Traitement des variabilités métier dans les Systèmes d'Information biologiques

Eric Leclercq, Marinette Savonnet, Pierre Naubourg

Traitement des variabilités métier dans les Systèmes d'Information biologiques

Éric Leclercq — Marinette Savonnet — Pierre Naubourg

Laboratoire LE2I - UMR CNRS 6306
Université de Bourgogne - 9, Avenue Alain Savary
21078 Dijon, France
Prenom.Nom@u-bourgogne.fr

RÉSUMÉ. *Les systèmes d'information scientifique nécessitent des fonctionnalités pour supporter deux types de variabilités majeures, la variabilité inter-acteurs et la variabilité inter-études. Nous traitons la variabilité inter-acteurs par un système d'importation des données garant de la qualité des données et la variabilité inter-études par l'utilisation d'un mécanisme d'annotation couplé au mécanisme de persistance. Afin de contrôler la qualité des données lors de leur importation en provenance de différents acteurs ou lors de leur annotation, nous proposons une approche basée sur deux niveaux de connaissance : 1) la connaissance relative aux applications du SI est représentée par une architecture de modèles, garante de la complétude et de la consistance des données du SI ; 2) la connaissance du domaine, représentée par une ontologie d'application, permet de fournir un cadre sémantique reconnu pour garantir la cohérence des données. Le système eClims qui implémente ces deux mécanismes est décrit.*

ABSTRACT. *Scientific information systems require consideration of two types of variability: variability between actors and variability between studies. We treated the variability between actors in a data import system that guarantees data quality and the variability between studies by the use of an annotation mechanism. To control high quality data when importing information from different actors, we propose an approach based on two levels of knowledge: 1) knowledge in the IS is represented by models and their implementation. They guarantee the completeness and consistency of data in the IS; 2) domain knowledge is represented by an application ontology. eClims system which implements these two mechanisms is described.*

MOTS-CLÉS : *connaissance évolutive, importation de données, annotation, SI biologique*

KEYWORDS: *evolutive knowledge, data import, annotation, biological IS*

1. Introduction

Le contexte de notre étude concerne des Systèmes d'Information (SI) qui fournissent un support aux activités scientifiques comme les SI biologiques. Pour S. Turki (Turki, 2005), ces SI ont une influence directe sur la production, la sélection, la gestion, l'utilisation et la diffusion de l'information et prennent de plus en plus d'importance face à la complexité des tâches et des techniques. Nous présentons dans la suite des éléments qui caractérisent les SI scientifiques.

En général, la portée et la complexité de l'activité scientifique sont telles qu'il est nécessaire de l'aborder dans un contexte d'équipes de recherche multi-disciplinaires réparties géographiquement sur plusieurs sites. L'infrastructure du SI doit supporter les notions de composition d'équipe, de rôles et de politiques de gestion des acteurs et proposer des outils collaboratifs. La complexité des questions abordées par les chercheurs s'accompagne d'une multiplicité des points de vue. Par exemple, un médecin spécialisé en physiologie aura besoin d'un point de vue global sur les paramètres physiologiques d'un sujet (ECG, respiration, pression artérielle, etc.). Un expert en parasitologie et mycologie médicale aura besoin d'exploiter des données protéiques pour l'étude d'une protéine agissant comme marqueur d'une maladie. Une autre caractéristique à prendre en compte est la multiplicité des sources de données. Galperin et Cochrane (Galperin *et al.,* 2011) ont dénombré en 2011, 1330 bases accessibles contenant plus de deux petabytes de données couvrant différents aspects de la biologie cellulaire et moléculaire. De même, plus de 60 ontologies ont été répertoriées dans le seul consortium OBO[1] (Smith *et al.,* 2007). Dans ce contexte, la capacité d'importation de données est une des fonctionnalités fondamentales des SI scientifiques. Pour traiter cette problématique, nous proposons une architecture capable de prendre en compte la variabilité des acteurs (section 3.2) en permettant d'importer des données provenant de sources multiples tout en garantissant leur qualité.

Dans le cas d'un SI d'entreprise, les fonctionnalités développées supportent les processus métier, par conséquent des procédures d'erreur préétablies et exhaustives sont développées pour répondre aux éventuelles exceptions. Les systèmes d'information scientifiques s'articulent autour d'études. Une étude est la recherche d'une réponse à une question donnée. Il peut être décidé à tout moment de mettre un terme à la poursuite de l'étude si les résultats partiels acquis permettent d'invalider une des hypothèses de l'étude, ainsi il n'existe pas de processus complétement modélisé pour représenter les études. En effet, la nature des recherches peut évoluer au fil des expérimentations et de nouvelles questions seront posées engendrant de nouvelles études. Un SI scientifique a donc comme finalité de produire de la connaissance ou d'améliorer la connaissance d'un sujet au travers d'activités de Recherche et Développement. La gestion de données scientifiques nécessite, au niveau du SI, un degré de souplesse qui est généralement beaucoup plus élevé que dans un SI d'entreprise. Par exemple, la découverte de nouvelles relations entre des composants biologiques (gènes, acides

1. Le consortium Open Biological and Biomedical Ontologies a pour but de fédérer les initiatives réalisées dans le développement d'ontologies biomédicales. `http://obofoundry.org`

aminés, protéines, etc.) sera rapidement prise en compte par les scientifiques et devra conduire à une évolution des SI gérant ce type de données. L'évolution de la connaissance se traduit entre autres par la modification de la structure des données. On estime par exemple, que la base de données Genbank[2] voit son schéma modifié deux fois par an (Navathe *et al.,* 2007). Un SI scientifique est donc un système extensible permettant l'ajout de connaissances et de données non initialement prévues. Pour traiter cette variabilité, nous proposons d'intégrer dans l'architecture du SI une base de données annotées (section 3.3).

