# Fouille de bases de données hétérogènes pour alimenter le web de données Quel compromis entre qualité des données induites et robustesse des méthodes ?

Farid Cerbah

Dassault Aviation, Direction de la Prospective
farid.Cerbah@dassault-aviation.fr

**Résumé** : Le succès du web de données présuppose une capacité à produire des ressources interconnectées, en grande quantité, et dotées d'une sémantique explicite les rendant directement interprétables par des applications tierces. L'exploitation de bases de données existantes pour alimenter le web de données est une orientation qui s'est très vite imposée et qui a suscité des développements technologiques significatifs. Cependant, les modèles résultants héritent des faiblesses du modèle relationnel et, face aux exigences du web de données en matière de structuration sémantique de l'information, nombre de bases complexes ne s'avèrent être que des sources faiblement structurées. Une part conséquente de la sémantique que l'on souhaite rendre explicite reste enfouie dans les données stockés. Dans cet article, nous étudions différents motifs « sémantiques », identifiables dans les données stockées, dont l'exploitation peut apporter une amélioration sensible des ressources à inscrire dans le web de données et pour lesquels des méthodes robustes peuvent être élaborées. Nous prenons appui sur des développements récents menés autour de la plateforme RDBToOnto.

**Mots-clés** : Web de données, ontologies, bases de données relationnelles, fouille de données

## 1 Introduction

Le succès du web de données présuppose une capacité à produire des ressources interconnectées, en grande quantité, et dotées d'une sémantique explicite les rendant directement interprétables par des applications tierces. Pour répondre à des exigences d'ordre aussi bien qualitatif que quantitatif, l'exploitation de bases de données existantes pour alimenter le web de données est une orientation qui s'est très vite imposée et qui a ces dernières

années suscité des développements technologiques significatifs. Le groupe de travail W3C RDB2RDF a produit en 2009 un état des lieux représentatifs des principales contributions visant à produire des données interconnectées en RDF à partir de bases de données relationnelles (RDB2RDF, 2009). De par leur caractère structuré, les bases relationnelles sont en effet des sources à privilégier. On peut aisément produire des données RDF inscrites dans des modèles sémantiques calqués sur les schémas des bases. Toutefois, si les techniques automatiques de transformation exploitant les schémas se caractérisent par une bonne robustesse, force est de constater que les résultats ne sont en général pas à la hauteur des attentes. Les données produites héritent des faiblesses du modèle relationnel et, face aux exigences du web de données en matière de structuration sémantique de l'information, nombre de bases complexes ne s'avèrent être guère plus que des sources faiblement structurées. Une part conséquente de la sémantique que l'on souhaite rendre explicite reste bien souvent enfouie dans les données stockées.

Fort de ce constat, nous développons l'idée qu'en exploitant également le contenu des bases, il est possible de produire des données de meilleure qualité, conformes aux exigences du web de données en matière d'explicitation de la sémantique des données. Cependant, nous cherchons aussi à ne pas (trop) sacrifier pour autant à la robustesse des méthodes.

Nous mettons en oeuvre notre démarche dans la réalisation du logiciel RDBToOnto[1] (Cerbah, 2008). Notre principal objectif est d'élaborer des techniques de fouille de données fondées sur une bonne assise théorique et garantissant un bon niveau de robustesse. De plus, dans notre démarche, l'interactivité joue un rôle important avec le souci constant de trouver les moyens d'injecter de la connaissance *a priori* dans un processus qui doit rester largement automatisé.

Dans cet article, nous étudions différents motifs « sémantiques », identifiables dans les données stockées, dont l'exploitation peut apporter une amélioration sensible des ressources à inscrire dans le web de données et pour lesquels des méthodes robustes peuvent être élaborées. Nous prenons appui sur des développements récents menés autour de la plateforme RDBToOnto.

