Intégration de données hétérogènes dans SenPeer

David Faye[*-**] — Gilles Nachouki[**] — Patrick Valduriez[***]

[*] Laboratoire d'Analyse Numérique et d'Informatique (LANI)
Université Gaston Berger de Saint-Louis
BP 234 Saint-Louis (SENEGAL)
David.Faye@univ-nantes.fr

[**] Laboratoire d'Informatique de Nantes Atlantique (LINA)
Université de Nantes
2 rue de la Houssinière , BP-92208, 44322 Nantes cedex 03 (FRANCE)
Gilles.Nachouki@univ-nantes.fr

[***] Equipe ATLAS (INRIA et LINA)
Patrick.Valduriez@inria.fr

RÉSUMÉ. Dans cet article, nous présentons SenPeer, un nouveau système Pair-à-Pair de gestion de données distribuées permettant le partage décentralisé et flexible de données se conformant à des modèles de données différents. SenPeer est un système de type Super-Pair reposant sur une organisation des pairs en domaines sémantiques et dans lequel les pairs peuvent publier des bases de données relationnelles, objets ou des documents XML. Chaque pair exporte ses données dans un formalisme pivot commun qui a une structure de graphe enrichi sémantiquement avec des mots-clés destinés à guider la découverte des correspondances entre les éléments des schémas. Les requêtes sont échangées dans un format interne commun, routées vers les pairs pertinents grâce à une sélection basée sur la notion d'expertise et enfin réecrites vers les pairs liés grâce aux correspondances sémantiques.

ABSTRACT. In this article, we present SenPeer, a new Peer-to-Peer data management system allowing the sharing of data conforming to various data models in a decentralized and flexible fashion. SenPeer has a Super-peer network topology based on an organization of peers in semantic domains and in which peers can publish XML documents, relational or object databases. Each peer exports its data in a common formalism which has a graph struture semantically enriched with a set of keywords in order to guide mappings discovery. Queries are exchanged in the network by the mean of a common query exchange language, routed towards promising peers by an expertise based selection of relevant and rewrited towards those relevant peers using the semantic mappings.

MOTS-CLÉS : Médiation de données, Similarité sémantique, Réseaux Pair-à-Pair, Traitement de requêtes.

KEYWORDS : Data mediation, Semantic similarity, Peer-to-Peer Networks, Query processing.

1. Introduction

Les systèmes Pair-à-Pair de partage de fichiers se sont illustrés par une description des nœuds par clés et une localisation des données par un routage de ces mêmes clés dans le réseau. Les systèmes Pair-à-Pair de gestion de données communément appelés PDMS (*Peer Data Management System*)[3] combinent quant à eux la technologie Pair-à-Pair et celle des bases de données distribuées et s'appuient sur une description sémantique des sources de données pour le traitement des requêtes. PeerDB[7], Edutella[6], APPA[1] ou encore Piazza[3] constituent des implémentations actuelles de PDMS. Cependant, la plupart de ces PDMS ne permettent pas le partage de données décrites par des modèles de données différents. Nous nous interesserons à ce dernier problème.

SenPeer est un système Pair-à-Pair de partage de données se conformant à différents modèles de données. Nous l'illustrons dans cet article au partage de données relatives à la mise en valeur de la vallée du fleuve sénégal où les acteurs sont des experts de différents domaines de compétence localisés dans différents pays (Sénégal, Mali, Mauritanie). SenPeer repose sur une organisation de type Super-Pair avec un regroupement des pairs par domaines sémantiques. Chaque pair publie des données décrites par un modèle conforme aux modèles de données relationnel, objet ou XML et dispose de son propre langage d'interrogation. Dans le but d'une médiation flexible, les données sont exportées dans un modèle pivot qui a une structure de graphe enrichi sémantiquement avec des mots-clés issus des schémas et destinés à guider la découverte des correspondances sémantiques. Les requêtes sont échangées dans un format interne commun, routées et réecrites vers les pairs pertinents grâce aux correspondances sémantiques.

Le reste de l'article est organisé comme suit. La partie suivante présente l'architecture de SenPeer. Les parties 3 et 4 décrivent les processus de médiation de données et de traitement des requêtes. Nous terminons par une conclusion et des travaux futurs.