Les SI scientifiques ont besoin de données de qualité pour répondre de manière fiable aux études alors que les variabilités entre les acteurs et inter-études tendent à réduire cette qualité. Les SI scientifiques doivent conserver les grands principes des bases de données mais ne peuvent en adopter les mêmes solutions. En effet, l'évolution de schémas dans les bases de données est un processus long et complexe qui ne permet pas d'atteindre le niveau de flexibilité requis pour l'importation de données multi-sources et la production de nouvelles connaissances. Notre approche se place dans le contexte biomédical et a pour objectif d'apporter une solution pour l'importation de données dans un LIMS (Laboratory Information Management System) et d'améliorer la flexibilité du stockage des données au moyen d'un mécanisme d'annotation tout en garantissant la qualité des données. En effet tout comme l'importation, l'annotation des données doit être contrôlée pour atteindre le niveau de qualité requis par le SI. Le plan de cet article est le suivant, la section 2 présente les caractéristiques des données bio-médicales. La section 3 présente les composants du système eClims pour l'importation et l'annotation. La section 4 présente les résultats expérimentaux du système eClims.

2. Caractéristiques des données biomédicales

Certaines caractéristiques sont partagées par toutes les données biologiques et biomédicales. Les travaux de Jagadish et Olken (Jagadish *et al.,* 2004) ont montré que ces données induisent des problèmes de représentation et de gestion aux concepteurs de SI. Chen et Carlis (Chen *et al.,* 2003) ont cherché à identifier les sources de ces problèmes. Ils ont mis en exergue les deux points suivants : 1) la connaissance nécessaire à la compréhension des données biologiques est importante, et 2) les personnes travaillant en bio-informatique ont divers profils, le plus souvent ce sont des personnes ayant des compétences différentes collaborant afin de répondre à un objectif commun. Les données à représenter et à gérer sont donc complexes car elles sont disparates, hétérogènes et elle souffrent d'un problème de qualité. La disparité des données provient des multiples processus mis en œuvre lors d'expérimentations biologiques, ainsi des données de différentes natures peuvent concerner le même élément physique. En effet, selon les traitements à réaliser les modèles sous-jacents seront différents comme

2. GenBank (qui s'appelle dorénavant Nucleotide) est une base de données de gènes, disponible publiquement, proposant les séquences nucléotidiques et leur traduction en protéines. `http://www.ncbi.nlm.nih.gov/sites/entrez?db=nucleotide`

par exemple des modèles de graphes pour représenter les voies métaboliques ou des modèles physico-mathématiques pour représenter le repliement des protéines.

La qualité des données biologiques dépend de l'incertitude, l'incomplétude, l'incohérence et l'inconsistance de certaines données (Willson, 1998). Ces caractéristiques sont inhérentes à la démarche scientifique. Par exemple à l'issue d'une analyse statistique, un pic de concentration protéique inattendu peut soulever des interrogations sur la pertinence de l'analyse et amener à considérer les données comme incertaines. Cette incertitude peut conduire soit à remettre en cause les expériences, soit à demander des renseignements complémentaires sur les échantillons soit aboutir à une nouvelle étude. De nombreux acteurs, internes ou externes à la plateforme, interviennent lors des expériences. Chacun de ces intermédiaires doit ajouter des informations à propos des manipulations qu'il a effectuées. Néanmoins il est très rare d'obtenir une liste exhaustive des manipulations effectuées par chacun des acteurs. En effet, certaines actions réalisées par un acteur ne portent pas à conséquence dans son contexte, il est donc, selon lui, inutile de les mentionner. De plus, lors de l'importation des données dans un SI biologique, il peut se produire des incohérences ou des inconsistances car les données à importer n'ont pas été acquises avec le même contexte que les données déjà présentes dans le SI.

Pour répondre aux exigences de qualité, les bio-informaticiens ont proposé la notion de base de données nettoyées (*curated databases*) et de base de données annotées (*annotated databases*). Le terme de base de données nettoyées est associé à des bases de données qui sont mises à jour avec un effort humain considérable (Buneman, 2009). Par exemple, il y a environ 150 experts (les curateurs) qui travaillent à temps-plein sur la base de données UniProt. Les curateurs sont des spécialistes du domaine ciblé par la base, leur rôle est de récupérer, corriger, valider et annoter les données. La plupart de ces bases de données jouent le rôle de publications scientifiques et remplacent les dictionnaires, les encyclopédies[3]. Les bases de données nettoyées ont généralement un schéma très simple qui représente les données "core", cependant la structure évolue au fil du temps en fonction des découvertes réalisées. Dans les bases de données annotées (Chiticariu *et al.,* 2005, Eltabakh *et al.,* 2008), les annotations sont utilisées pour permettre principalement de conserver la provenance de la donnée que ce soit la source ou le programme qui l'a généré mais aussi pour permettre une meilleure compréhension de la donnée. Par exemple, en indiquant comment la donnée a été obtenue, pourquoi certaines valeurs ont été ajoutées ou modifiées, quelles expériences ou analyses ont été exécutées pour obtenir les valeurs. Les systèmes d'annotations proposent différents niveaux de granularité des annotations. Par exemple, le système DBNotes[4] (DataBase anNOTation managEment System) permet d'annoter les valeurs des attributs d'une base de données relationnelles (Bhagwat *et al.,* 2005). Il utilise la forme la plus naïve de stockage puisque à chaque attribut de chaque relation est associé un autre attribut qui contiendra les annotations. Le système bdbms (biological database

3. Uniprot est devenue la référence scientifique du séquençage des protéines. `http://www.uniprot.org`
4. `http://users.soe.ucsc.edu/~wctan/Projects/dbnotes/index.html`

management system) (Eltabakh *et al.,* 2009), propose différents types d'annotations prédéfinis (commentaire, provenance). À chaque relation, plusieurs relations d'annotation peuvent être associées. Le système Belief Database (Gatterbauer *et al.,* 2009) présente un modèle de données relationnelles qui permet aux utilisateurs d'annoter à la fois le contenu (au niveau du tuple) et les autres annotations avec des croyances (*beliefs*). Les inconsistances entre les différents utilisateurs du système sont gérées par une logique modale qui représente les croyances sous la forme de structure de Kripke (Kripke, 1963).