[1] `http://sourceforge.net/projects/rdbtoonto`

## 2 Méthodes d'extraction d'ontologies à partir de bases de données

L'acquisition automatique de modèles sémantiques ou d'ontologies à partir de bases de données est une problématique de recherche relativement récente. Cependant, elle s'inscrit en continuité de la problématique de rétro-ingénierie de modèles relationnels. L'objectif des travaux menés dans ce cadre était d'extraire des modèles conceptuels à partir de modèles relationnels. Une part importante des règles de transformation définies dans ce domaine précurseur restent pertinentes dans une perspective de construction d'ontologies et on retrouve les règles les plus fiables dans plusieurs approches ayant des ontologies pour cibles (Stojanovic *et al.*, 2002; Li *et al.*, 2005).

La plupart des approches, aussi bien en rétro-ingénierie de modèles qu'en acquisition d'ontologies, se contentent d'exploiter les méta-données issues du schéma relationnel. On note toutefois quelques tentatives visant à exploiter aussi les données. L'identification de corrélations entre tuples est envisagée dans (Tari *et al.*, 1997; Astrova, 2004) mais en ne considérant que les valeurs de clés (i.e. identifiant de tuples). Les rapports d'inclusion entre clés peuvent révéler des relations d'héritage. Il faut souligner qu'en pratique les règles basées sur ces corrélations sont peu productives car les motifs sous-jacents ne sont rencontrés que dans les bases de données dont la conception a été optimisée.

L'approche proposée dans (Lammari *et al.*, 2007) définit un processus de fouille qui n'est pas restreint aux seuls identifiants de tuples. Cette approche s'appuie sur une interprétation précise de la sémantique des valeurs nulles. Il s'agit pour ainsi dire de profiter des défauts de modélisation qui résultent de l'absence de constructions dédiées à l'héritage de concepts dans le modèle relationnel. Typiquement, lorsque toutes les instances (tuples) d'un concept complexe sont rassemblées dans une même relation, certains attributs peuvent s'avérer n'être pertinents que pour certains « sous-concepts ». Par exemple, dans une relation Employés sensée contenir tout l'effectif d'une compagnie aérienne, les attributs Heures de vols et Numéro de licence ne seraient renseignés que pour les entrées correspondant à des membres du personnel naviguant. Un partitionnement de la relation sur la base des valeurs nulles présentes dans ces attributs peut rendre explicite la structure hiérarchique sous-jacente.

## 3 Trouver des motifs de catégorisation dans les données

La production de données de qualité passe par une identification et une exploitation efficaces de motifs de catégorisation enfouis dans les données. Nous nous intéressons ici à des motifs qui sont à la base des méthodes de catégorisation mises en oeuvre dans RDBToOnto. Les méthodes et algorithmes présentés succinctement dans cette partie de l'article sont décrits de manière détaillée dans (Cerbah, 2010) et (Cerbah & Lammari, 2012).

### - Attributs catégorisants

Certains motifs particulièrement fréquents dans les bases de données relationnelles reposent sur des attributs introduits dans les tables pour catégoriser les tuples. La figure 1 montre un exemple où un attribut de ce type est exploité pour structurer les points d'accès d'un aéronef réunis dans une table.

