Comparaison de deux modes de représentation de données faiblement structurées en sciences du vivant

Rallou Thomopoulos*, Patrice Buche**, Ollivier Haemmerlé***, Frédéric Mabille* et Nongyao Mueangdee*

*INRA, UMR IATE, 2 place Viala, 34060 Montpellier cedex 1
{rallou, mabille, nongyao}@ensam.inra.fr
**INRA, UMR Mét@risk, 16 rue Claude Bernard, 75231 Paris cedex 5
Patrice.Buche@inapg.fr
***GRIMM-ISYCOM, Univ. Toulouse le Mirail, Dépt. Mathématiques-Informatique
5 allées Antonio Machado, 31058 Toulouse cedex
Ollivier.Haemmerle@univ-tlse2.fr

Résumé. Cet article présente deux modes de représentation de l'information dans le cadre d'une problématique en sciences du vivant. Le premier, appliqué à la microbiologie prévisionnelle, s'appuie sur deux formalismes, le modèle relationnel et les graphes conceptuels, interrogés uniformément via une même interface. Le second, appliqué aux technologies des céréales, utilise le seul modèle relationnel. Cet article décrit les caractéristiques des données et compare les solutions de représentation adoptées dans les deux systèmes.

1 Introduction

L'étude de la représentation de données faiblement structurées (ou semi-structurées) a connu une explosion récente avec l'émergence de l'internet et la popularité du standard XML. Abiteboul (1997) recense les principaux aspects pouvant caractériser ces données : une structure irrégulière, implicite ou partielle ; un schéma qui se veut indicatif plutôt qu'impératif, souvent construit *a posteriori*, de grande taille, évoluant rapidement ; des types de données éclectiques et une difficulté à établir la distinction entre schéma et données.

De telles données sont courantes dans les sciences du vivant, où l'on trouve également d'autres "verrous" liés à la complexité des phénomènes étudiés (Keet, 2003) : des données dont la précision est limitée par les techniques de mesure, des données variables, non répétables, voire contradictoires, des paramètres nombreux et imbriqués, des données ne pouvant couvrir tous les cas d'expérimentation possibles. C'est notamment en génomique que les bases de données biologiques ont été le plus tôt et le plus abondamment étudiées (Cherry et al., 1998). On trouve cependant des bases de données dans de nombreux autres domaines (environnement, botanique, etc.) (Keet, 2004), avec des modèles de représentation différents fondés notamment sur le modèle relationnel (Bukhman et Skolnick, 2001), le modèle objet (Raguenaud et al., 2002) ou les graphes (Zhong et al., 1999).

Dans cet article, nous proposons une comparaison entre deux bases de données en sciences du vivant, développées à l'INRA[1] : la base de données Sym'Previus, appliquée au risque microbiologique alimentaire, et la base de données Grain Virtuel, appliquée aux technologies des céréales. Dans les deux cas, les données sont faiblement structurées pour les raisons suivantes : (i) elles sont hétérogènes tant dans leur contenu que dans leur format ; (ii) les relations existant entre elles sont complexes ; (iii) leur évolution est difficilement prévisible. Les formalismes choisis pour représenter les données sont différents : la base de données Sym'Previus s'appuie sur deux formalismes distincts, le modèle relationnel et le modèle des graphes conceptuels, tandis que le projet Grain Virtuel utilise le seul modèle relationnel. L'objectif de cet article est de montrer comment est traitée la faible structuration des données dans les deux cas et de discuter des avantages et inconvénients des deux approches.

Les parties 2 et 3 présentent respectivement le projet Sym'Previus et le projet Grain Virtuel. Les deux approches sont comparées dans la partie 4.

2 Le projet Sym'Previus, une application en microbiologie prévisionnelle

2.1 Les données du projet

Le projet Sym'Previus, initié en 1999, associe des organismes de recherche appliquée, des industriels et des centres techniques de conseil en agroalimentaire. L'objectif est de mettre en place un outil d'analyse du risque microbiologique dans les aliments. Afin de pouvoir chiffrer ce risque, l'étape initiale consiste à regrouper un maximum de données scientifiques fiables et permettre leur stockage et leur interrogation sous la forme d'un système d'information.

Les données sont issues à la fois de la bibliographie scientifique en microbiologie et de données industrielles. Elles décrivent la teneur en microorganismes de produits alimentaires (concentration en germes pathogènes tels que *Listeria* dans le lait par exemple) et l'évolution de cette teneur sous l'effet de différents facteurs intervenant au cours du cycle de vie des aliments (chauffage, conservation, ...). Ces données sont difficiles à structurer pour deux raisons principales. D'une part, les sources d'information sont hétérogènes, elles regroupent des publications traitant de thématiques variées et des données industrielles de différents formats. D'autre part, l'information n'est pas stabilisée, de nouvelles données émergent avec l'avancée de la recherche. Il est donc difficile de prévoir un schéma qui ne soit rapidement "périmé".