La majorité des systèmes de base de données annotées propose une extension du langage SQL pour manipuler et interroger les annotations. Très peu de systèmes utilisent les techniques développées dans le cadre du Web Sémantique afin de modéliser et stocker les annotations. De plus, aucun système ne traite de la transposition des contraintes de bases de données sur les annotations.

3. Le système eClims

Afin d'assurer la qualité des données dans le SI biomédical, deux mécanismes doivent être particulièrement contrôlés : l'importation qui permet d'introduire de nouvelles données dans le SI et l'annotation qui permet de compléter des données existantes avec des données non initialement prévues. Le système eClims (experiments Clinical Information Management System) que nous proposons a été développé comme un module spécifique du LIMS ePims (experiments Proteomics Information Management System) afin de traiter les problèmes liés aux données biomédicales (Dupierris *et al.,* 2009).

3.1. *Exigences*

Lors de l'importation dans un environnement caractérisé par une forte variabilité inter-acteurs, les composants mis en place doivent permettre de : 1) connaître et exploiter la sémantique des données des SI source et cible. Pour cela, nous proposons d'utiliser des ontologies qui nous permettront d'aligner les informations dans un référentiel où elles peuvent être comparées ; 2) définir les règles de transformation des données source pour les importer dans le système cible ; 3) contrôler avant l'introduction dans la base de données, que les données source respectent les règles imposées par le système cible.

Lors de l'importation, les données pour lesquelles la structure d'accueil (c'est-à-dire les tables du SGBD) est déjà existante, sont stockées dans la base de données, les autres données sont transformées en annotations. Le mécanisme d'annotation peut être déclenché par l'importation ou par les utilisateurs du système souhaitant compléter des données. Dans ces deux cas, les composants mis en place doivent permettre de créer des annotations en utilisant la connaissance du domaine et de contrôler la consistance

et la cohérence des annotations ajoutées (entre elles et par rapport aux annotations existantes).

La figure 1 présente les différents éléments fonctionnels pour gérer l'importation des données et leur annotation. Les données communes aux différentes études sont stockées dans une base de données, les données complémentaires sont stockées sous la forme de triplets RDF. Le processus d'importation effectue un contrôle de la qualité en vérifiant la complétude, la consistance et la cohérence. Le problème de complétude peut exister à deux niveaux : 1) un concept peut avoir des attributs obligatoires non présents lors de l'importation, 2) un concept peut être associé à un autre concept de façon obligatoire et l'association n'existe pas lors de l'importation. L'inconsistance des données survient lorsque des caractéristiques différentes existent pour un même concept (par exemple, deux échantillons prélevés sur un même patient dont l'un est associé à une donnée de sexe féminin et l'autre à une donnée de sexe masculin). La cohérence des donnée est relative à la prise en compte de la connaissance du domaine (par exemple, est-il cohérent dans une étude sur le cancer du sein d'étudier un échantillon provenant d'un foie ?). Le processus d'annotation travaille sur des données déjà existantes mais nécessite seulement une vérification de la cohérence et de la consistance. Les trois mécanismes de contrôle de la qualité que nous proposons, exploitent un modèle UML représentant les données communes du système et une ontologie représentant la connaissance du domaine.

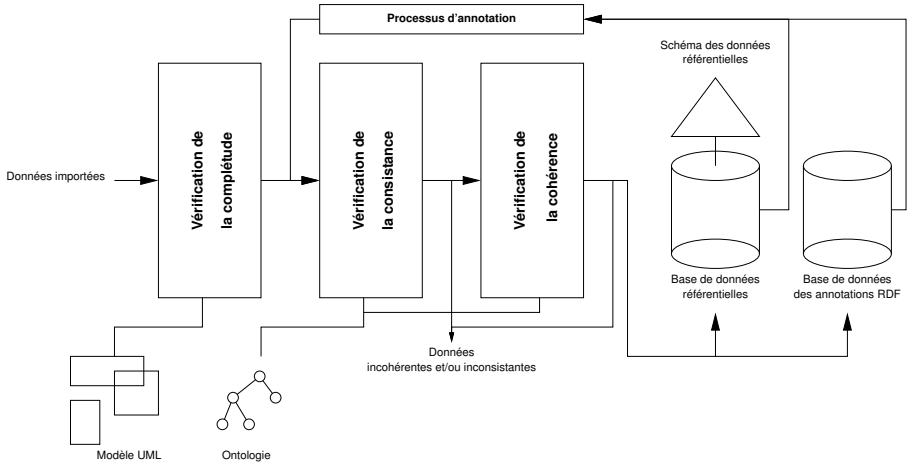

Figure 1. *Architecture générale*

Dans la suite de cette section, nous présentons l'architecture de modèles sur lequel repose le processus d'importation et les mécanismes de contrôle de la consistance et de la cohérence. Nous présentons ensuite le processus d'annotation et son modèle.