2. Architecture du système SenPeer

SenPeer est un système de type Super-pair avec un regroupement des pairs par domaines sémantiques correspondant à des thèmes (hydrologie, activités agricoles, etc., pour le fleuve sénégal par exemple). Chaque pair publie des données de son domaine dans les formats suivant : données relationnelles ou objets, documents XML. Les pairs d'un domaine sont gérés par un super-pair auquel est rattaché un réseau sémantique expliquant les données du domaine. Cependant ces pairs peuvent décrire leurs données avec des vocabulaires différents. Au sein du chaque super-pair un gestionnaire de correspondances permet de générer les correspondances avec les autres domaines mais aussi celles entre les pairs du domaine. Elles sont stockées dans des matrices de correspondance et réutilisées pour

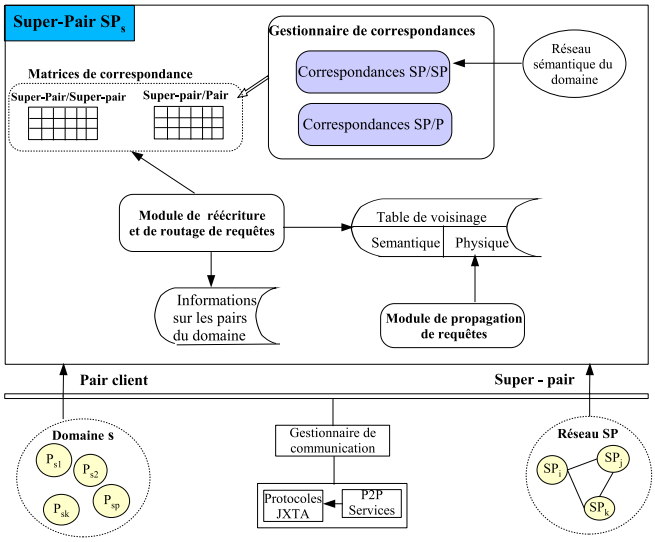

Figure 1. *Architecture d'un super-pair*

la réécriture des requêtes vers les pairs ou super-pairs pertinents. La communication est assurée par la plate-forme JXTA de Sun[5].

3. Médiation sémantique de données

Pour faciliter la découverte des correspondances en présence de plusieurs modèles de données, les schémas sont représentés dans un formalisme pivot appelé $sGraph$ (*semantic Graph*). Un $sGraph$ est un graphe orienté étiqueté et acyclique dont les nœuds correspondent aux éléments figurant dans les schémas (tables, colonnes, éléments ou attributs XML, etc.) et dont les arcs sont pris dans l'ensemble des relations sémantiques possibles entre éléments dans les langages des schémas : isA, $partOf$, $contains$, $typeOf$, $associates$, $dataType$, $sourceType$, $hasId$, $references$. De plus à chaque nœud n_i est associé un ensemble de synonymes $syn(n_i)$ issus des schémas et destinés à guider la découverte des correspondances sémantiques. Ces correspondances sont définies grâce à des mesures de similarités permettant de comparer deux nœuds.

La similarité $S(n_i, n_j)$ de deux nœuds appartenant à deux $sGraph$ SG_1 et SG_2 dépend de leurs similarités linguistique $S_l(n_i, n_j)$ et de voisinage $S_v(n_i, n_j)$, soit :

$$S(n_i, n_j) = \lambda_l.S_l(n_i, n_j) + \lambda_v.S_v(n_i, n_j) \quad \text{avec} \quad \lambda_l, \lambda_v \geq 0 \text{ et } \lambda_l + \lambda_v = 1 \qquad (1)$$

La similarité $S_l(n_i, n_j)$ évalue l'affinité linguistique de n_i et n_j en tenant compte des

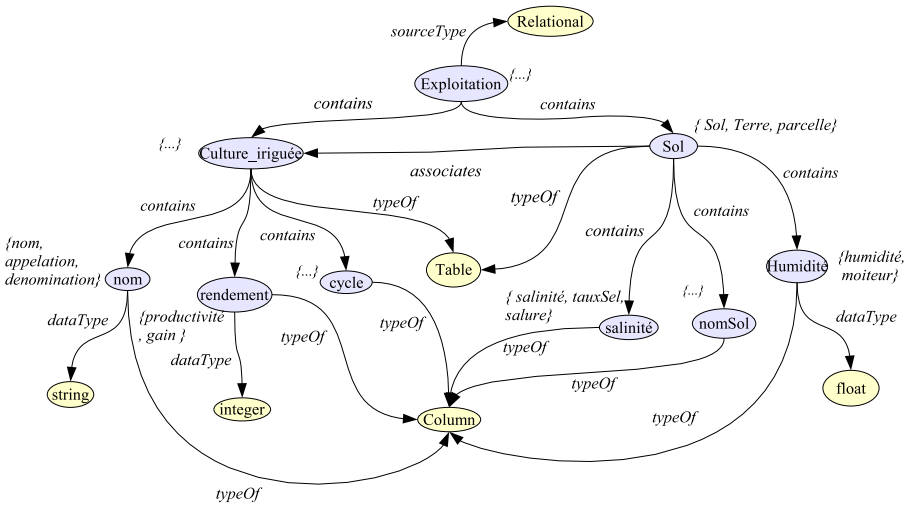

Figure 2. *$sGraph$ partiel sur les cultures pratiquées sur les sols du bassin du fleuve.*