Les données présentent d'autres caractéristiques (imprécision, incomplétude, organisation taxonomique) qui ne sont pas développées dans cet article. Le lecteur intéressé pourra se reporter à Thomopoulos (2003); Buche et al. (2005).

2.2 Solution initiale

Le projet Sym'Previus s'est initialement construit autour d'une base de données relationnelle. Une première version commerciale devant être rapidement disponible, ce choix a été fait en raison de la robustesse et de l'efficacité du modèle relationnel, largement étudié et utilisé et ayant "fait ses preuves". Le modèle relationnel a également fait l'objet de travaux concernant la représentation des données imprécises, une des caractéristiques des données du projet. Cette base de données est composée d'environ 90 tables et contient 10000 enregistrements de

[1] Institut National de la Recherche Agronomique

résultats scientifiques issus de plus de 700 publications en microbiologie. Elle est accessible via une interface de saisie et une interface d'interrogation des données.

Toutefois, une partie des publications ne peuvent pas être saisies dans la base de données, bien qu'elles soient estimées d'un intérêt scientifique justifiant leur prise en compte. Certaines contiennent des informations proches de celles de la base de données relationnelle, mais avec des données supplémentaires ou exprimées dans un format différent, et des données manquantes ; d'autres contiennent des informations incompatibles avec la structure de la base de données relationnelle (composition des aliments en acides organiques, interactions entre microorganismes, ...), l'intérêt pour ce type d'informations ayant émergé postérieurement à la création de la base relationnelle.

Une modification de la base de données relationnelle, afin de prendre en compte ces données, est une opération coûteuse et ne peut pas être envisagée fréquemment. Elle nécessite la modification du schéma de la base de données relationnelle, la migration des données existantes vers le nouveau schéma, la modification de l'interface de saisie des données, enfin la modification de l'interface d'interrogation. Il a donc paru préférable de limiter les modifications de la base de données relationnelle à une certaine périodicité (par exemple, tous les deux ans) et d'utiliser une base complémentaire pour stocker les informations non prévues par le schéma relationnel. Le formalisme de cette base complémentaire doit permettre de pallier les limites de la base de données relationnelle, c'est-à-dire essentiellement la rigidité de sa structure, en étant suffisamment souple pour traiter des informations initialement non prévues.

2.3 Le modèle des graphes conceptuels

Le formalisme choisi pour cette base complémentaire est le modèle des graphes conceptuels (Sowa, 1984). Ce modèle, issu de l'intelligence artificielle et fondé sur des graphes étiquetés, présente plusieurs avantages : (i) son aptitude à la représentation de données faiblement structurées. Soulignons que nombre de travaux concernant la représentation de données faiblement structurées s'appuient sur des graphes étiquetés, parmi lesquels le modèle OEM (Object Exchange Model, Abiteboul et al. (1997)) ou le standard XML (eXtensible Markup Language, Bray et al. (1998)) ; (ii) sa représentation graphique, relativement intuitive pour des utilisateurs non-spécialistes ; (iii) la gestion d'une terminologie du domaine, permettant de représenter l'organisation taxonomique des données ; (iv) l'existance d'opérations permettant le raisonnement sur les données ; (v) son interprétation logique, offrant un cadre théorique robuste ; (vi) des plates-formes de développement disponibles, fournissant des algorithmes efficaces.

Le modèle distingue le *support*, qui contient les connaissances terminologiques, et les *graphes conceptuels*, qui contiennent les connaissances assertionnelles. La figure 1 est un exemple de graphe conceptuel représentant l'information suivante : "l'expérience E1 a pour objet l'interaction I1 entre Listeria Scott A et la nisine dans du lait écrémé, qui a pour résultat une réduction". Il comprend notamment : (i) un ensemble de *sommets concepts* (notés par des rectangles). Chaque sommet concept est étiqueté par un type de concept (*Expérience*, *Interaction*, ...) et un marqueur individuel représentant une instance de ce type de concept (*E1*, *I1*) ou générique (noté $*$). L'ensemble des types de concepts constitue une hiérarchie partiellement ordonnée par la relation *sorte de* et déclarée dans le support ; (ii) un ensemble de *sommets relations* (notés par des ovales) qui expriment la nature des liens entre les concepts. Chaque sommet relation est étiqueté par un type de relation (*Obj*, *Agt*, ...).