3.2. *Traitement de la variabilité des acteurs lors de l'importation de données*

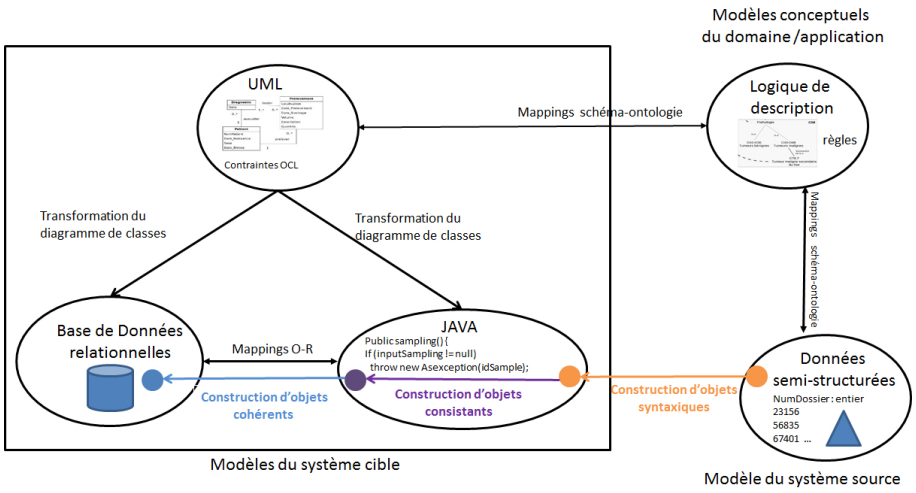

Figure 2. *Espaces technologiques utilisés*

Nous proposons une architecture basée sur une modélisation en couches (données, schémas et connaissances) contenant des modèles conceptuels et des modèles exécutables. Un modèle exécutable est défini comme la plus haute couche d'abstraction basée sur les langages d'implémentation (Jiang *et al.,* 2008).

3.2.1. *Architecture de modèles*

Le modèle de données du système cible, c'est-à-dire le module eClims que nous avons développé, est décrit grâce au langage de modélisation UML et principalement à l'aide du diagramme de classes. Ce diagramme de classes est transformé de manière semi-automatique en schéma de base de données relationnelles et en une collection de classes Java qui effectuent le mapping objet-relationnel. La communication (bi-directionnelle) entre la base de données et les objets Java peut être réalisée grâce au framework Hibernate[5] (partie gauche de la figure 2). Afin d'identifier les données à stocker dans la base de données et celles qui seront des annotations, nous nous sommes basés sur le principe des données référentielles[6] (Dreibelbis *et al.,* 2008). Les données référentielles sont identifiables au sein d'une application, d'un système ou d'un ensemble de systèmes, par leur importance primordiale au bon fonctionnement des processus mis en œuvre. L'identification de ces données porte sur trois notions principales :

5. http ://www.hibernate.org
6. Les données référentielles sont aussi connues sous le nom anglais de *Master Data Management*

– le partage concerne des données utilisées par différents systèmes ou par différents blocs fonctionnels au sein d'un même système ;

– la stabilité concerne des données étant rarement amenées à évoluer ;

– la fréquence de consultation concerne des données, servant de pivot à de nombreux processus et consultées fréquemment.

Une ontologie d'application, vue comme référence à la connaissance du domaine, est utilisée comme médiatrice entre les schémas sources (c'est-à-dire les schémas des acteurs) et le schéma du système cible. Elle doit mettre en relation les jeux de données des sources (essentiellement des fichiers CSV ou XLS) avec le modèle de données cible, traiter les problèmes de formats, de domaines et d'échelles (partie droite de la figure2). L'ontologie d'application est structurée en trois branches distinctes : 1) la première branche décrit les concepts qui seront utilisés afin de mettre en correspondance les descripteurs des schémas des sources avec le modèle du système cible selon leur signification. Il s'agit d'une représentation ontologique des données référentielles du système cible ; 2) la deuxième branche est utilisée afin de définir quels types et formats de données sont concernés pour chaque descripteur de données des schémas ; 3) la troisième branche définit les opérations de conversion entre les différents types et formats de données.

La mise en correspondance est basée sur le concept de mapping. Un mapping schéma-ontologie MSO est un couple $< \{D\}; C_o >$ constitué d'un ensemble de descripteurs D et d'un concept C_o de l'ontologie o.

Par exemple, un descripteur de type chaîne appartenant au système cible dont le label est *NumPatient* sera mis en correspondance avec le concept de l'ontologie *IdPatient* grâce au mapping MSO1 :

MSO1 : $<< NumPatient; \textbf{\textit{Chaîne}} >; IdPatient >$

Un descripteur de type entier appartenant à un jeu de données de l'acteur A1 dont le label est *NumDossier* sera mis en correspondance avec le concept de l'ontologie d'application *IdPatient* grâce au mapping MSO2. En effet, l'acteur A1 étant un CHU, le patient est connu par son numéro de dossier :

MSO2 : $<< NumDossier; Entier >; IdPatient >$

Le mapping MSO3 est un exemple de mapping n..1 qui associe trois descripteurs au concept TNM[7] de l'ontologie :

MSO3 : $< \{< T; Entier >, < N; \textbf{\textit{Chaîne}} >, < M; Entier >\}; TNM >$

3.2.2. *Processus d'importation*

Pour mettre en place le contrôle de la qualité des données lors de l'importation dans un environnement caractérisé par une variabilité inter-acteurs, nous proposons

7. TNM est une méthode de description, de l'étendue d'un cancer.

un processus reposant sur trois niveaux (figure 3). Ce processus est présenté en détail dans (Naubourg *et al.,* 2011).

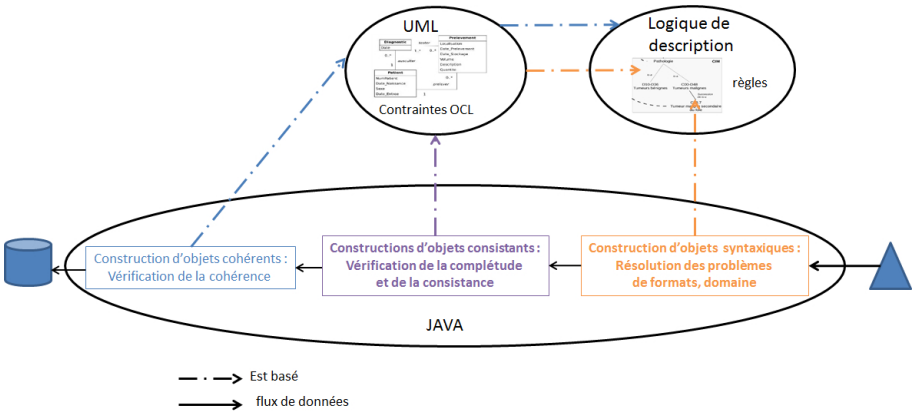

Figure 3. *Processus d'importation*

Le premier niveau consiste en la création d'objets Java correspondant aux données. Pour construire de tels objets syntaxiques : 1) nous associons des mappings schéma-ontologie afin de découvrir les couples de mappings acteur-ontologie et système cible-ontologie qui correspondent, 2) nous convertissons éventuellement des valeurs des données de l'acteur en des valeurs admises par le système cible et 3) nous affectons ces valeurs au sein d'objets Java. Ces objets Java peuvent être manipulés afin de vérifier leur complétude, leur consistance et leur cohérence.

