conceptual Entity
  - Idea or Concept
    - Temporal Concept
    - Qualitative Concept
    - Quantitative Concept
    - Functional Concept
      - Body System
    - Spatial Concept
      - Body Space or Junction
      - Body Location or Region
      - Molecular Sequence
        - Nucleotide Sequence
        - Amino Acid Sequence
        - Carbohydrate Sequence
      - Geographic Area
  - Finding
    - Laboratory or Test Result
    - Sign or Symptom
  - Organism Attribute
    - Clinical Attribute
  - Intellectual Product
    - Classification
    - Regulation or Law
  - Language
  - Occupation or Discipline
    - Biomedical Occupation or Discipline
  - Organization
    - Health Care Related Organization
    - Professional Society
    - Self help or Relief Organization
  - Group Attribute
  - Group
    - Professional or Occupational Group
    - Population Group
    - Family Group
    - Age Group
    - Patient or Disabled Group

Event
- Activity
  - Behavior
    - Social Behavior
    - Individual Behavior
  - Daily or Recreational Activity
  - Occupational Activity
    - Health Care Activity
      - Laboratory Procedure
      - Diagnostic Procedure
      - Therapeutic or Preventive Procedure
    - Research Activity
      - Molecular Biology Research Technique
    - Governmental or Regulatory Activity
    - Educational Activity
  - Machine Activity
- Phenomenon or Process
  - Human caused Phenomenon or Process
    - Environmental Effect of Humans
  - Natural Phenomenon or Process
    - Biologic Function
      - Physiologic Function
        - Organism Function
          - Mental Process
        - Organ or Tissue Function
        - Cell Function
        - Molecular Function
          - Genetic Function
        - Pathologic Function
          - Disease or Syndrome
            - Mental or Behavioral Dysfunction
            - Neoplastic Process
          - Cell or Molecular Dysfunction
          - Experimental Model of Disease
  - Injury or Poisoning

FIG. 8.1 – **Liste des types sémantiques de l'UMLS suivant les deux hiérarchies.**

Annexe B : Hiérarchie des domaines de WordNet

TOP LEVEL
–> doctrines
–> free_time
–> applied_science
–> pure_science
–> social_science
–> factotum
——> number
——> color
——> time_period
——> person
——> quality
–> metrology
FACTOTUM
factotum
HIERARCHY : DOCTRINES
doctrines
–> archaeology
–> astrology
–> history
–> linguistics
–> literature
–> philosophy
–> psychology
–> art
–> religion
archaeology
art
–> dance
–> drawing
–> music
–> photography
–> plastic_arts
–> theatre
drawing
–> painting
–> philately
astrology
history
–> heraldry
linguistics

–> grammar
literature
–> philology
philosophy
plastic_arts
–> jewellery
–> numismatics
–> sculpture
psychology
–> psychoanalysis
religion
–> mythology
–> occultism
–> theology
HIERARCHY : FREE_TIME
free_time
–> play
–> sport
play
–> betting
–> card
–> chess
betting
card
sport
–> badminton
–> baseball
–> basketball
–> cricket
–> football
–> golf
–> rugby
–> soccer
–> table_tennis
–> tennis
–> volleyball
–> cycling
–> skating
–> skiing
–> hockey
–> mountaineering
–> rowing
–> swimming

–> sub
–> diving
–> racing
–> athletics
–> wrestling
–> boxing
–> fencing
–> archery
–> fishing
–> hunting
–> bowling
racing
HIERARCHY : APPLIED_SCIENCE
applied_science
–> agriculture
–> alimentation
–> architecture
–> computer_science
–> engineering
–> medicine
–> veterinary
alimentation
–> gastronomy
architecture
–> town_planning
–> building_industry
–> furniture
engineering
–> mechanics
–> astronautics
–> electrotechnics
–> hydraulics
mechanics
medicine
–> dentistry
–> pharmacy
–> psychiatry
–> radiology
–> surgery
veterinary
–> zootechnics
HIERARCHY : PURE_SCIENCE
pure_science