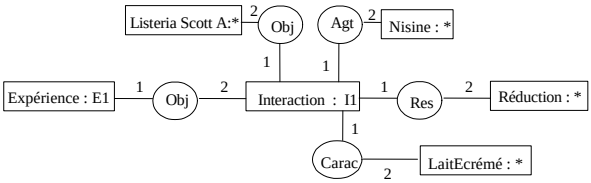

FIG. 1 – *Un exemple de graphe conceptuel*

L'ensemble des graphes conceptuels est partiellement préordonné par une *relation de spécialisation*, qui peut être calculée par l'opération de *projection* (un morphisme de graphes autorisant une restriction de l'étiquette des sommets). La projection est une opération fondamentale du modèle puisqu'elle permet de rechercher des réponses à une requête (les réponses pouvant être considérées comme des spécialisations de la requête).

2.4 Une solution fondée sur deux formalismes, structuré et faiblement structuré

La figure 2 schématise l'architecture de la solution adoptée dans le projet Sym'Previus : les deux modèles sont interrogés via le même système, de façon transparente pour l'utilisateur. Chaque requête est envoyée vers les deux sous-systèmes, où elle est traduite dans le forma-

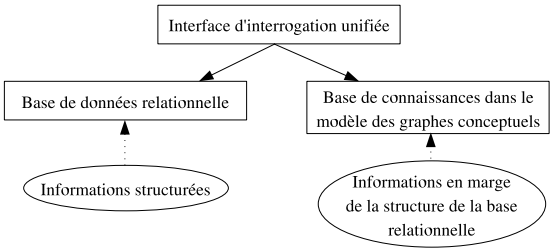

FIG. 2 – *Solution adoptée dans le projet Sym'Previus*

lisme correspondant : une requête SQL pour le modèle relationnel, un graphe-requête pour le modèle des graphes conceptuels. Du fait de la préexistance du sous-système relationnel, le sous-système graphes conceptuels s'est adapté au langage d'interrogation, inspiré du formalisme relationnel. Dans le système d'interrogation unifiée, une requête comprend un ensemble d'attributs de projection et un ensemble de critères de sélection de la forme <attribut, valeur> (voir Buche et Haemmerlé (2000) pour plus de détails). Elle est exprimée dans une vue donnée. La notion de vue, classique en bases de données, correspond à une table virtuelle qui regroupe de façon cohérente l'ensemble des attributs nécessaires à l'utilisateur. Par exemple, la requête *Q = {Vue=Interaction, Substrat, RésultatExpé., <Substrat, Lait>, <Pathogène, Listeria>}* spécifie que l'utilisateur souhaite connaître le substrat et le résultat expérimental (attributs de projection), lorque le substrat est le lait et que le germe pathogène est Listeria (critères de sélection), dans la vue Interaction.

Cette syntaxe implique certaines contraintes au sein du sous-système graphes conceptuels afin d'assurer la cohérence de l'interrogation : un attribut est représenté dans le modèle des graphes conceptuels par un type de concept, tandis qu'une valeur peut être représentée, selon les cas, par un type de concept ou un marqueur individuel. La notion de vue a été transposée dans le modèle des graphes conceptuels via la notion de graphe-schéma. Un graphe-schéma est un graphe conceptuel dont les marqueurs sont génériques et qui contient les attributs de sélection et les attributs de projection d'une vue. Il permet de lier entre eux, avec sens, les différents attributs d'une vue. Ces mécanismes de traduction ont été proposés dans Buche et Haemmerlé (2000) et étendus au flou dans Thomopoulos (2003).

Une requête, en termes de graphes conceptuels, est obtenue en instanciant le graphe-schéma correspondant à l'aide des valeurs de sélection spécifiées. Ainsi, la requête Q est traduite dans le modèle des graphes conceptuels par le graphe-requête de la figure 3. Le

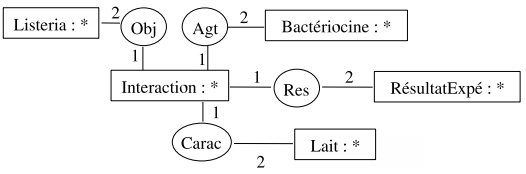

FIG. 3 – *Un exemple de graphe-requête*

graphe-requête est alors comparé à chaque graphe-donnée de la base de graphes conceptuels en utilisant l'opération de projection. Si le graphe-requête se projette dans le graphe-donnée, celui-ci est considéré comme une réponse à la requête de l'utilisateur. Les réponses obtenues dans les deux sous-systèmes sont ensuite regroupées et présentées à l'utilisateur sous la forme d'un tableau de résultats correspondant aux valeurs prises par les attributs de projection de la requête dans une donnée de la base relationnelle ou de la base de graphes conceptuels. Par exemple, le graphe conceptuel de la figure 1 est une spécialisation du graphe-requête de la figure 3 et donc une réponse à la requête Q. Il fournit le résultat suivant :

