Appropriate Global Ontology Construction: A Domain-Reference-Ontology Based Approach

Cheikh Niang
Université François Rabelais
Laboratoire d'Informatique
Tours, France
niangcat@yahoo.fr

Béatrice Bouchou
Université François Rabelais
Laboratoire d'Informatique
Tours, France
beatrice.bouchou@univ-tours.fr

Moussa. LO
Université Gaston Berger
LANI
Saint-Louis, Sénégal
moussa.lo@ugb.edu.sn

Yacine Sam
Université François Rabelais
Laboratoire d'Informatique
Tours, France
yacine.sam@univ-tours.fr

ABSTRACT

Data integration involves combining data residing in different sources and providing users with a unified view of these data through what is called a "global schema". We address here the problem of automatic construction of this global schema in the semantic Web context, where data sources are annotated with ontologies. We aim in other words to automatically build a common vocabulary (ontology) that will serve as a shared conceptual level for several heterogeneous data sources needing to share their data in a specific application domain. We propose a solution based on the use of a domain reference ontology (or "background knowledge") as a mediation support.

Keywords

Data integration, semantic Web, domain reference ontology, background knowledge.

1. INTRODUCTION

Data integration involves combining data residing in different sources and providing users with a unified view of these data. This process is important and appears with increasing frequency as the need to share existing data increases (in the commercial or biological domains for example). We address here the first data integration challenge pointed out in [16]: "How to build an appropriate global schema". Indeed, many organizations hold some similar data in specific domains and want to share some parts of it (merging databases of similar companies or combining research results from different bioinformatics sources for example). Data integration may then alleviate users from knowing the structure of different sources, as well as the way they are conciliated, when making queries [16], through the provision of a global schema. However, the question which arises is how to automatically construct an appropriate global schema for a given set of data sources?

In the context of semantic Web, for automation purpose such a global schema is generally represented by an ontology. Indeed, the number of data sources describing their data with local-ontologies is growing and the integrated access to heterogeneous data sources, called ontology-based data integration [28], is becoming a challenging issue. Ontologies offer a formal semantics which allows the automation of tasks such as heterogeneity resolution, consistency checking, inference, and global schema (ontology) construction. Our aim in this article is to show how one can automatically build an appropriate global ontology for several data sources owners that want to share parts of their data for a specific Web application, but that do not want to (or can not) invest much efforts on the hard task of building a consensual appropriate shared conceptual level. This integration process can be done upon data sources sharing a specific application domain where the domain itself is described with a background-knowledge or what is called "domain reference ontology". A domain-reference (or reference, called also mediator) ontology is an ontology developed independently from any specific objective by experts in knowledge engineering with the collaboration of domain experts. It is a robust conceptualization of the knowledge about a given generic domain such as medicine, tourism, agriculture, etc. AGROVOC[1] and NALT[2] in the agriculture domain and MeSH[3] in the medical field are some examples of reference ontologies. The development of semantic Web allows to expect that such reference ontologies will be formally represented and more and more accessible and usable by humans and by machines in the next few years.

[1] http://www.fao.org/agrovoc
[2] http://agclass.nal.usda.gov/agt
[3] http://www.nlm.nih.gov/mesh/

In this article, the underlying idea of our proposal is thus the use of a reference ontology to automatically build a global ontology that is appropriate to the sources to be integrated as well as to the target application domain (reference ontology). In our context, we consider that an appropriate global ontology *(i)* should provide an appropriate conceptualization of the application domain (maximizing relevant information for the sharing process and minimizing irrelevant one). Moreover, it has to *(ii)* allow easily adding/querying data sources and *(iii)* be automatically built and maintained.

We have integrated the above conditions to our solution (our algorithm for global-ontology construction). For an easy query processing, our algorithm lies on the Global As View [11] approach. However it generalizes existing proposals so that it is no longer necessary to have sources known in advance. An anchoring phase allows each source to participate in the global ontology to some extent, whatever it is. For easily adding data sources, it incrementally integrates data sources, so it is easy to add a new source involved in the sharing process. For an appropriate conceptualization, it selects the smallest relevant information portion from the reference ontology and only relevant information to be shared in the application domain from each data source involved in the sharing process.

The rest of this article is organized as follows: some preliminary notions are presented in Section 2 followed by our global-ontology-construction process in Section 3. Section 4 presents a case study and Section 5 some related works. We conclude and evoke some futures work in Section 6.