–> astronomy
–> biology
–> chemistry
–> earth
–> mathematics
–> physics
astronomy
–> topography
biology
–> biochemistry
–> ecology
–> botany
–> zoology
–> anatomy
–> physiology
–> genetics
botany
chemistry
earth
–> geology
–> meteorology
–> oceanography
–> paleontology
–> geography
mathematics
–> geometry
physics
–> acoustics
–> atomic_physic
–> electricity
–> optics
zoology
–> entomology
geography
HIERARCHY : SOCIAL_SCIENCE
social_science
–> administration
–> anthropology
–> artisanship
–> body_care
–> commerce
–> economy
–> fashion

–> industry
–> law
–> military
–> pedagogy
–> politics
–> publishing
–> sexuality
–> sociology
–> telecommunication
–> tourism
–> transport
administration
anthropology
–> ethnology
ethnology
–> folklore
artisanship
body_care
commerce
economy
–> banking
–> book_keeping
–> enterprise
–> exchange
–> insurance
–> money
–> tax
fashion
industry
law
–> state
military
pedagogy
–> school
–> university
politics
–> diplomacy
publishing
sexuality
sociology
telecommunication
–> cinema
–> post

–> radio
–> telegraphy
–> telephony
–> tv
tourism
transport
–> aeronautic
–> auto
–> merchant_navy
–> railway

Annexe C : Liste des symboles de gènes et noms associés constituant notre corpus nécessaire permettant d'interroger les sources biomédicales

ADAMTS2||a disintegrin-like and metalloprotease (reprolysin type) with thrombospondin type 1 motif, 2

ALAD||aminolevulinate, delta-, dehydratase

ALAS1||aminolevulinate, delta-, synthase 1

ALAS2||aminolevulinate, delta-, synthase 2 (sideroblastic/hypochromic anemia)

ALS2||amyotrophic lateral sclerosis 2 (juvenile)

APOE||apolipoprotein E

APP||amyloid beta (A4) precursor protein (protease nexin-II, Alzheimer disease)

ARHI||ras homolog gene family, member I

ASPA||aspartoacylase (aminoacylase 2, Canavan disease)

ATM||ataxia telangiectasia mutated (includes complementation groups A, C and D)

ATP7B||ATPase, Cu++ transporting, beta polypeptide (Wilson disease)

BRCA1||breast cancer 1, early onset

BRCA2||breast cancer 2, early onset

CFTR||cystic fibrosis transmembrane conductance regulator, ATP-binding cassette (subfamily C, member 7)

CHEK2||CHK2 checkpoint homolog (S. pombe)

CMT4B2||Charcot-Marie-Tooth neuropathy 4B2 (autosomal recessive, with myelin outfolding)

COL11A1||collagen, type XI, alpha 1

COL11A2||collagen, type XI, alpha 2

COL1A1||collagen, type I, alpha 1

COL1A2||collagen, type I, alpha 2

COL2A1||collagen, type II, alpha 1 (primary osteoarthritis, spondyloepiphyseal dysplasia, congenital)

COL3A1||collagen, type III, alpha 1 (Ehlers-Danlos syndrome type IV, autosomal dominant)

COL5A1||collagen, type V, alpha 1

COL5A2||collagen, type V, alpha 2

CPO||coproporphyrinogen oxidase (coproporphyria, harderoporphyria)

DMPK||dystrophia myotonica-protein kinase

EGR2||early growth response 2 (Krox-20 homolog, Drosophila)

ERBB2||v-erb-b2 erythroblastic leukemia viral oncogene homolog 2, neuro/glioblastoma derived oncogene homolog (avian)

FBN1||fibrillin 1 (Marfan syndrome)

FECH||ferrochelatase (protoporphyria)

FGFR1||fibroblast growth factor receptor 1 (fms-related tyrosine kinase 2, Pfeiffer syndrome)

FGFR2||fibroblast growth factor receptor 2 (bacteria-expressed kinase, keratinocyte growth factor receptor, craniofacial dysostosis 1, Crouzon syndrome, Pfeiffer syndrome, Jackson-Weiss syndrome)

FGFR3||fibroblast growth factor receptor 3 (achondroplasia, thanatophoric dwarfism)

FGFR4||fibroblast growth factor receptor 4

FGFRL1||fibroblast growth factor receptor-like 1

GARS||glycyl-tRNA synthetase

GBA||glucosidase, beta ; acid (includes glucosylceramidase)

GCH1||GTP cyclohydrolase 1 (dopa-responsive dystonia)

GDAP1||ganglioside-induced differentiation-associated protein 1

GJB1||gap junction protein, beta 1, 32kDa (connexin 32, Charcot-Marie-Tooth neuropathy, X-linked)