Le diagramme de classes UML du système cible permet de spécifier des contraintes portant sur la complétude et la consistance des données comme les attributs et les associations obligatoires ou l'unicité de valeur. Nous avons utilisé les typologies proposées par Costal (Costal *et al.,* 2008) et Miliauskaite (Miliauskaite *et al.,* 2005) sur les types de contrainte exprimables au sein des diagrammes UML pour réaliser ce niveau. Nous travaillons à partir d'objets complets afin d'assurer leur consistance par rapport au diagramme de classes UML et nous vérifions les contraintes suivantes :

– les valeurs admises ;

– l'unicité interne est basée sur la définition d'attributs identifiants. Dans le cas où deux objets possèdent les mêmes valeurs pour les attributs identifiants, ils doivent aussi posséder les mêmes valeurs pour les autres attributs. Par exemple, si deux patients ayant le même identifiant HG65 possèdent deux valeurs différentes pour le sexe, les deux objets ne sont pas consistants ;

– les association récursives : une contrainte de ce type est traitée en fonction de son type (réflexive, acyclique, symétrique, asymétrique, anti-symétrique) ;

– la comparaison de chemins : une contrainte de ce type détermine deux chemins. Chaque chemin contient un point de départ et la succession des associations don-

nant l'ensemble final du chemin. La vérification des ensembles des deux chemins commence par les types des objets des ensembles qui doivent être identiques. Des vérifications différentes sont ensuite mises en place en fonction des propriétés des ensembles (inclusifs, distincts ou égaux). Par exemple une contrainte détermine deux chemins depuis la classe *Patient* jusqu'à la classe *Pathologie*. Le premier chemin définit l'ensemble des pathologies d'un patient via la classe *Diagnostic*. Le deuxième définit l'ensemble des pathologies d'un patient via la classe *Hospitalisation*. Les deux ensembles d'objets doivent être égaux.

Le dernier niveau porte sur la vérification de la cohérence et s'appuie sur une modélisation de la connaissance du domaine. Le processus de vérification de la cohérence repose sur un moteur de règles prenant en compte les faits, c'est-à-dire les objets nouvellement créés, la connaissance représentée sous la forme de l'ontologie et les règles métiers. Afin de ne travailler qu'avec des règles décidables, nous devons respecter les recommandations DL-Safe. Ces recommandations portent principalement sur la définition de règles travaillant sur des individus connus appartenant à des concepts nommés (Motik *et al.,* 2010). Pour respecter ces deux points, nous devons créer les individus représentant des objets Java, ces objets sont transformés en plusieurs individus représentant ses attributs. Puis nous devons affecter chaque individu à un concept de l'ontologie. Une fois les individus correspondant aux objets Java créés, le moteur de règles vérifie les règles portant sur l'ontologie. Par exemple, le moteur de règles utilise ces individus, l'ontologie, et la règle énonçant « qu'un prélèvement est valide si la pathologie pour lequel il est étudié et l'organe dont il provient sont mutuellement pertinents » pour vérifier, pour chaque individu du concept *Prelevement* lié un individu du concept *Organe* et à un individu du concept *Pathologie*, qu'une relation de type *organeTouché* existe au sein de l'ontologie. Si le moteur de règles ne trouve aucune relation définissant que le foie est un organe touché par la pathologie néoplasme du sein, il déterminera que le prélèvement est invalide et les données s'y référant.

À la fin de ce processus, nous obtenons soit des objets qui ont passés les trois vérifications avec succès soit des objets non validés. Tous les objets sont insérés dans le système cible, les objets non validés sont annotés comme "incertain" empêchant toute utilisation par les autres modules du système. Ce choix est dicté par le fait que le matériel biologique humain est rare est qu'il est inconcevable de s'en séparer sans rechercher aurpès des différents acteurs des informations complémentaires pour lever l'incertitude.

3.3. *Traitement de la variabilité inter-études par les annotations*

3.3.1. *Modèle d'annotation*

Le modèle d'annotation repose sur la définition d'une annotation, sous la forme d'un triplet liant le sujet et un objet au moyen d'un prédicat ayant une sémantique faisant référence à des termes d'une ontologie.

Définition 1. Une annotation Ann est un triplet (Sj, Pr, Ob) où Sj est la donnée que l'on souhaite annoter nommée sujet, Pr est un prédicat qui identifie la sémantique de l'annotation et Ob est un objet qui désigne le contenu de l'information que l'on souhaite ajouter.

Définition 2. Un sujet d'annotation Sj est un couple $(Conteneur, IdSujet)$ où $Conteneur$ est le nom du conteneur de l'objet identifié par une valeur $IdSujet$.

Dans le cas de l'utilisation d'un SGBDR le conteneur est composé d'un nom de la table, préfixé par la chaîne de connexion permettant de se connecter à la base de données via un protocole réseau (ODBC, JDBC par exemple). L'identifiant $idSujet$ peut être le ROWID ou un identifiant technique ou encore un identifiant logique.

Définition 3. Un prédicat d'annotation Pr est constitué d'une composante (C_o) désignant un concept C_o de l'ontologie o.

D'un point de vue opérationnel, il s'agira par exemple d'une URI faisant référence à un terme d'un fichier OWL ou une référence identifiant un tuple dans une base de données.