Substrat	***RésultatExpé.***
LaitEcrémé	Réduction

3 Le projet Grain Virtuel, une application sur les propriétés des grains de céréales

3.1 Objectif et données

L'étude des céréales en laboratoire a conduit à l'accumulation d'une grande quantité de données de nature et d'origine très variées, aboutissant au constat de la difficulté à réaliser des confrontations entre l'ensemble de ces travaux. L'objectif du projet, qui a débuté en 2003, est de construire un outil permettant de regrouper ces données pluridisciplinaires et d'en faciliter l'utilisation. Il s'agit plus précisément de représenter dans un même système des données concernant les propriétés morphologiques, histologiques, mécaniques et biochimiques

des grains de céréales, afin de pouvoir mettre en évidence les éventuelles relations entre des propriétés jusqu'ici étudiées de façon disjointe.

Les informations à stocker décrivent le produit initial et son origine, les opérations de préparation avant analyse, le type d'analyse ainsi que les conditions de réalisation de cette analyse, enfin le résultat de cette analyse, qui peut prendre des formes extrêmement variées : une ou plusieurs valeurs mesurées ou issues de calculs ou de publications, images, ...

De même que les types de résultats sont hétérogènes, les objets sur lesquels portent les analyses sont également multiples. Ainsi, six types d'objets analysés ont été identifiés : (i) l'échantillon, un ensemble de grains prélevés dans un lot ; (ii) la fraction, obtenue par un procédé technologique ; (iii) le tissu, portion histologiquement pure ; (iv) le grain ou individu ; (v) l'éprouvette, qui correspond à une ou plusieurs parties extraites d'un individu ; (vi) la zone locale, position précise d'une éprouvette. De plus, les filiations entre ces objets sont variables : (i) un échantillon provient d'un lot ; (ii) un grain provient d'un échantillon ; (iii) une éprouvette provient d'un grain ; (iv) une zone locale provient d'une éprouvette ; (v) une fraction provient d'un échantillon, d'une fraction, ou éventuellement du mélange de plusieurs objets pouvant être de natures différentes (échantillons, fractions, tissus) ; (vi) un tissu provient d'un échantillon ou éventuellement de l'accumulation de plusieurs éprouvettes.

Enfin, il est difficile de prévoir de façon exhaustive l'ensemble des propriétés à stocker pour chaque type d'objet, propriétés qui diffèrent d'un type d'objet à l'autre, mais qui dépendent aussi des orientations à venir de la recherche.

3.2 Recherche d'une solution souple fondée sur un formalisme unique, le modèle relationnel

Ici encore, le modèle relationnel a été choisi *a priori* pour représenter les données, d'une part pour des raisons d'homogénéité avec d'autres applications de l'équipe, d'autre part du fait de l'aspect multidimensionnel des données : ce type de données est généralement stocké physiquement soit dans des structures multidimensionnelles spécialisées, soit dans des bases relationnelles (Laurent et al., 2000). La base comporte une trentaine de tables et contient les résultats d'environ 3000 analyses.

Pourtant dans cette application, le polymorphisme des objets étudiés rappelle assez naturellement le modèle objet ; d'où l'idée d'aborder certains aspect de la modélisation à l'aide d'une approche objet, puis d'étudier les possibilités de traduction (ou "mapping") objet-relationnel pour passer au schéma relationnel. Notons qu'il existe des bases de données objet-relationnel implémentant l'héritage (telles que Postgresql par exemple), mais le modèle relationnel classique a été choisi ici pour des raisons d'indépendance vis-à-vis d'un système de gestion de base de données particulier.

Apports de l'approche objet. Concernant la variabilité des objets analysés, dans une approche objet l'utilisation de l'héritage facilite la représentation des informations. Ainsi, l'"Objet analysé" pourrait correspondre à une classe abstraite, comportant les attributs communs aux différents objets analysés et admettant différentes classes dérivées : "Echantillon", "Grain", "Eprouvette", "Tissu", "Fraction", "Zone locale". Les filiations entre grains, éprouvettes, etc., correspondent à des relations entre objets de la classe "Objet analysé".

Pour le mapping d'une hiérarchie d'héritage simple, telle que dans l'exemple proposé ci-dessus, trois solutions de traduction en relationnel sont principalement proposées dans la litté-