similarités $S_s(n_i, n_j)$ des ensembles de synonymes et $S_t(n_i, n_j)$ de leurs types :

$$S_l(n_i, n_j) = \omega_s.S_s(n_i, n_j) + \omega_t.S_t(n_i, n_j) \quad \text{avec} \quad \omega_s, \omega_t \geq 0 \,\text{et}\, \omega_s + \omega_t = 1 \quad (2)$$

Les similarités de types sont prédéfinies dans l'intervalle $[0, 1]$. La similarité $S_s(n_i, n_j)$ se base sur la mesure de Tversky[9], mais nous faisons un appariement flou des synonymes, plutôt qu'un appariement exact. Etant donné A et B deux ensembles de termes leur intersection floue $A \cap_f B$ et leur différence floue $A -_f B$ se définit comme suit :

$$\begin{aligned} A \cap_f B &= \{a \in A / \max_{b \in B} SM(a, b) > \epsilon_{acc}\} \\ A -_f B &= \{a \in A / \max_{b \in B} SM(a, b) \leq \epsilon_{acc}\} \end{aligned} \quad (3)$$

avec SM fonction de comparaison lexicale entre deux mots et ϵ_{acc} seuil au dessus duquel la similarité calculée est considérée comme acceptable. La similarité $S_s(n_i, n_j)$ est alors :

$$S_s(n_i, n_j) = \frac{|syn(n_i) \cap_f syn(n_j)|}{|syn(n_i) \cap_f syn(n_j)| + \alpha|syn(n_i) -_f syn(n_j)| + (1-\alpha)|syn(n_j) -_f syn(n_i)|} \quad (4)$$

La similarité $S_v(n_i, n_j)$ compare les nœuds dans les voisinages $V(n_i)$ et $V(n_j)$. Elle dépend de la taille de leurs voisinages, mais aussi de l'intersection floue de ces voisinages.

$$S_v(n_i, n_j) = \frac{|\{x \in V(n_i) / \exists y \in V(n_j) \wedge S(x, y) > \epsilon_{acc}\}|}{|V(n_i) \cup V(n_j)|} \quad (5)$$

Si la similarité est supérieure à un certain seuil ϵ, nous avons une correspondance notée $< n_i, n_j, \cong >$ (nous ne considérons que l'équivalence $\cong$) pouvant être stockée dans deux types de matrices de correspondance : Super-pair/Pair (SP/P) et Super-pair/Super-pair (SP/SP). La figure 3 illustre le processus de médiation en présence des domaines non disjoints Agriculture et Pédologie. Par souci de lisibilité, les pairs de ce dernier domaine de sont pas représentés.