Définition 4. Un objet d'annotation Ob est constitué soit :

– d'une seule composante $(Sujet)$ définissant un sujet d'annotation existant ou *null* ;

– de deux composantes $(Valeur, Type_o)$ où $Valeur$ désigne la valeur de l'objet et $Type_o$ désigne le type de la $Valeur$ issu de l'ontologie o. $Valeur$ appartient à l'ensemble des individus de l'ontologie o ou $Valeur$ est un littéral.

3.3.2. *Expressivité du modèle*

Grâce au triplet (Sj, Pr, Ob) le modèle d'annotation permet d'associer les sujets à des objets selon des prédicats. Six modes d'annotations sont possible en fonction de la nature des sujets et des objets :

1) le sujet est une donnée de référence et l'objet est une valeur. Par exemple, une annotation associe le patient P1 à une maladie et indique qu'il est fumeur. Ces annotations peuvent être développées sous la forme suivante : $A_1 = (Patient\#P1, maladie, endocardite)$ et $A_2 = (Patient\#P1, fumeur, null)$;

2) le sujet est une annotation et l'objet est une valeur. Ce mode sert à compléter une annotation déjà réalisée. Par exemple les annotations suivantes expriment que la quantité de cigarettes fumées par la patient P1 est de 4 et est exprimée en cigarettes par jour $A_3 = (A_2, \textit{quantité}, 4)$, $A_4 = (A_3, \textit{unité}, cigarette/j)$;

3) le sujet et l'objet sont des données de référence. Par exemple lors de la réalisation d'une étude sur l'hérédité d'une maladie, il est possible de créer une annotation entre deux patients via un prédicat déterminant leur relation de filiation $A_5 = (Patient\#P1, \textit{père}, Patient\#P5)$;

4) le sujet est une donnée de référence et l'objet est une annotation. Ce mode est utile pour spécifier que plusieurs données partagent une même annotation. Par

exemple plusieurs patients suivent le même traitement ;

5) le sujet est une annotation et l'objet est une donnée de référence. Ce mode est utilisé pour exprimer une relation complexe entre deux données au moyen de plusieurs annotations ;

6) le sujet et l'objet sont des annotations. Ce mode permet de relier deux annotations pour signifier l'existence d'une relation entre les deux annotations ou bien pour partager des annotations complexes. Les annotations suivantes expriment le fait que le patient P1 suit deux traitements simultanément : $A_6 = (Patient\#P1, traitement, SprayX54)$, $A_7 = (Patient\#P1, traitement, SprayY15)$ et $A_8 = (A_6, \textit{simultané}, A_7)$.

Dans notre modèle, le prédicat (terme de l'ontologie) ne peut pas faire l'objet d'annotation. En effet, la connaissance sur les éléments de l'ontologie est exprimée dans l'ontologie exclusivement.

4. Expérimentation

La plateforme protéomique CLIPP[8] a servi de base pour expérimenter le système eClims. Le suivi des échantillons de protéomique clinique nécessite la mise en place d'une gestion rigoureuse. Cette gestion passe par l'utilisation d'un LIMS permettant de prendre en compte les données en amont, pendant et en aval des expériences. Notre travail dans ce contexte consiste à gérer les données au moment de leur entrée dans eClims. De nombreux acteurs fournissent des données de qualité inégale, cependant une fois importées avec les données présentes dans eClims, les données des acteurs doivent avoir la même qualité que les données existantes. En effet, les données concernent des études cliniques sur les humains et peuvent conduire à des conclusions erronées si la qualité n'est pas acceptable. Le type de données à importer peut varier d'une étude à une autre, il a été impossible de prévoir lors de la construction de eClims l'ensemble des besoins et des évolutions.

Au sein de notre système, les données cliniques importées concernent les patients, les prélèvements et les échantillons. La connaissance relative au domaine de la protéomique clinique est représentée par la branche signification de l'ontologie (figure 4). Cette ontologie représente l'utilisation que fait eClims du domaine et permet ainsi de mettre en correspondance les éléments du modèle de eClims avec les éléments des acteurs. Dans eClims, l'ontologie a été réalisée en OWL et englobe des ressources consensuelles telles que les recommandations aux tumorothèques, la CIM, MeSH, la classification TNM. Les recommandations sur les tumorothèques hospitalières, réalisées en 2006 par l'INCa[9], présentent l'ensemble des données cliniques considérées comme pertinentes (renseignements sur le patient, sur la maladie) et un ensemble de techniques et protocoles à respecter afin de préserver la qualité des échantillons biologiques. Pour identifier les maladies, nous avons retenu la Classification Internationale

8. CLIPP : CLinical and Innovation Proteomic Platform `http://www.clipproteomic.fr/`
9. Institut National du Cancer (INCa) `http://www.e-cancer.fr`

des Maladies[10] (CIM) proposée par l'Organisation Mondiale de la Santé (OMS). La CIM permet le codage des maladies, des traumatismes et d'une manière générale de l'ensemble des motifs de recours aux services de santé. Le thésaurus MeSH[11] (Medical Subject Headings) est un outil réalisé par la National Library of Medicine. Il est utilisé pour l'indexation et la recherche d'informations médicales. Les descripteurs MeSH sont organisés en 16 catégories : la catégorie A pour les termes anatomiques, la catégorie B pour les organismes, la catégorie C pour les maladies, etc. Chaque catégorie est subdivisée en sous-catégories comportant des descripteurs structurés hiérarchiquement, des plus généraux aux plus spécifiques. La nomenclature TNM (Tumor, Nodes, Metastasis) est un système international permettant de définir les stades de développement des tumeurs.