rature (voir par exemple Ambler (1997); Keller (1997)). Dans le cas d'une classe-mère abstraite C admettant des classes dérivées concrètes C'_i, ces solutions consistent respectivement à créer :
1. une table unique T pour l'ensemble de la hiérarchie d'héritage. T contient l'union des attributs de la classe C et des classes C'_i, et éventuellement un attribut spécifiant le type d'objet (c'est-à-dire la classe C'_i de l'objet). Cette solution a l'avantage d'être simple, mais implique la mise en commun de tous les attributs même si certains ne concernent qu'une seule classe de la hiérarchie ;
2. une table par classe de la hiérarchie d'héritage : la table T et les tables T'_i correspondent respectivement à la classe C et aux classes C'_i. Chaque table contient les attributs de la classe correspondante, seul l'identifiant étant un attribut commun (clé primaire dans T, clé primaire et étrangère dans les T'_i). L'intérêt de cette solution est sa conformité à la hiérarchie de classes, en revanche l'accès aux données est plus complexe du fait de la pluralité des tables ;
3. une table par classe concrète de la hiérarchie d'héritage : les tables T'_i correspondent respectivement aux classes C'_i. Chaque T'_i contient l'union des attributs de C et de C'_i ; les clés primaires des T'_i sont distinctes. L'avantage de cette solution est que chaque T'_i regroupe exactement les attributs nécessaires à la classe C'_i, en revanche elle ne permet pas la factorisation des propriétés communes aux T'_i, et notamment des relations communes que celles-ci peuvent avoir avec les autres tables de la base.

Solution retenue. La solution retenue, sur ce point, dans la construction du schéma relationnel du projet Grain Virtuel, est proche :
- de la deuxième solution de mapping présentée ci-dessus, du fait que nous avons créé une table "Objet analysé" correspondant à la classe-mère "Objet analysé" ;
- de la première solution de mapping présentée ci-dessus, du fait que nous avons regroupé les informations correspondant aux classes dérivées dans une même table appelée "Propriété". De plus la table "Objet analysé" contient, comme dans la première solution, un attribut spécifiant le type d'objet.

La présentation qui suit est simplifiée mais permet d'expliciter le choix réalisé (se référer à la figure 4). La table "Objet analysé" comprend un identifiant, un champ indiquant le type d'objet analysé (échantillon, grain, ...), ainsi que les attributs hérités, c'est-à-dire communs à tous les types d'objet (par exemple, la date d'obtention). Les attributs spécifiques à un type d'objet ne figurent pas dans cette table. Ces informations sont représentées dans la table "Propriété", qui comprend : (i) l'identifiant de l'objet analysé, en tant que clé étrangère ; (ii) le type de propriété que l'on veut décrire (masse, longueur, etc.), prenant ses valeurs dans une table de référence des types de propriétés ; (iii) un champ numérique permettant de stocker la valeur de la propriété décrite, si celle-ci est numérique ; (iv) un champ alphanumérique permettant de stocker la valeur de la propriété décrite, si celle-ci est symbolique ; (v) l'unité dans laquelle est exprimée la propriété décrite.

L'intérêt de cette solution est de faciliter l'ajout : (i) de nouveaux types d'objets analysés (soit de nouvelles classes héritées dans le modèle objet). L'ajout d'un nouveau type d'objet se traduit par une nouvelle valeur de l'attribut *typeObjet* ; (ii) de nouveaux types de propriétés (soit de nouveaux attributs de classes dans le modèle objet). L'ajout d'une nouveau type de propriété se traduit par une nouvelle valeur de l'attribut *typePropriété*.

Les filiations entre objet sont représentées dans une table "Filiation" (qui n'apparaît pas sur la figure 4). Elle comporte les champs *idObjet* et *idObjetPère*, clés étrangères référençant la clé primaire *idObjet* de la table "Objet analysé", ainsi qu'un attribut *proportion*. Elle permet de

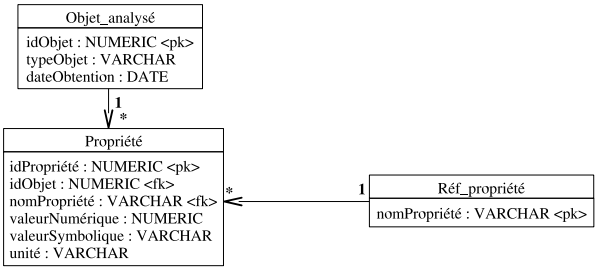

FIG. 4 – *Une partie de la solution adoptée dans le projet Grain Virtuel*

mémoriser de façon générique, pour chaque objet analysé, le ou les objets-pères dont il est issu – sans contrainte sur le type d'objet-père, échantillon, fraction, etc. – et en quelle proportion.

4 Discussion

Dans cette partie, nous commençons par situer les deux approches dans la problématique d'intégration de sources d'informations hétérogènes, puis nous proposons une comparaison selon les quatre critères suivants : facilité de mise en place, souplesse dans la représentation des données, caractère évolutif, facilité au croisement des informations.