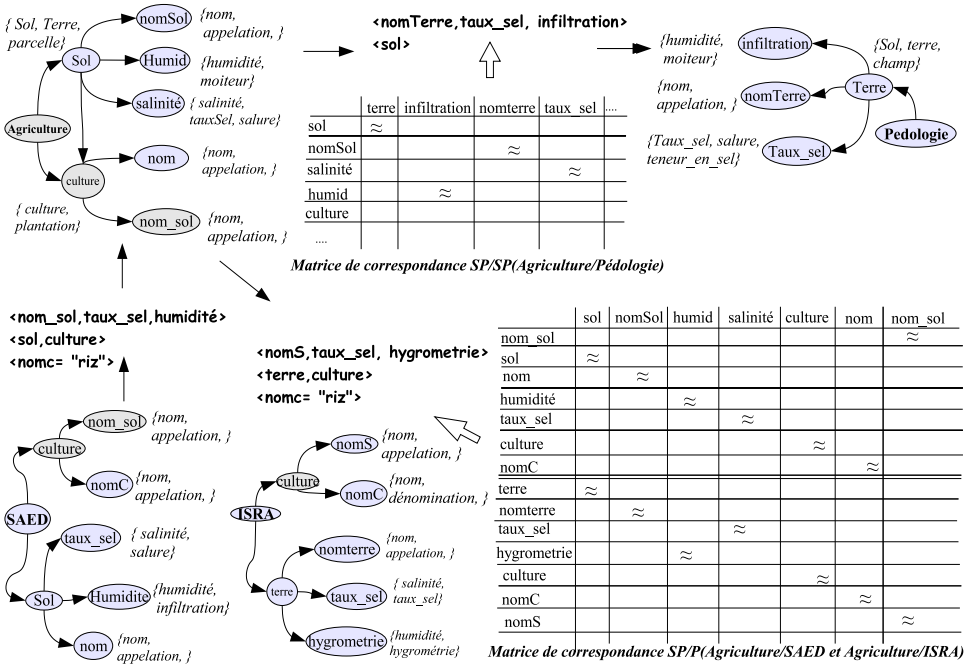

Figure 3. *Médiation de données des domaines Agriculture et Pédologie. ISRA(Institut Sénégalais de Recherche Agronomique), SAED(Société d'Aménagement et d'Exploitation du Delta).*

4. Traitement des requêtes

Processus d'interrogation

Quand un pair formule une requête avec son langage d'interrogation (SQL, XQuery, etc.) celle-ci est d'abord exécutée localement puis translatée dans le formalisme d'échange

de requêtes SQUEL(*SenPeer Query Exchange Language*) et enfin envoyée à son super-pair. Le super-pair extrait alors le sujet de la requête en se basant sur les informations contenues dans le $sGraph$ du pair actuel pour pouvoir le comparer aux descriptions d'expertises des pairs du domaine. Le résultat est la liste des pairs pertinents accompagnée de la réécriture sémantique de la requête en accord avec la matrice de correspondance SP/P. La requête peut maintenant être routée vers ces différents pairs. En suivant le même procédé, le super-pair détecte les domaines liés pertinents pour la requête en s'aidant cette fois-ci des expertises des super-pairs. La réécriture sémantique est faite dans ce cas avec la matrice de correspondance SP/SP.

Modèle de représentation interne de requête SQUEL

SQUEL s'inspire du modèle de requête QEM[8]. Une requête SQUEL est un arbre constitué de deux sous-arbres : sous arbre-condition et sous-arbre résultat. De plus pour faciliter la sélection des pairs pertinents pour la requête nous associons à chaque nœud racine, attribut ou sous-requête les synonymes du nœud correspondant dans le $sGraph$ du pair initiateur de la requête. Le Tableau 1 définit formellement une requête SQUEL.

$< QNode >$: := class **{**
$name$: <*word*>
$synonym$: {}|{<*syn-set*>}
$type$: <*typeNode*>
$constraint$: <*constraint*>
$return$: <*return*>
$operand$: <*operand*>**}**
<*syn-set*> : := <*word*> | <*word*>,<*synset*>
<*typeNode*> : :={} | $root$ | $subQuery$ | $operator$ |$function$| $attribute$ | $literal$
<*result*> : := {} | {$< QNode >$}
<*constraint*> : :={} | {$< QNode >$}
<*operand*> : := {} | {$< QNode >$}

Tableau 1. *Définition de la classe représentant un nœud d'une requête interne SQUEL.*

Si par exemple le pair SAED de la figure 3 exprime la requête SQL suivante :

```
SELECT nom_sol,taux_sel,humidite
FROM   culture,sol
WHERE  culture.nom_sol=sol.nom AND nomC='riz'
```

La requête enrichie exprimée en SQUEL sera alors celle représentée par la figure 4.

Description d'expertise de pair et sujet de requête

Chaque pair informe son super-pair des données qu'il partage à travers une description d'expertise automatiquement extraite de son $sGraph$. Soit un pair P de $sGraph$ SG, son

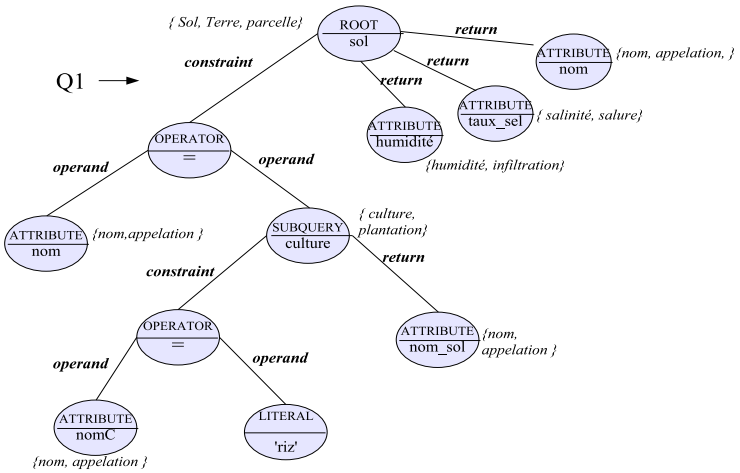

Figure 4. *Requête SQUEL correspondant à la requête SQL du pair SAED.*

expertise est l'ensemble de couples de nœuds de SG liés par la relation de contenance $contains$:

$$Exp(P) = \{(n_i, n_j) \in SG / contains(n_i, n_j)\}$$

Le sujet d'une requête Q est une abstraction de la requête en termes de couples de nœuds figurant dans l'arbre de la requête et référencés dans le $sGraph$ du pair initial :

$$Sub(Q) = \{(n_q, m_q) \in Q \mid return(n_q, m_q)\} \cup \{(n_q, m_q) \in Q \mid \exists\, o_q \in Q \wedge constraint(n_q, o_q) \wedge operand(o_q, m_q)\} \quad (6)$$

Par exemple, le sujet de la requête $Q1$ dans la figure 4 est :

```
Sub(Q1)={(sol,nom),(sol, taux_sel),(sol,humdite),(culture,nomC),
         (culture, nomsol)}
```

Sélection des pairs pertinents

L'algorithme de sélection prend en entrée une requête SQUEL et retourne un ensemble de sous-requêtes, chacune avec le pair pouvant la traiter. La sélection est basée sur une fonction map vérifiant l'inclusion du sujet de la requête dans l'expertise du pair.

$$map(Sub(Q), Exp(P)) = \begin{cases} vrai & \text{si } \forall\, (n_q, m_q) \in Sub(Q) \quad \exists (n_p, m_p) \in Exp(P) \mid \\ & Sim(n_q, n_p) \geq \epsilon_{acc} \ \wedge \ Sim(m_q, m_p) \geq \epsilon_{acc} \\ faux & \text{sinon} \end{cases}$$

Les résultats de l'algorithme de sélection des pairs pertinents seront utilisés pour générer les plans de requêtes distribuées. Ces plans de requêtes seront ainsi exécutés par les pairs ou super-pairs appropriés.

5. Conclusion

SenPeer permet le partage décentralisé de données de n'importe quel domaine, en présence de plusieurs modèles de données, à conditions que ces données soient classées par thèmes. La découverte des correspondances sémantiques est facilitée en autorisant les pairs à exporter leurs connaissances sous forme de modèle pivot enrichi sémantiquement avec des mots-clés introduits à la conception des schémas. Les requêtes sont échangées dans un modèle interne commun et réécrites vers les pairs pertinents grâce aux correspondances sémantiques. La sélection de ces derniers est basée sur une comparaison entre le sujet de la requête et l'expertise des pairs. Nos travaux futurs concerneront la configuration du système, la génération des plans de requêtes et la réécriture de celles-ci.

6. Bibliographie

[1] AKBARINIA R., MARTINS V., PACITTI E., VALDURIEZ P. « Replication and Query Processing in the APPA Data Management System » *Distributed Data & Structures 6 (WDAS) : (Lausanne, Switzerland), Waterloo* : Carleton Scientific, 2004.

[3] HALEVY A., IVES Z. G., MORK P. , TATARINOV I., « Piazza : Data Management Infrastructure for Semantic Web Applications », *In Proc. of the 12th Intl. World Wide Web Conf. (WWW2003) Budapest*, 2003.

[4] HEESE R., HERSCHEL S., NAUMANN F., ROTH A., « Self-Extending Peer Data Management », *In GI-Fachtagung fur Datenbanksysteme in Business, Technologie und Web (BTW 2005), Karlsruhe, Germany*, 2005.

[5] JXTA, « http ://www.jxta.org/ ».

[6] NEJDL W., WOLF B., QU C., DECKER S., SINTEK M. , NAEVE A., NILSSON M. , PALMER M., RISCH T. , « Edutella : A P2P networking infrastructure based on RDF », *In Proc. of the 11th Intl. World Wide Web Conf. (WWW2002), Hawaii, USA , May,* 2002.

[7] NG W. S., OOI B. C., TAN L. , ZHOU A., « PeerDB : A P2P-based System for Distributed DataSharing », *In Intl. Conf. on Data Engineering (ICDE2003) :, Bangalore, India* 2003.

[8] B. PANCHAPAGESAN , J.HUI, G. WIEDERHOLD. , G. WIEDERHOLD , S. ERICKSON, « The INEEL Data Integration Mediation System », *Proc. International ICSC Symposium on Advances in Intelligent Data Analysis (AIDA'99), Rochester, NY, June* 1999.

[9] TVERSKY A., EGENHOFER M. , RUGG R. , CONFORTI G. , « Features of similarity », *Psychological Review, 84, 327-352,* 1977.