HAMP||hepcidin antimicrobial peptide

HBB||hemoglobin, beta

HD||huntingtin (Huntington disease)

HEXA||hexosaminidase A (alpha polypeptide)

HFE||hemochromatosis

HMBS||hydroxymethylbilane synthase

HRAS||v-Ha-ras Harvey rat sarcoma viral oncogene homolog

KIF1B||kinesin family member 1B

LITAF||lipopolysaccharide-induced TNF factor

MPZ||myelin protein zero (Charcot-Marie-Tooth neuropathy 1B)

MTMR2||myotubularin related protein 2

NDRG1||N-myc downstream regulated gene 1

NEF3||neurofilament 3 (150kDa medium)

NEFH||neurofilament, heavy polypeptide 200kDa

NEFL||neurofilament, light polypeptide 68kDa

NF1||neurofibromin 1 (neurofibromatosis, von Recklinghausen disease, Watson disease)

NF2||neurofibromin 2 (bilateral acoustic neuroma)

PAH||phenylalanine hydroxylase

PCBD||6-pyruvoyl-tetrahydropterin synthase/dimerization cofactor of hepatocyte nuclear factor 1 alpha (TCF1)

PLOD||procollagen-lysine, 2-oxoglutarate 5-dioxygenase (lysine hydroxylase, Ehlers-Danlos syndrome type VI)

PMP22||peripheral myelin protein 22

PPOX||protoporphyrinogen oxidase

PPP1R12A||protein phosphatase 1, regulatory (inhibitor) subunit 12A

PRX||periaxin

PSEN1||presenilin 1 (Alzheimer disease 3)

PSEN2||presenilin 2 (Alzheimer disease 4)

PTEN||phosphatase and tensin homolog (mutated in multiple advanced cancers 1)

PTS||6-pyruvoyltetrahydropterin synthase

QDPR||quinoid dihydropteridine reductase

RAB7||RAB7, member RAS oncogene family

RAD51||RAD51 homolog (RecA homolog, E. coli) (S. cerevisiae)

RB1||Retinoblastoma 1 (including osteosarcoma)

SERPINA1||serine (or cysteine) proteinase inhibitor, clade A (alpha-1 antiproteinase, antitrypsin), member 1

SLC40A1||solute carrier family 40 (iron-regulated transporter), member 1

SOD1||superoxide dismutase 1, soluble (amyotrophic lateral sclerosis 1 (adult))

STK11||serine/threonine kinase 11 (Peutz-Jeghers syndrome)

TFR2||transferrin receptor 2

TNXB||tenascin XB

TP53||tumor protein p53 (Li-Fraumeni syndrome)

TSC1||tuberous sclerosis 1

TSC2||tuberous sclerosis 2

UROD||uroporphyrinogen decarboxylase

UROS||uroporphyrinogen III synthase (congenital erythropoietic porphyria)

ZNF9||zinc finger protein 9 (a cellular retroviral nucleic acid binding protein)

LMLN||leishmanolysin-like (metallopeptidase M8 family)

DHPS||deoxyhypusine synthase

CA8||carbonic anhydrase VIII

LENG1||leukocyte receptor cluster (LRC) member 1

CYP3A||cytochrome P450, family 3, subfamily A

TRBC1||T cell receptor beta constant 1

TUBBP5||tubulin, beta polypeptide pseudogene 5

IGHV3-30-2||immunoglobulin heavy variable 3-30-2

C8orf15||chromosome 8 open reading frame 15

C22orf13||chromosome 22 open reading frame 13

MPE||malignant proliferation, eosinophil

C18orf12||chromosome 18 open reading frame 12

MOSPD1||motile sperm domain containing 1

IGHV3-76||immunoglobulin heavy variable 3-76

OCA2||oculocutaneous albinism II (pink-eye dilution homolog, mouse)