4.1 Positionnement en intégration de sources hétérogènes

Dans les projets présentés, la première étape a consisté à définir une base de données permettant d'intégrer des informations issues de différentes sources en vue de leur interrogation et de leur exploitation. On distingue deux grands types d'approches pour l'intégration de sources d'informations hétérogènes :

L'approche **entrepôt de données** (Jarke et al., 2000) consiste à construire une base de données (l'entrepôt) regroupant dans un même schéma toutes les informations issues des différentes sources. Les requêtes sont posées directement sur l'entrepôt. Les problèmes posés par l'approche entrepôt de données concernent la définition du schéma de l'entrepôt, sa mise à jour ainsi que la saisie des données.

L'approche **médiateur** (Wiederhold, 1992) consiste à construire une interface entre l'utilisateur et les sources de données, donnant l'illusion d'interroger un système homogène et centralisé, tandis que les données restent au niveau des sources. Pour chaque source, un "wrapper" traduit les requêtes de l'utilisateur dans le langage spécifique à la source. Les problèmes posés par l'approche médiateur concernent la mise en place des "wrappers", des vues sur les sources de données et, selon les cas, la mise à jour du langage d'interrogation centralisée lors de l'ajout de nouvelles sources et la construction des réponses à une requête en combinant les éléments fournis par chaque source. On distingue le modèle d'intégration *Global as views* (GAV) (Chawathe et al., 1994), dans lequel le schéma médiateur est défini en fonction des schémas des sources à intégrer qui sont supposées connues, du modèle *Local as views* (LAV) (Rousset et al., 2002), où le schéma médiateur est défini indépendamment des sources de données qui ne sont pas connues *a priori*, mais décrites de manière déclarative au moment de leur

intégration. Les avantages et inconvénients des deux modèles d'intégration sont inversés : en GAV, la construction des réponses à une requête est simple, mais l'ajout de nouvelles sources de données peut remettre en question le schéma médiateur ; le modèle LAV, au contraire, est flexible quant à l'ajout de nouvelles sources, mais en contrepartie la construction des réponses à une requête nécessite une réécriture complexe. Dans les deux cas, le choix d'une approche médiateur implique la mise en place de "wrappers" et de vues sur les sources de données.

L'approche initiale adoptée dans les deux projets présentés a été une approche de type entrepôt de données, l'entrepôt étant constitué par une base de données relationnelle. L'approche médiateur était difficilement utilisable, du fait que de nombreuses informations se trouvaient sous forme de publications et n'étaient pas nécessairement disponibles dans un format informatisé exploitable.

Dans le projet Sym'Previus, les limites de l'approche entrepôt se sont traduites par une difficulté de mise à jour du schéma de l'entrepôt, ce qui a mené à la création d'une base complémentaire dans le formalisme des graphes conceptuels. Du fait de l'insertion de cette base complémentaire, l'approche utilisée dans le projet Sym'Previus devient une approche de type médiateur, schématisée sur la figure 2, le nombre de sources de données étant ici limité à deux. La solution Sym'Previus relève plutôt du modèle d'intégration GAV. En effet, bien que dans un premier temps le sous-système graphes conceptuels ait dû s'adapter au langage d'interrogation préexistant (ce qui correspond au modèle LAV), l'émergence d'informations tout à fait nouvelles dans le sous-système graphes conceptuels (concernant la composition des aliments en acides gras notamment) a nécessité la mise à jour du langage d'interrogation unifiée, afin de rendre ce nouveau type de données accessibles via le système d'interrogation (ce qui correspond au modèle GAV). De plus, dans le modèle Sym'Previus, les réponses à une requête n'utilisent pas de combinaison des informations issues des deux sources, contrairement aux systèmes LAV, qui combinent les éléments de réponse partielle issus des différentes sources pour répondre à une requête.

4.2 Comparaison des deux approches

Une première vision schématique de la comparaison des deux approches est proposée dans le tableau 4.2. Chaque critère de comparaison est ensuite nuancé.

Critère	**Formalisme mixte (Sym'Previus)**	**Formalisme unique (Grain Virtuel)**
Facilité de mise en place	-	+
Souplesse de représentation	+	-
Caractère évolutif	+	-
Croisement des informations	-	+

TAB. 1 – *Comparaison des deux approches*

Facilité de mise en place. Comme dans tout système médiateur, la solution Sym'Previus a nécessité la mise en place de "wrappers" traduisant les requêtes de l'utilisateur dans le langage spécifique à chaque source, ainsi que la définition de vues sur les données. En l'occurrence, la création d'un moteur de traduction des requêtes en termes de graphes conceptuels, utilisant les graphes-schémas, a constitué une charge importante. Pour les utilisateurs, ce double

formalisme est transparent lors de l'interrogation des données. En revanche, la saisie des données nécessite de se familiariser avec un formalisme supplémentaire. A cet égard, le modèle des graphes conceptuels a été bien accueilli par les utilisateurs, perçu comme un formalisme relativement intuitif et convivial.