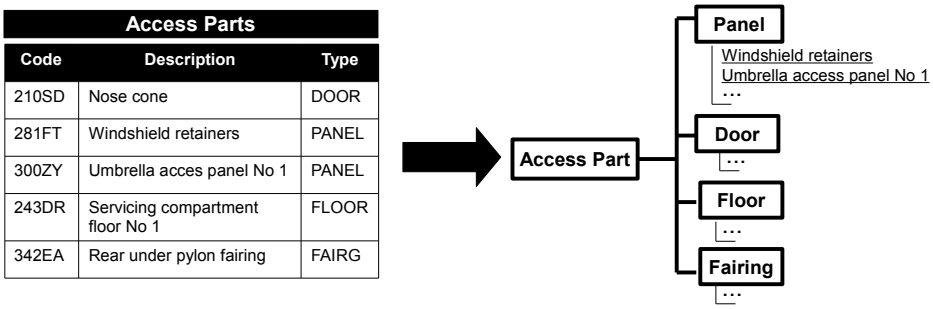


FIG. 1 – Une hiérarchie de classes dérivée d'un attribut catégorisant

Une méthode de transformation type n'exploitant pas ce type de motifs se contenterait de dériver une unique classe à partir de cette table et une instance à partir de chacun des tuples. Ces transformations simples s'avèrent souvent suffisantes mais la présence d'attributs catégorisants incite naturellement à dériver une structure de classes plus complexe comme illustrée sur cet exemple. La classe dérivée de la définition de la table dans le schéma de la base est enrichie de sous-classes dérivées des valeurs de la colonne Type. Deux attributs peuvent être impliqués dans la catégorisation. Par exemple, d'une table de fournisseurs contenant les attributs Pays et Ville, une structuration régionale à deux niveaux peut être induite en exploitant les valeurs de ces deux attributs catégorisants.

Pour exploiter ces motifs de catégorisation, deux opérations doivent être mises en oeuvre : le repérage des attributs pertinents dans les données et la construction et le peuplement des hiérarchies de classes associées.

RDBToOnto intègre un support étendu de ces motifs, avec une méthode de repérage des attributs pertinents mêlant deux sources d'information : des indices lexicaux susceptibles d'apparaître dans les noms d'attributs et une estimation de la récurrence des valeurs dans l'extension des attributs. Ces deux sources s'avèrent constituer de bons révélateurs du rôle catégorisant joué par certains attributs (Cerbah, 2010).

Une fois les bons attributs repérés, la construction et le peuplement de la hiérarchie de classes résultante s'effectue sans difficulté majeure. Le niveau de performance de ce processus dépend donc surtout de la phase de repérage des attributs catégorisants qui, selon nos évaluations (Cerbah, 2010) est relativement efficace. Cependant, cette phase reste une opération heuristique. A des fins de robustesse, RDBToOnto offre, outre l'implémentation rigoureuse d'une méthode automatisée, des moyens interactifs de sélection d'attributs catégorisants.

**- Hiérarchies à base terminologique**

On constate que de nombreuses bases techniques intègrent des listes de termes, souvent polylexicaux, à partir desquels des éléments de structuration peuvent être dérivés. Comme exemples, on peut citer, dans le domaine de la gestion de données techniques, les bases décrivant la structure hiérarchique de produits complexes où sont recensées dans les colonnes de tables des listes de composants (figure 2), ou encore dans le domaine pharmaceutique, des listes de médicaments.

Dans l'approche suivie dans RDBToOnto, nous considérons que :

– Ces listes de termes gagnent à être analysées pour en extraire les structures hiérarchiques sous-jacentes. Les techniques classiques d'analyse terminologique du domaine du traitement des langues peuvent être exploitées à cette fin (Bourigault *et al.*, 2001; Kageura *et al.*, 2004).
– Des hiérarchies de classes peuvent être dérivées plus ou moins directement des hiérarchies de termes.

Le passage au niveau « sémantique » implique la sélection des termes à introduire dans la structure résultante de classes et d'instances. Cette phase de sélection est à l'évidence un problème difficile. Elle ne peut être menée à bien sans une connaissance précise de la sémantique des termes manipulés.

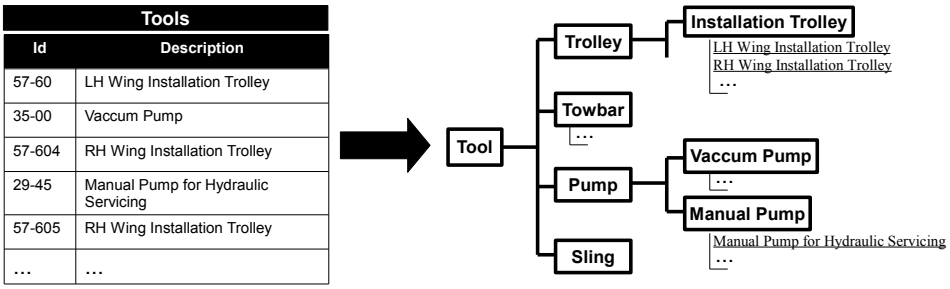


FIG. 2 – Construction d'une hiérarchie de classes à partir d'une liste de termes issue d'une base de données technique

La robustesse parait dès lors hors d'atteinte pour un processus fortement automatisé.