DCP1A||decapping enzyme

Annexe D : Liste des éléments de données extraits de la source Aceview

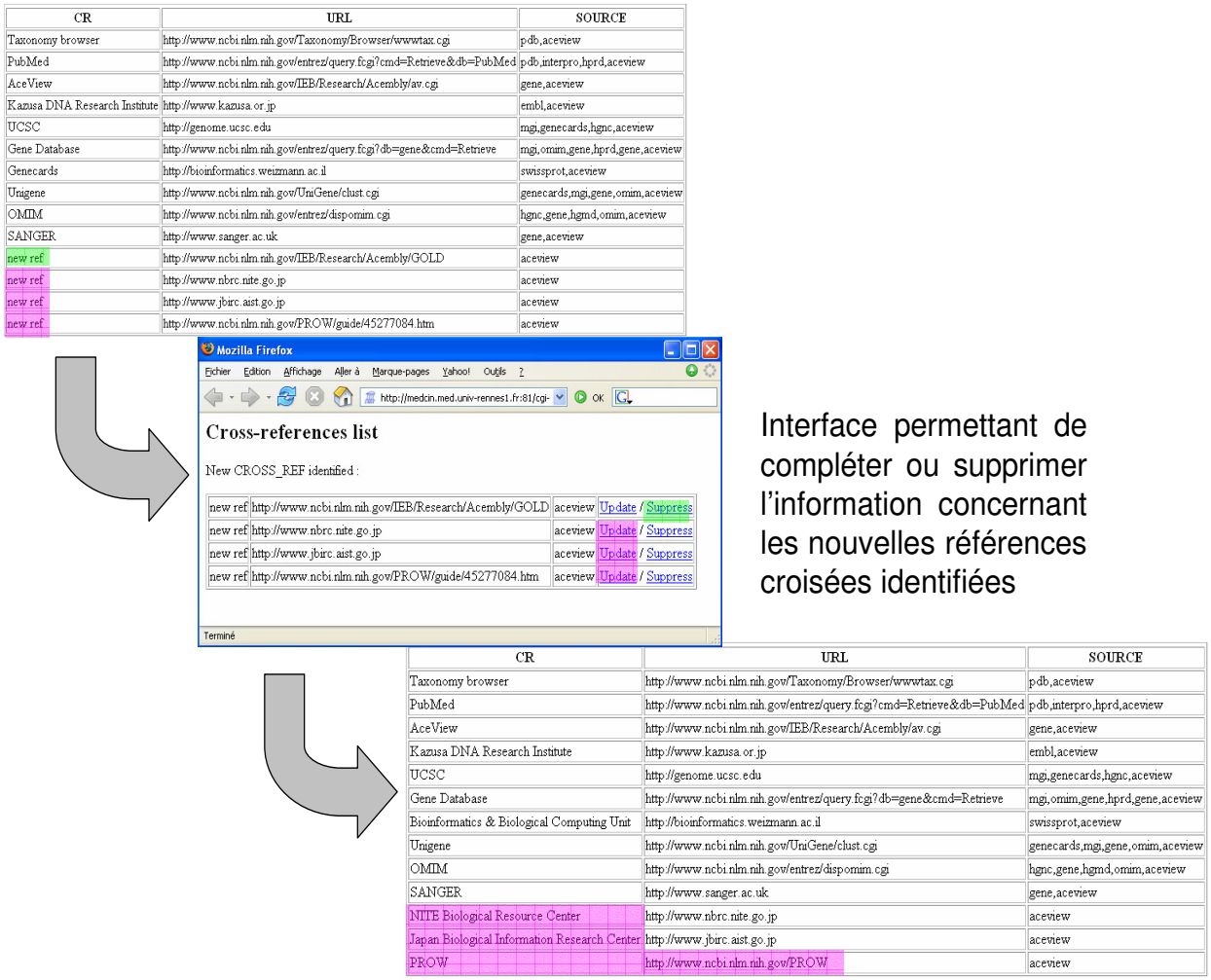
Liste des éléments de données extraits de la source Aceview

Alias	Liver and Spleen
All supporting clones for gene symb	Localisation
Alternative exon	M28668 NM 000492
Alternative features	Met to Stop
Annotation of variants	Molecular properties
Anomalies	mRNA gap
ATP binding	mRNA variant
Bibliography	Neuroblastoma Cot 25-normalized
Bibliography, abstracts and RIFs	Overview
cDNA clones	Phenotype
Completeness	Phenotype and Function
COOH complete	Placenta Cot 25-normalized
Coordinates on gene	Product
Description	Protein
Description of the protein family	Proteins
Diagram	Psort
EC number	Regulation
Expression and regulation	Sequence links to clone
Expression level and number of variants	Sequencing gap
Extends from	Source
Full Page	Supporting clone
Function	Tiling clones for gene symb
Functional annotation	Tissue
Gene and transcript	Transcription unit
Intron exon structure and support	Transcripts and sequences
Introns	Transcripts, proteins and sequences
Introns and exons	Type
Length & DNA	Variant
Links	