Le projet Grain Virtuel offre l'avantage d'un formalisme unique, beaucoup plus simple à gérer. Notons que la question de la saisie des données, qui constitue souvent un point fort de l'approche médiateur où les données restent dans leur format d'origine, n' a été évitée ici dans aucune des deux applications. En effet, une grande partie des informations (publications papier, ...) ne se trouvait pas dans un format directement exploitable.

Souplesse dans la représentation des données. Dans la solution Grain Virtuel, l'objectif a été d'introduire, dans le formalisme structuré qu'est le modèle relationnel, des éléments de souplesse pour certaines parties du schéma dont la structure est la plus variable ou l'évolution la plus imprévisible. Il est difficile de prédire à long terme si les parties du schéma susceptibles de variabilité ont été identifiées de façon judicieuse et par conséquent si la structure proposée sera adaptée aux besoins. Cependant, à l'heure actuelle le schéma relationnel a convenu pour l'ensemble des données répertoriées (résultats bibliographiques, mesures effectuées en laboratoire, données de capteurs, calculs, ...). Le projet Sym'Previus bénéficie sur ce point à la fois des avantages d'un double formalisme, à savoir le choix du format de représentation le plus adapté à chaque donnée, et de la spécificité d'un modèle fondé sur les graphes, offrant une grande souplesse de représentation.

Caractère évolutif. De par sa structure de médiateur, le projet Sym'Previus peut relativement facilement évoluer vers l'intégration de nouvelles sources de données. Le modèle s'est d'ailleurs récemment enrichi par la suite d'une troisième base de données au format XML, conçue pour permettre l'intégration de données disponibles sur le Web. Cette évolution nécessite tout de même la mise place d'un nouveau "wrapper" et la définition des vues permettant d'interroger la nouvelle source de données. De plus, le modèle d'intégration relevant de l'approche GAV, la prise en compte de nouvelles sources, voire de nouveaux types de données dans les sources existantes, peut nécessiter une mise à jour du langage d'interrogation unifiée.

L'application Grain Virtuel peut également être amenée à évoluer, avec différentes options : par une mise à jour du schéma relationnel, qui peut selon les cas s'avérer coûteuse, comme mentionné en 2.2 pour le projet Sym'Previus ; par un passage vers une architecture de type médiateur, nécessitant la mise en place d'un schéma médiateur et d'un mécanisme de transcription des requêtes via des "wrappers" et des vues sur les sources de données ; par une passage vers un autre formalisme, impliquant notamment la migration des données. Ces évolutions sont coûteuses du fait qu'elles supposent une refonte du système.

Croisement des informations. La solution Sym'Previus n'est pas adaptée au croisement des informations. A l'heure actuelle, elle ne permet pas de combiner des informations issues des deux sous-systèmes, relationnel et graphes conceptuels. C'est un des éléments qui la distingue des approches LAV, où les différentes sources sont combinées pour fournir une réponse complète à une requête.

La gestion de données multidimensionnelles est un point fort de la solution Grain Virtuel, utilisant le seul modèle relationnel. Notons que, par rapport à une utilisation "classique" du modèle relationnel, les requêtes se complexifient. Les sélections sur les valeurs d'attributs (clause *where* dans une requête SQL) prennent le pas sur de simples projections (clause *select* dans une requête SQL). Malgré cette restriction, la récupération de plusieurs milliers de valeurs

de variables, dans le but d'effectuer des tests de corrélation, reste négligeable (de l'ordre de la seconde).

5 Conclusion et perspectives

Dans cet article, nous avons présenté deux modes de représentation de l'information appliqués à des domaines des sciences du vivant où les données manipulées sont faiblement structurées. Le premier système (Sym'Previus) s'appuie sur un double formalisme, structuré et faiblement structuré : le modèle relationnel et le modèle des graphes conceptuels. Ce dernier a été introduit pour sa flexibilité dans la représentation des données. Le système utilise une architecture de type médiateur pour interroger les deux formalismes de façon unifiée. Le second système (Grain Virtuel) s'appuie sur un formalisme unique structuré, le modèle relationnel, tout en essayant d'introduire des éléments de souplesse, inspirés de l'approche objet, pour certaines parties du schéma. Il relève de l'approche entrepôt de données.

La comparaison des deux systèmes montre que leurs avantages et inconvénients respectifs sont inversés. Le formalisme unique a l'avantage d'être facile à mettre en place et de permettre le croisement des informations, tandis que le formalisme mixte offre une plus grande souplesse dans la représentation des données et une plus grande facilité d'évolution.

Une problématique récurrente dans les deux projets est celle de la saisie des données contenues dans des publications. Compte tenu de l'hétérogénéité des données, il est difficile d'automatiser cette étape. La mise en œuvre d'un processus semi-automatique d'extraction de l'information apparaît comme une perspective fondamentale pour ces systèmes.