Dans RDBToOnto, sont mis à disposition plusieurs moyens interactifs aidant à orienter, aussi bien globalement que localement, le processus de production de la structure sémantique résultante. En particulier, pour agir globalement sur le potentiel des termes intermédiaires à être introduits dans la hiérarchie de classes, il est possible de fixer, pour chaque niveau de profondeur, le nombre de descendants requis. L'intérêt de cette heuristique d'ordre structurel est de favoriser des termes qui ont un fort potentiel structurant (estimé par le nombre de fils) pour fixer les classes intermédiaires. Pour agir localement, des listes de termes à préserver et à exclure peuvent être fournies en entrée.

Dans le processus supporté par RDBToOnto, l'analyse terminologique est une étape préalable, externe, qui doit produire une hiérarchie de termes encodée en SKOS. Nous donnons ci-après un extrait de fichier SKOS interprétable par RDBToOnto :

```
<rdf :RDF xmlns :rdf="http ://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xmlns :rdfs="http ://www.w3.org/2000/01/rdf-schema#"
        xmlns :skos="http ://www.w3.org/2004/02/skos/core#">
        ...
    <skos :Concept rdf :about="indicator">
        <skos :prefLabel xml :lang="en">indicator</skos :prefLabel>
        <historyNote>0</historyNote>
    </skos :Concept>
    <skos :Concept rdf :about="level_indicator">
```

```
        <skos :prefLabel xml :lang="en">level indicator</skos :prefLabel>
        <skos :broader rdf :resource="indicator"/>
        <historyNote>0</historyNote>
    </skos :Concept>
    <skos :Concept rdf :about="tank_level_indicator">
        <skos :prefLabel xml :lang="en">tank level indicator</skos :prefLabel>
        <skos :broader rdf :resource="level_indicator"/>
    </skos :Concept>
    <skos :Concept rdf :about="water_tank_level_indicator">
        <skos :prefLabel xml :lang="en">water tank level indicator</skos :prefLabel>
        <skos :broader rdf :resource="tank_level_indicator"/>
        <historyNote>234</historyNote>
    </skos :Concept>
        ...
</rdf :RDF>
```

Représentant la portion de hiérarchie de termes :

*indicator* $\longrightarrow$*level indicator* $\longrightarrow$*tank level indicator* $\longrightarrow$*water tank level indicator*

Dans cette description, les éléments de type HistoryNote permettent de désigner les entrées de la table source dont les termes sont issus. Ces informations de traçabilité sont ensuite exploitées pour intégrer les classes et instances dérivées de la structure de termes dans l'ontologie résultant de l'analyse de l'ensemble de la base (en particulier, par l'ajout de relations inter-classes).
Selon le paramétrage de l'algorithme de transformation, les deux termes intermédiaires pourront donner lieu à l'introduction de classes ou être simplement ignorés et le terme en bout de chaîne pourra être la source d'une classe ou d'une instance. Ce paramétrage est décrit plus en détails dans (Cerbah & Lammari, 2012).

## 4 Nommage des ressources et processus automatisé

Pour assurer une bonne réutilisabilité des ressources produites, il est essentiel d'adopter des motifs de nommage précis. Dans les sources relationnelles, ces motifs sont rarement explicites et l'éclatement des données dans de multiples tables ne facilite pas l'identification des entités manipu-

lées.