Alias	Liver and Spleen
All supporting clones for gene symb	Localisation
Alternative exon	Met to Stop
Alternative features	Molecular properties
Annotation of variants	mRNA gap
Anomalies	mRNA variant
ATP binding	Neuroblastoma Cot 25-normalized
Bibliography	Overview
Bibliography, abstracts and RIFs	Phenotype
cDNA clones	Phenotype and Function
Completeness	Placenta Cot 25-normalized
COOH complete	Product
Coordinates on gene	Protein
Description	Proteins
Description of the protein family	Psort
Diagram	Regulation
EC number	Sequence links to clone
Expression and regulation	Sequencing gap
Expression level and number of variants	Source
Extends from	Supporting clone
Function	Tiling clones for gene symb
Functional annotation	Tissue
Gene and transcript	Transcription unit
Intron exon structure and support	Transcripts and sequences
Introns	Transcripts, proteins and sequences
Introns and exons	Type
Length & DNA	Variant
Links	

FIG. 8.2 – Liste des 57 éléments de données extraits de la source Aceview (image de gauche) et liste des 54 éléments de données obtenus à partir de la source Aceview après filtrage (image de droite). Les trois éléments en rouge ont été supprimés.

Validation des références croisées identifiées dans la source Aceview.



CR	URL	SOURCE
Taxonomy browser	http://www.ncbi.nlm.nih.gov/Taxonomy/Browser/wwwtax.cgi	pdb,aceview
PubMed	http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?cmd=Retrieve&db=PubMed	pdb,interpro,hprd,aceview
AceView	http://www.ncbi.nlm.nih.gov/IEB/Research/Acembly/av.cgi	gene,aceview
Kazusa DNA Research Institute	http://www.kazusa.or.jp	embl,aceview
UCSC	http://genome.ucsc.edu	mgi,genecards,hgnc,aceview
Gene Database	http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=gene&cmd=Retrieve	mgi,omim,gene,hprd,gene,aceview
Genecards	http://bioinformatics.weizmann.ac.il	swissprot,aceview
Unigene	http://www.ncbi.nlm.nih.gov/UniGene/clust.cgi	genecards,mgi,gene,omim,aceview
OMIM	http://www.ncbi.nlm.nih.gov/entrez/dispomim.cgi	hgnc,gene,hgmd,omim,aceview
SANGER	http://www.sanger.ac.uk	gene,aceview
new ref	http://www.ncbi.nlm.nih.gov/IEB/Research/Acembly/GOLD	aceview
new ref	http://www.nbrc.nite.go.jp	aceview
new ref	http://www.jbirc.aist.go.jp	aceview
new ref	http://www.ncbi.nlm.nih.gov/PROW/guide/45277084.htm	aceview

CR	URL	SOURCE
Taxonomy browser	http://www.ncbi.nlm.nih.gov/Taxonomy/Browser/wwwtax.cgi	pdb,aceview
PubMed	http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?cmd=Retrieve&db=PubMed	pdb,interpro,hprd,aceview
AceView	http://www.ncbi.nlm.nih.gov/IEB/Research/Acembly/av.cgi	gene,aceview
Kazusa DNA Research Institute	http://www.kazusa.or.jp	embl,aceview
UCSC	http://genome.ucsc.edu	mgi,genecards,hgnc,aceview
Gene Database	http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=gene&cmd=Retrieve	mgi,omim,gene,hprd,gene,aceview
Bioinformatics & Biological Computing Unit	http://bioinformatics.weizmann.ac.il	swissprot,aceview
Unigene	http://www.ncbi.nlm.nih.gov/UniGene/clust.cgi	genecards,mgi,gene,omim,aceview
OMIM	http://www.ncbi.nlm.nih.gov/entrez/dispomim.cgi	hgnc,gene,hgmd,omim,aceview
SANGER	http://www.sanger.ac.uk	gene,aceview
NITE Biological Resource Center	http://www.nbrc.nite.go.jp	aceview
Japan Biological Information Research Center	http://www.jbirc.aist.go.jp	aceview
PROW	http://www.ncbi.nlm.nih.gov/PROW	aceview