Références

Abiteboul, S. (1997). Querying semi-structured data. In *Proceedings of the 6th International Conference on Database Theory (ICDT'1997)*, pp. 1–18. Springer-Verlag.

Abiteboul, S., D. Quass, J. McHugh, J. Widom, et J. Wiener (1997). The Lorel query language for semistructured data. *Journal of Digital Libraries 1*(1), 68–88.

Ambler, S. (Ed.) (1997). *Building Object Applications That Work*. Cambridge University Press/SIGS Books.

Bray, T., J. Paoli, et C. Sperberg-McQueen (Eds.) (1998). *Extensible markup language (XML) 1.0*. Recommandation du W3C disponible sur http ://www.w3.org/TR/1998/REC-xml-19980210.

Buche, P., C. Dervin, O. Haemmerlé, et R. Thomopoulos (2005). Fuzzy querying of incomplete, imprecise and heterogeneously structured data in the relational model using ontologies and rules. *IEEE Transactions on Fuzzy Systems 13*(3).

Buche, P. et O. Haemmerlé (2000). Towards a unified querying system of both structured and semi-structured imprecise data using fuzzy views. In *Proceedings of the 8th International Conference on Conceptual Structures (ICCS'2000), Lecture Notes in Artificial Intelligence*, Volume 1867, Darmstadt, Germany, pp. 207–220. Springer.

Bukhman, Y. et J. Skolnick (2001). BiomolQuest : integrated database-based retrieval of protein structural and functional information. *Bioinformatics 17*(5), 468–478.

Chawathe, S., H. Garcia-Molina, J. Hammer, K. Ireland, Y. Papaconstantinou, J. Ullman, et J. Widom (1994). The TSIMMIS project : Integration of heterogeneous information sources.

In *Proceedings of the 16th Meeting of the Information Processing Society of Japan*, Tokyo, Japan, pp. 7–18.

Cherry, J., C. Adler, C. Ball, S. Chervitz, S. Dwight, E. Hester, Y. Jia, G. Juvik, T. Roe, M. Schroeder, S. Weng, et D. Botstein (1998). Sgd : Saccharomyces genome database. *Nucleic Acids Research 26*(1), 73–79.

Jarke, M., M. Lenzerini, Y. Vassiliou, et P. Vassiliadis (Eds.) (2000). *Fundamentals of Data Warehouses*. Springer-Verlag.

Keet, C. (2003). Biological data and conceptual modelling methods. *Journal of Conceptual Modeling 29*.

Keet, C. (2004). Conceptual modelling and ontologies for biology : experiences with the bacteriocin database. In *Interjoven 2004*, San Jose de las Lajas, Cuba.

Keller, W. (1997). Mapping objects to tables : A pattern language. In *Proceedings of EuroPLoP'1997*, Irsee, Germany.

Laurent, A., S. Gancarski, et C. Marsala (2000). Coopération entre un système d'extraction de connaissances floues et un système de gestion de bases de données multidimensionnelles. In *Actes des Rencontres Francophones sur la Logique Floue et ses Applications (LFA'2000)*, La Rochelle, France, pp. 325–332. Cepaduès éditions.

Raguenaud, C., M. Pullan, M. Watson, J. Kennedy, M. Newman, et P. Barclay (2002). Implementation of the Prometheus Taxonomic Model : a comparison of database models and query languages and an introduction to the Prometheus Object-Oriented Model. *Taxon 51*, 131–142.

Rousset, M., A. Bidault, C. Froidevaux, H. Gagliardi, F. Goasdoué, C. Reynaud, et B. Safar (2002). Construction de médiateurs pour intégrer des sources d'information multiples et hétérogènes : le projet PICSEL. *Revue I3 : Information - Interaction - Intelligence*.

Sowa, J. (1984). *Conceptual structures - Information processing in Mind and Machine*. Addison-Welsey.

Thomopoulos, R. (2003). *Représentation et interrogation élargie de données imprécises et faiblement structurées*. Ph. D. thesis, Institut national agronomique Paris-Grignon.

Wiederhold, G. (1992). Mediators in the architecture of future information systems. *Computer 25*(3), 38–49.

Zhong, Y., Y. Luo, S. Pramanik, et J. Beaman (1999). HICLAS : a taxonomic database system for displaying and comparing biological classifications and phylogenetic trees. *Bioinformatics 15*, 149–156.

Summary

This article presents two ways of representing information in the framework of life science issues. The first one, applied to predictive microbiology, is based on two formalisms, the relational model and the conceptual graph model, which are uniformly queried through the same interface. The second one, applied to cereal technologies, uses the relational model only. This paper describes the characteristics of the data and compares the modeling solutions chosen in both systems.