En général, les motifs de nommage intéressants sont à retrouver dans les données. Par exemple, dans une table réunissant les clients d'un service, le nommage des instances dérivées des entrées de cette table reposera de préférence sur un motif combinant les attributs prénom et nom. Les constituants d'un produit complexe pourront être nommés en suivant un motif impliquant la désignation type des composants, éventuellement associée au numéro de série.

L'identification automatique de motifs de nommage par fouille de données constituerait une fonction particulièrement utile qui, à notre connaissance, n'a pas été étudiée. Certains motifs récurrents semblent pouvoir être identifiés automatiquement avec une bonne efficacité, comme par exemple, les associations prénom - nom ou autres motifs basés sur des informations géographiques. Pour élaborer des procédés d'identification de motifs de nommage, il convient d'associer des profils de données types, récurrents, repérables dans des bases, à des ontologies de référence. Nous reviendrons en section 5 sur l'importance des modèles de référence, ayant vocation à être largement réutilisés, dans le processus de génération de ressources pour le web de données.

RDBToOnto ne permet que la définition interactive, assistée, de motifs de nommage. Les procédés de génération automatique d'instances assurent la bonne interprétation et l'application des motifs définis pour produire des URIs (ou des labels) non ambiguës.

## 5 Réutiliser des ontologies de référence

L'exploitation de bases existantes pour alimenter le web de données ne doit pas être réduit à un processus endogène qui se limiterait à extraire de la sémantique dans les données et méta-données des bases sources. Un gain substantiel en terme d'interopérabilité ne peut être acquis que par la mise en correspondance des éléments de modèle et données, souvent obtenus à partir de sources applicatives, avec des ontologies de référence qui ont statut de standards.

Les ontologies de référence à exploiter pour inscrire les modèles et données générées dans un cadre sémantique commun peuvent correspondre à des ontologies de portée générale, comme Foaf[2] ou Sioc[3], mais également à des standards plus spécialisés. Par exemple, le domaine aéronau-

[2] `http://xmlns.com/foaf/`

[3] `http://rdfs.org/sioc`

tique compte de nombreux standards qui intègrent des modèles de données structurés utilisés à des fins d'interopérabilité sémantique entre applications hétérogènes. Des ontologies peuvent être construites plus ou moins directement à partir de ces standards. A titre illustratif, nous présentons dans (Cerbah & Vatant, 2007) une tentative de formalisation d'un modèle de données issu de l'un des principaux standards de la maintenance aéronautique.

De part ses objectifs et les techniques applicables, la réutilisation d'ontologies de référence telle qu'elle se présente dans ce contexte rejoint en partie la problématique d'alignement d'ontologies (Euzenat & Shvaiko, 2007), en ce sens que les ontologies applicatives générées à partir de bases de données sont à apparier à des ontologies de référence.

Outre un accroissement de l'interopérabilité des modèles et données générés – ce qui constitue déjà un apport majeur – la réutilisation d'ontologies de référence peut entraîner un gain d'ordre structurel appréciable, en évitant notamment la répercution sur les modèles cibles de certains défauts de conception assez classiques dans les bases de données relationnelles. En particulier, il n'est pas rare d'observer dans les bases mal conçues la présence de tables qui gagneraient à être fragmentées, afin de séparer des entités sémantiquement distinctes, artificiellement réunies dans une même table (par exemple, une table dédiée aux commandes intégrant des attributs descriptifs des produits ou des clients concernés). Le recours à des ontologies de référence lors de l'analyse de telles sources peut aider à repérer ces défauts de conception et générer des modèles offrant une meilleure structuration sémantique de l'information.

Il nous semble clair qu'une amélioration conséquente résulterait de l'intégration de ces mécanismes d'appariement dans le processus de transformation, mais les automatiser avec robustesse apparaît comme un challenge à relever sur un plus long terme. Les tentatives d'automatisation ne sont pas légion, ou souvent réduites à l'identification de correspondances simples (Hu & Qu, 2007).