FIG. 8.3 – Parmi les 14 références croisées identifiées (fenêtre en haut à gauche), quatre d'entre elles sont nouvelles (surlignées en rose ou vert). Ces dernières nécessitent d'être validées ou supprimées et cela est possible au travers de l'interface développée à cet effet (fenêtre centrale). Pour chaque référence croisée, il est possible de la supprimer ou d'en modifier le nom et/ou l'URL. La référence surlignée en vert est supprimée car non pertinente et les trois autres (en rose) sont gardées et complétées de leur réel nom. De plus, la dernière référence présentait une URL qui ne correspondait pas à la page d'accueil de la source référencée et nous avons modifié son URL en conséquence.

Annexe E : Exemple de requête de la forme : *Recherche des informations concernant le locus du gène ayant pour symbole « HAMP »*

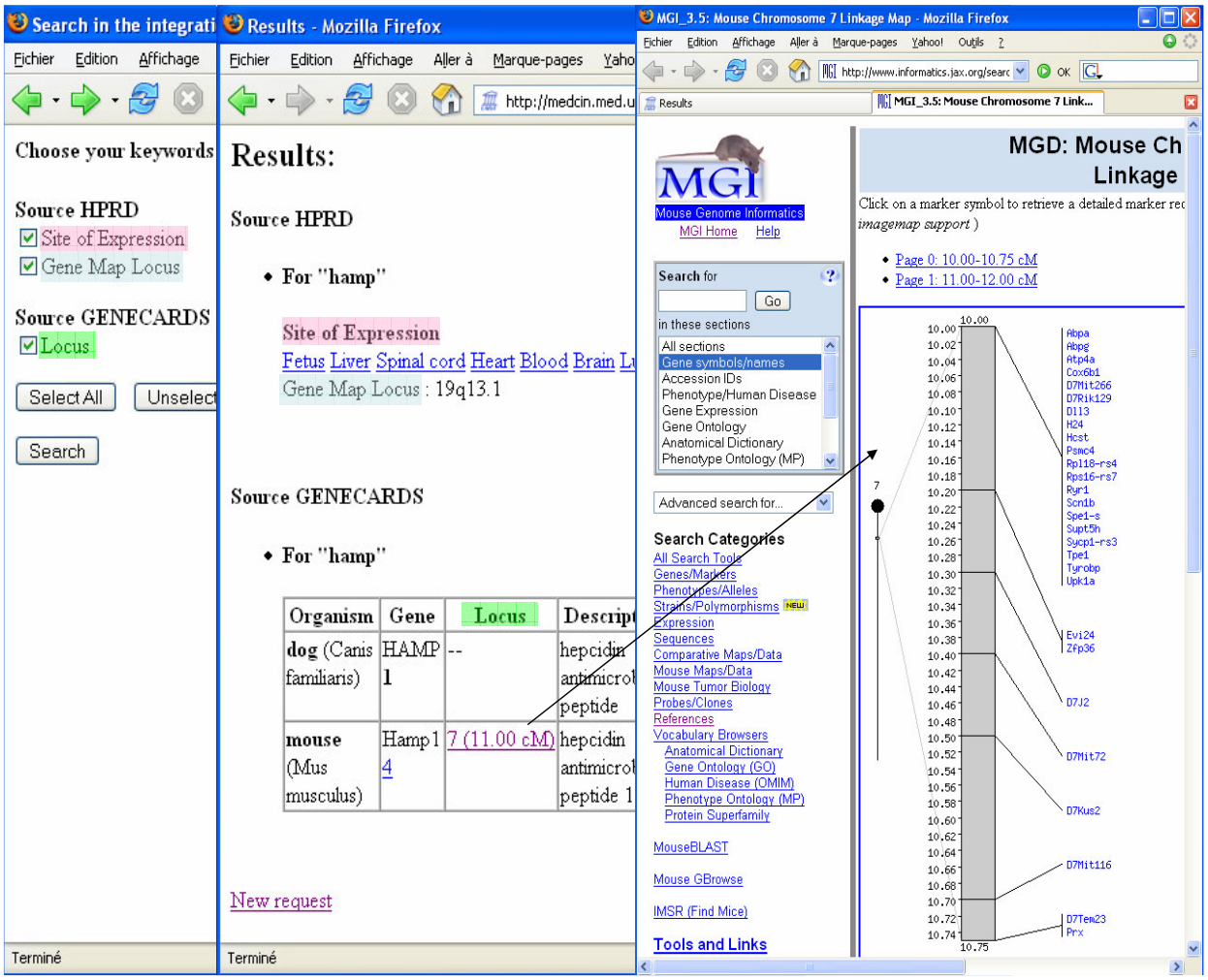


FIG. 8.4 – À partir de cette requête, notre système tel qu'il est disponible actuellement propose trois éléments de données pouvant fournir des résultats intéressants (page de gauche). Sur la page du milieu, on constate d'abord que dans HPRD, il est possible d'accéder aux sites d'expression du gène considéré (ce qui n'est pas vraiment pertinent par rapport à la requête initiale) ainsi qu'au locus du gène. Dans GeneCards, des informations concernant le locus du gène « HAMP » dans d'autres organismes sont accessibles. Un lien vers MGD permet de visualiser ce gène sur le(s) chromosome(s) concernés chez la souris (page de droite).

Annexe F : Exemples pour l'amélioration du processus de requêtes

Nous présentons deux exemples pour chacune des perspectives proposées pour améliorer le processus de requêtes de notre système.

Exploitation des propriétés.

Pour l'exploitation des propriétés, une solution est d'utiliser les relations entre les types sémantiques de l'UMLS. Par exemple, il existe une relation du type *carries_out* entre les types sémantiques Gene Or Genome et Molecular Function. Une propriété *carries_out* (de type ObjectProperty) ayant pour domaine une entité biologique et pour co-domaine une fonction biologique pourrait être créée. Il suffit ensuite de compléter la description de la classe associée au type sémantique Gene Or Genome en lui ajoutant la propriété *carries_out* restreinte à