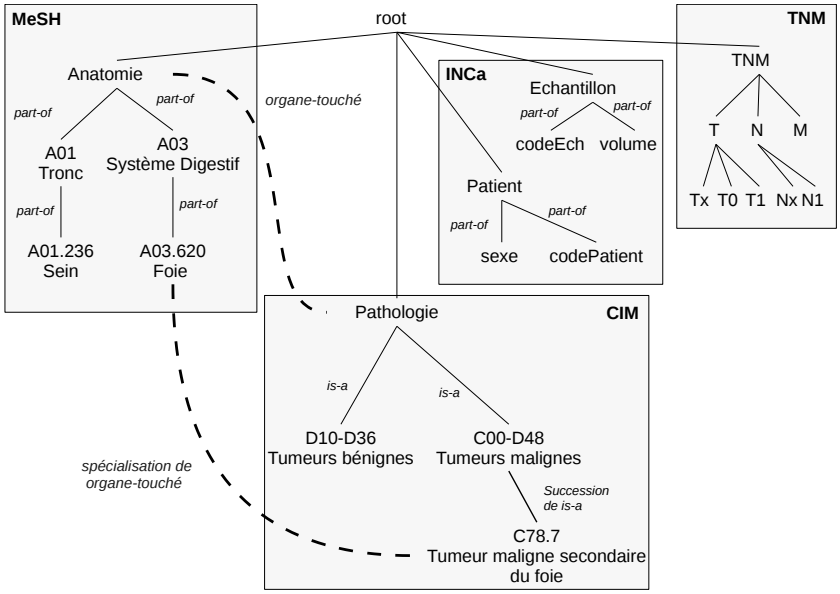

Figure 4. *Branche de l'ontologie décrivant les données référentielles de eClims*

La figure 5 est une capture d'écran présentant les cinq étapes nécessaires à l'importation 1) le choix des options d'importation, 2) la sélection du fichier à importer, 3) la sélection du programme de recherche (c'est-à-dire d'une étude, dans la figure 5 nommée "infection fongique"), 4) la visualisation du fichier et 5) la réalisation des mappings schéma-ontologie. La visualisation des données du fichier autorise la modification des données avant l'importation. Enfin la réalisation des mappings schéma-ontologie permet de lier les descripteurs du fichier d'importation au schéma d'eClims.

10. En anglais : International Classification of Diseases (ICD) `http://www.who.int/classifications/icd`
11. `http://www.ncbi.nlm.nih.gov/mesh`

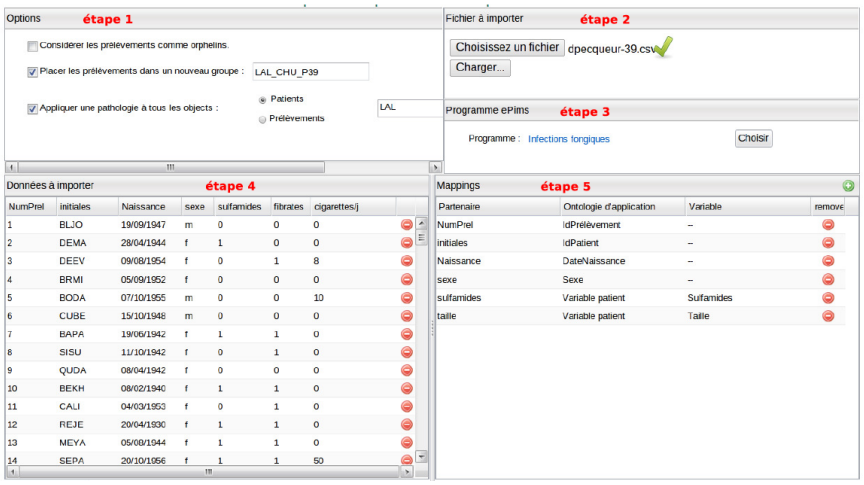

Figure 5. *Interface d'importation des fichiers cliniques*

Une fois la définition de l'importation réalisée, les données sont créées et présentées à l'utilisateur qui peut les modifier avant leur stockage en base de données.

L'annotation des données de eClims peut être effectuée, lors de la consultation et lors de l'importation des données. L'utilisateur peut choisir le prédicat et le type parmi une liste. Si le prédicat souhaité n'existe pas, il a la possibilité d'en créer un nouveau. La figure 6 présente une capture d'écran où les données complémentaires de descripteurs sulfamides et taille sont associées à des annotations de prédicats Sulfamides et Taille. Les annotations mises en place dans eClims ne s'appliquant qu'aux données

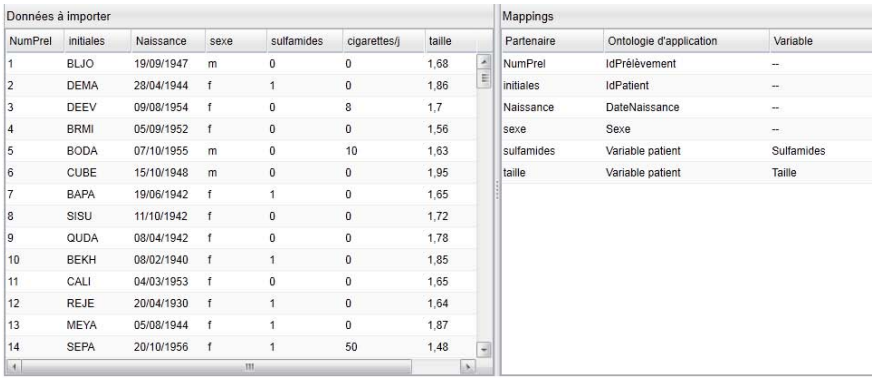

Figure 6. *Données complémentaires sous forme d'annotations*

désignant des patients, des prélèvements et des échantillons. L'implémentation a permis de valider les fonctionnalités suivantes :

– l'ajout de données complémentaires durant : 1) le processus d'importation, quand elles sont présentes dans les jeux de données des acteurs ou 2) "à la volée" lorsqu'elles sont fournies a posteriori par les partenaires dans le cadre par exemple d'une demande complémentaire d'information par la plateforme CLIPP ;

– l'interrogation des annotations et des données en utilisant les Criteria Queries du framework Hibernate.