RDBToOnto offre la possibilité de réutiliser des ontologies de référence, mais restreinte à la spécification interactive d'appariement entre tables sources et classes ou propriétés[4] et entre colonnes et propriétés (les classes et propriétés étant issues d'ontologies externes prédéfinies). Le processus de transformation assure la bonne instanciation des classes et propriétés, en particulier la remontée des instances dans des hiérarchies com-

[4]plus précisément, des propriétés de type ObjectProperty.

binant des classes dérivées de la base source et des classes issues d'ontologies externes.

## 6 Générer des règles de mise en correspondance

Les méthodes et outils de génération de données RDF à partir de bases relationnelles s'inscrivent a priori dans un processus ETL[5], impliquant une extraction intégrale ou partielle, d'un seul tenant, a des fins de publication de données « semantisées ». La mise en oeuvre de techniques de mise en correspondance entre bases relationnelles et ontologies (Bizer, 2003; Barrasa *et al.*, 2004; Auer *et al.*, 2009; R2RML, 2012) est une thématique apparentée. Le but est de proposer des moyens déclaratifs pour associer des modèles relationnels à des ontologies prédéfinies et d'offrir des procédés d'instanciation à la volée des ontologies. Cette approche est mieux adaptée si les bases sources sont susceptibles de subir des mises à jour.

Dans ce cadre, il est aussi souhaitable de s'éloigner du modèle relationnel source pour inscrire les données dans un cadre sémantique plus riche. Cela passe par la définition de correspondances complexes. Malheureusement, le concepteur est peu assisté dans cette tâche, le support proposé étant souvent limité à la génération de règles élémentaires basées sur le schéma de la base.

Il est instructif de constater qu'en cherchant à générer des règles plus complexes, on rejoint *in fine* la problématique d'identification de motifs dans les données discutée précédemment et mise en oeuvre à travers RDBToOnto. Cette constatation nous a conduit à envisager sous cet angle les transformations opérées. En effet, en exploitant la traçabilité des décisions prises par RDBToOnto lors de la transformation d'une base, il est aisé de générer également des règles de correspondance dans une syntaxe donnée. Ainsi, il est possible avec cet outil de générer des configurations complètes, directement exécutables et incluant un ensemble de règles complexes, pour les interpréteurs D2R Server[6] et Triplify[7].

[5] ETL : Extract Transform Load

[6] `http://d2rq.org`

[7] `http://triplify.org`

## 7 Conclusion

Dans cet article, nous nous sommes intéressés à la problématique spécifique d'analyse de bases relationnelles comme moyen d'alimenter le web de données. Nous défendons l'idée qu'un palier qualitatif ne peut être franchi qu'en s'autorisant à rechercher des motifs sémantiques dans les données. Cela soulève des interrogations quant à la robustesse des méthodes envisageables. En prenant appui sur les méthodes implémentées dans la plateforme RDBToOnto, nous avons examiné plusieurs motifs pouvant être identifiés avec efficacité, par la mise en oeuvre de procédés automatiques ou semi-automatiques. Nous avons également discuté quelques pistes d'investigation, à plus ou moins long terme, visant à mieux identifier et exploiter ces motifs qui échappent en partie à la structuration formelle imposée par les méta-données.

Un prolongement naturel de cette approche consisterait à tirer davantage profit du contenu textuel souvent prédominant dans les bases hétérogènes (Mansuri & Sarawagi, 2006). A l'évidence, un enrichissement conceptuel significatif résulterait d'une exploitation étendue et efficace de ce matériau textuel, bien plus délicat à manipuler mais associé ici à des données structurées qui en précisent la sémantique. Sous l'angle de la robustesse, la question clé nous semble résider dans la capacité à adapter et à renforcer des techniques d'extraction d'information habituellement dédiées à des corpus textuels de structure plus libre.