la classe correspondant au type sémantique Molecular Function. Une requête de la forme *Functions carried out by the gene « beta-2-microglobulin »* sera donc traitée comme suit : le médiateur ne récupérera que les concepts descendants du type sémantique Molecular Function et non pas des autres types sémantiques descendants du type sémantique Biologic Function (comme par exemple Organism Function, Mental Process, Organ or Tissue Function et Pathologic Function). Cela permettra ainsi de traiter les requêtes de manière plus spécifique et fournir des résultats plus précis aux utilisateurs (les fonctions moléculaires exécutées par les gènes dans notre exemple).

Exploitation des références croisées.

En pratique, on pourrait mettre en œuvre cette perspective de la façon suivante. D'abord, le processus se chargerait de déterminer les EDs pouvant répondre à la requête (comme cela est implémenté actuellement) et dans un deuxième temps, une analyse de la forme de la phrase devrait être mise en place afin de savoir dans quel ordre utiliser ces EDs. Par exemple, pour une requête de la forme *Search for the structure of the enzyme involved in the pathology named « Porphyria cutanea tarda »*, il faudrait d'abord déterminer l'enzyme recherché puis en repérer la structure. On chercherait donc d'abord les valeurs des EDs associés au mot « enzyme ». Notre système identifierait trois EDs pertinents : `Catalytic activity` (synonyme « enzyme activity ») et `Enzyme Number`, extraits de GeneCards, ainsi que `Enzyme IDs` issu de la source HGNC. `Enzyme Number` fournit l'identifiant de l'enzyme concerné : c'est l'« uroporphyrinogen decarboxylase » (EC 4.1.1.37) et un lien vers la page le décrivant dans la source IUBMB Enzyme Nomenclature est disponible[1]. À ce stade, on ne disposerait pas encore de l'information voulue puisque c'est plus précisément la structure moléculaire de cet enzyme qui est recherchée. Il faudrait donc trouver une référence croisée dans la page IUBMB Enzyme Nomenclature vers une source intégrée à notre système et qui puisse fournir ce type d'informations. Une contrainte pourrait donc être ajoutée : ce lien ne devra concerner que des sources dont au moins un ED concerne la structure moléculaire. Parmi les douze sources de notre système, sept d'entre elles