5. Conclusion

Dans cet article, nous avons proposé un mécanisme d'importation de données et un mécanisme d'annotation afin de traiter les variabilités inter-partenaires et inter-études dans les SI scientifiques. Ces deux mécanismes s'appuient sur un contrôle de la qualité utilisant une ontologie applicative et des règles métier.

Notre proposition se distingue d'un ETL classique. En effet, au sein d'un ETL, l'opération d'extraction (E) nécessite de connaître les modes de communication avec le système source. Dans notre cas, les systèmes sources sont des jeux de données représentant une extraction réalisée par les acteurs. Notre processus de transformation (T) propose deux niveaux de complexité : 1) une transformation syntaxique consistant à convertir les valeurs données d'un type en un autre. Ce processus est automatique au sein de notre approche. L'utilisation d'un ETL obligerait les utilisateurs à réaliser un processus de transformation pour chaque type ; 2) une transformation sémantique consistant à modifier les valeurs selon leurs significations dans les différents systèmes.

L'application et son système d'annotation sont en production depuis juin 2011 au sein de la plateforme proéomique CLIPP, elle est interfacée avec le LIMS ePims.

Le système de gestion de contraintes pour la vérification de la cohérence est encore au stade de prototype, nos travaux actuels concernent l'implantation de contraintes de base de données sur les annotations comme explorée dans (Akhtar *et al.,* 2010), (Bidoit-Tollu, 2010).

6. Bibliographie

Akhtar W., Cortés-Calabuig A., Paredaens J., « Constraints in RDF », *SDKB*, p. 23-39, 2010.

Bhagwat D., Chiticariu L., Tan W. C., Vijayvargiya G., « An annotation management system for relational databases », *VLDB J.*, vol. 14, n° 4, p. 373-396, 2005.

Bidoit-Tollu N., « Types and Constraints : From Relational to XML Data », *SDKB*, p. 40-53, 2010.

Buneman P., « Curated Databases », *European Conference on Digital Libraries (ECDL)*, p. 2, 2009.

Chen J. Y., Carlis J. V., « Genomic data modeling », *Inf. Syst.*, vol. 28, n° 4, p. 287-310, 2003.

Chiticariu L., Tan W. C., Vijayvargiya G., « DBNotes : a Post-It System for Relational Databases based on Provenance », *Proceedings of the ACM SIGMOD International Conference on Management of Data*, p. 942-944, 2005.

Costal D., Gómez C., Queralt A., Raventós R., Teniente E., « Improving the definition of general constraints in UML », *Software and System Modeling*, vol. 7, n° 4, p. 469-486, 2008.

Dreibelbis A., Hechler E., Milman I., Oberhofer M., van Run P., Wolfson D., *Enterprise Master Data Management : An SOA Approach to Managing Core Information*, 1 edn, IBM Press, 2008.

Dupierris V., Barthe D., C. B., « ePIMS : un LIMS pour la gestion des données de spectrométrie de masse », *Spectra Analyse*, vol. 38, n° 269, p. 36-40, 2009.

Eltabakh M. Y., Aref W. G., Elmagarmid A. K., Ouzzani M., Silva Y. N., « Supporting Annotations on Relations », *12th International Conference on Extending Database Technology (EDBT)*, p. 379-390, 2009.

Eltabakh M. Y., Ouzzani M., Aref W. G., Elmagarmid A. K., Laura-Silva Y., Arshad M. U., Salt D., Baxter I., « Managing Biological Data using bdbms », *Proceedings of the 24th International Conference on Data Engineering (ICDE)*, p. 1600-1603, 2008.

Galperin M. Y., Cochrane G., « The 2011 *Nucleic Acids Research* Database Issue and the online Molecular Biology Database Collection », *Nucleic Acids Research*, vol. 39, n° Database-Issue, p. 1-6, 2011.

Gatterbauer W., Balazinska M., Khoussainova N., Suciu D., « Believe It or Not : Adding Belief Annotations to Databases », *Computing Research Repository (CoRR)*, 2009.

Jagadish H. V., Olken F., « Database Management for Life Sciences Research », *SIGMOD Record*, vol. 33, n° 2, p. 15-20, 2004.

Jiang K., Zhang L., Miyake S., « Using OCL in Executable UML », *ECEASST*, 2008.

Kripke S., « Semantical Considerations on Modal Logic », *Acta Philosophica Fennica*, vol. 16, p. 83-94, 1963.

Miliauskaite E., Nemuraite L., « Representation of integrity constraints in conceptual models », *Information Technology And Control*, vol. 34, p. 355-365, 2005.

Motik B., Rosati R., « Reconciling Description Logics and Rules », *Journal of the ACM*, vol. 57, n° 5, p. 1-62, 2010.

Naubourg P., Savonnet M., Leclercq E., Yétongnon K., « Approche préventive de la qualité des données dans le contexte de la protéomique clinique », *Revue des Nouvelles technologies de l'Information*, vol. E.22, p. 189-234, 2011.

Navathe S. B., Patil U., Guan W., « Genomic and Proteomic Databases : Foundations, Current Status and Future Applications », *JCSE*, vol. 1, n° 1, p. 1-30, 2007.

Smith B., Ashburner M., Rosse C., Bard J., Bug W., Ceusters W., Goldberg L. J., Eilbeck K., Ireland A., Mungall C. J., Leontis N., Rocca-Serra P., Ruttenberg A., Sansone S.-A., Scheuermann R. H., Shah N., Whetzel P. L., Lewis S., « The OBO Foundry : coordinated evolution of ontologies to support biomedical data integration », *Nat Biotech*, vol. 25, n° 11, p. 1251-1255, 2007.

Turki S., Des hyperclasses aux composants pour l'ingénierie des systèmes d'information, PhD thesis, Université de Genève, Université Joseph-Fourier de Grenoble, 2005.

Willson S., « Measuring Inconsistency in Phylogenetic Trees », *Journal of Theoretical Biology*, vol. 190, n° 1, p. 15-36, 1998.