## Références

ASTROVA I. (2004). Reverse engineering of relational databases to ontologies. In *The Semantic Web : Research and Applications, First European Semantic Web Symposium (ESWS 2004)*, Greece : Stringer-Verlag.

AUER S., DIETZOLD S., LEHMANN J., HELLMANN S. & AUMUELLER D. (2009). Triplify : light-weight linked data publication from relational databases. In J. QUEMADA, G. LEÓN, Y. S. MAAREK & W. NEJDL, Eds., *Proceedings of the 18th International Conference on World Wide Web, WWW 2009, Madrid, Spain, April 20-24, 2009*, p. 621–630 : ACM.

BARRASA J., CORCHO O. & GÓMEZ-PÉREZ A. (2004). R2O, an extensible and semantically based database-to-ontology mapping language. In *Second Workshop on Semantic Web and Databases (SWDB2004)*, Toronto, Canada.

BIZER C. (2003). D2R MAP - a database to rdf mapping language. In *Proceedings of WWW03*, Budapest.

BOURIGAULT D., JACQUEMIN C., L'HOMME M.-C. & (EDS) (2001). *Recent Advances in Computational Terminology*. John Benjamins.

CERBAH F. (2008). Learning highly structured semantic repositories from relational databases – the RDBToOnto tool. In *In The Semantic Web : Research and Applications  Proceedings of the 5th European Semantic Web Conference (ESWC 2008)* : Springer.

CERBAH F. (2010). Learning ontologies with deep class hierarchies by mining the content of relational databases. In F. GUILLET, G. RITSCHARD, D. A. ZIGHED & H. BRIAND, Eds., *Advances in Knowledge Discovery and Management*, Studies in Computational Intelligence. Springer.

CERBAH F. & LAMMARI N. (2012). Ontology learning from databases : Some efficient methods to discover semantic patterns in data. In J. VOLKER & J. LEHMANN, Eds., *Perspectives of Ontology Learning*. IOS Press, à paraître.

CERBAH F. & VATANT B. (2007). Building highly structured semantic repositories through reuse and formalisation of business standards. In *1st European Semantic Technology Conference*, Vienna.

EUZENAT J. & SHVAIKO P. (2007). *Ontology Matching*. Berlin Heidelberg (DE) : Springer-Verlag.

HU W. & QU Y. (2007). Discovering simple mappings between relational database schemas and ontologies. In *International Semantic Web Conference (ISWC 2007), 2nd Asian Semantic Web Conference (ASWC 2007)*, Busan, Korea.

KAGEURA K., DAILLE B., NAKAGAWA H. & CHIEN L. (2004). Recent trends in computational terminology. *Terminology*, **10**(2), 1–25.

LAMMARI N., COMYN-WATTIAU I. & AKOKA J. (2007). Extracting generalization hierarchies from relational databases. a reverse engineering approach. *Data and Knowledge Engineering*, **63**, 568–589.

LI M., DU X. & WANG S. (2005). Learning ontology from relational database. In *Proceedings of 2005 International Conference on Machine Learning and Cybernetics*, volume 6, p. 3410 – 3415 : IEEE.

MANSURI I. R. & SARAWAGI S. (2006). Integrating unstructured data into relational databases. In *Proceedings of the 22nd International Conference on Data Engineering (ICDE'06)*.

R2RML (2012). *R2RML : RDB to RDF Mapping Language*. Rapport interne, W3C.

RDB2RDF G. (2009). *A Survey of Current Approaches for Mapping of Relational Databases to RDF*. Rapport interne, W3C RDB2RDF Incubator Group.

STOJANOVIC L., STOJANOVIC N. & VOLZ R. (2002). Migrating data-intensive web sites into the semantic web. In *Proceedings of the ACM Symposium on Applied Computing (SAC 02)*, Madrid.

TARI Z., BUKHRES O. A., STOKES J. & HAMMOUDI S. (1997). The reengineering of relational databases based on key and data correlations. In *DS-7*, p. 184–.