[1] http ://www.chem.qmul.ac.uk/iubmb/enzyme/EC4/1/1/37.html

peuvent potentiellement répondre à cette contrainte : Entrez Gene et OMIM chacun avec l'ED `Structure`, Swiss-Prot ayant comme ED `3D structure databases`, GeneCards avec l'ED `3D structures`, Aceview dont l'ED est `Intron exon structure and support` et enfin PDB avec l'ED `Structure Summary`. Or, dans la page référencée par GeneCards (source IUBMB Enzyme Nomenclature), seule la source PDB apparaît et comme cette source fait partie de l'ensemble des sources identifiées comme pouvant fournir des données concernant la structure moléculaire d'une entité biologique, c'est ce lien qui serait suivi et donc la page correspondante qui serait proposée aux utilisateurs. On notera que d'autres sources telles que KEGG sont aussi proposées mais comme elles n'existent pas dans notre système, nous ne pouvons pas être sûrs qu'elles fournissent des données intéressant les utilisateurs, raison pour laquelle nous ne les intégrerions pas au résultat.

Résumé

Ce travail de thèse s'inscrit dans la problématique du Web sémantique et plus précisément l'utilisation de ses technologies dans le domaine de la génomique fonctionnelle. Au cours de leurs travaux de recherche, les biologistes et médecins doivent disposer de données concernant l'existant et ces informations sont généralement accessibles sur Internet. Le problème est qu'elles sont réparties dans des sources distribuées, autonomes et hétérogènes à de multiples niveaux.

C'est dans ce cadre que nous proposons le développement d'un système d'intégration basée médiateur. Celui-ci vise à offrir aux biologistes et médecins une interface unique permettant d'accéder aux différentes sources de manière centralisée et homogène. Des systèmes suivant cette approche existent dans la littérature mais présentent des limites en terme de conception et d'évolution. En effet, de nombreuses tâches restent manuelles, ce qui les rend particulièrement fastidieuses.

Pour répondre à ces limites, nous avons conçu un système médiateur au moyen de méthodes semi-automatiques. Tout d'abord, nous avons développé une méthode d'acquisition automatique des schémas des sources de données à intégrer. Ensuite, nous avons utilisé une ressource terminologique existante, l'UMLS, afin de définir un schéma global. Enfin, nous avons proposé différentes approches pour mettre en correspondance les éléments des schémas locaux avec ceux du schéma global. Celles-ci sont situées aux niveaux *schéma* et *instances* et sont automatiques, avec une possible validation manuelle par des experts.

Le système implémenté à partir de ces méthodes est opérationnel et accessible sur Internet et permet de traiter des requêtes simples qui impliquent plusieurs sources de données. Le processus de requêtes effectue une expansion de celles-ci en exploitant la hiérarchie des concepts identifiés comme pertinents, et rend finalement aux utilisateurs les valeurs des éléments des sources associés à ces concepts. La maintenance du système est facilitée par les méthodes développées pour sa conception, permettant ainsi de la gérer de manière semi-automatique.

Mots clés : bioinformatique, Web sémantique, représentation des connaissances, bio-ontologies, système d'intégration, hétérogénéités syntaxiques et sémantiques, mises en correspondance de schémas.

Abstract

The general framework of this work is that of the Semantic Web and, more precisely, the application of Semantic Web technologies to functional genomics. The data biologists and physicians access for research purposes are typically available on the Internet. However, this information is also generally distributed over multiple autonomous, heterogeneous sources.

We propose a mediator-based approach to integrating biomedical information sources. Our system provides a unique interface for biologists and physicians to access data sources in a centralized and homogeneous way. The use of a mediator is not novel in biomedicine. But existing systems have limitations in terms of conception and evolution, in particular, the need for manual intervention at many levels.

To address these issues, we employed semi-automatic methods for creating a mediator-based system. First, we developed an automatic method for acquiring the schemas of the sources to be integrated. Then we used an existing terminological resource, the UMLS, to define a global schema. Finally, we proposed different approaches to mapping local schemas' elements to those of the global schema. These techniques operate at the *schema* and *instance* levels and are fully automated. Domain experts can also validate the integration process.

The system we created is operational and available on the Internet. The information queried by the system is automatically integrated from multiple sources. Query processing expands users' queries by exploiting a hierarchy of relevant concepts and provides users with the values of the sources elements associated with these concepts. The system can be managed semi-automatically, as its evolution is facilitated by the methods developed for its conception.

Keywords : bioinformatics, semantic Web, knowledge representation, bio-ontologies, integration system, syntactic and semantic heterogeneities, schema mappings.