2. PRELIMINARIES

In this article, we consider that each ontology $\mathcal{O}$ is constructed with the following elements [19]:

– $\mathcal{C}$, a set of concepts, or classes;
– $\mathcal{I}$, a set of concepts' instances;
– $\mathcal{R}$, a set of binary relations defined on $\mathcal{C}$;
– $\mathcal{Z}$, a set of axioms, which can be interpreted as integrity constraints or relationships between instances and concepts, and which can not be expressed by the relations in $\mathcal{R}$.

We assume that the concepts of an ontology $\mathcal{O}$ are characterized by a finite set $\mathcal{A}$ of attributes, where each attribute $a \in \mathcal{A}$ has a domain values $\mathcal{V}_a$ (Integer, Literal, Date, etc) and $\bigcup_{a\in\mathcal{A}} \mathcal{V}_a = \mathcal{V}$ is the set of all domain values of the $\mathcal{O}$'s attributes. We accept in other words the following assumptions:

– *(i)* A concept is defined as a triplet (c, $\mathcal{A}^c$, $\mathcal{V}^c$) where c is the unique name of the concept, $\mathcal{A}^c$ / $\mathcal{A}^c \subseteq \mathcal{A}$ the set of attributes describing the concept and $\mathcal{V}^c$ / $\mathcal{V}^c \subseteq \mathcal{V}$ their domain values ($\mathcal{V}^c = \bigcup_{a\in\mathcal{A}^c} \mathcal{V}_a$).

The pair ($\mathcal{A}^c$, $\mathcal{V}^c$) is called the structure of the concept c. In this article, we call *generic concept* every concept with an empty structure.

– *(ii)* In an ontology $\mathcal{O}$, a set (may be a singleton) of relations can be defined between two concepts. If we denote by $R(c, c') = \{r_1(c, c'), ..., r_n(c, c')\}$ the set of the binary relations defined in $\mathcal{O}$ between c and c', then $\mathcal{R} = \bigcup_{c,c'\in\mathcal{C}} R(c, c')$.

In this paper, we explore more specifically the subsumption relation between two concepts of an ontology $\mathcal{O}$. This relation, denoted by ($\sqsubseteq$), means that if (c, $\mathcal{A}^c$, $\mathcal{V}^c$) and (c', $\mathcal{A}^{c'}$, $\mathcal{V}^{c'}$) are two concepts of $\mathcal{O}$ and ($c' \sqsubseteq c$) then:
– ($\mathcal{A}^c$, $\mathcal{V}^c$) $\subseteq$ ($\mathcal{A}^{c'}$, $\mathcal{V}^{c'}$).
– Ins($\mathcal{O}$,c') $\subseteq$ Ins($\mathcal{O}$,c), where Ins($\mathcal{O}$,c) denotes the set of instances belonging to the concept c in $\mathcal{O}$.
– if c'' is a concept of $\mathcal{O}$, then $R(c, c'') \subseteq R(c', c'')$.
– if c'' is a concept of $\mathcal{O}$ such that ($c'' \sqsubseteq c'$), then ($c'' \sqsubseteq c$).

As we will see in the next sections, we don't consider in this article the instance level of an ontology. That is why we don't characterize items $\mathcal{I}$ and $\mathcal{Z}$ above. Notice that this general formalization can correspond to any ontology expressed in a semantic web language like RDFS [17] or OWL [18].

3. AUTOMATIC BUILDING OF A GLOBAL ONTOLOGY

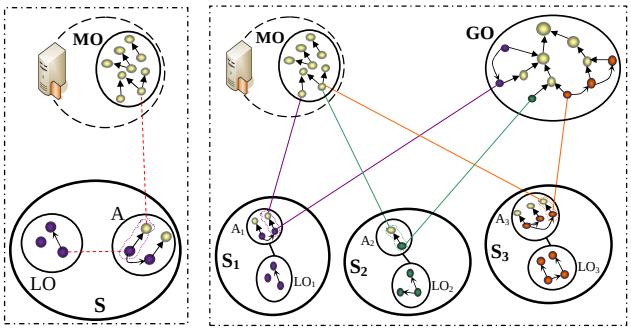
Our objective consists in automatically building a *global ontology* that provides a shared conceptual level for several data sources in a particular application domain. Our building process is done from the local ontologies and the domain reference ontology, that we also call mediator ontology. Hereafter, we mention the four kinds of ontologies that we deal with in the rest of this article.

- **Local ontologies** ($\mathcal{LO}_i$). Each source is represented by its own local ontology $\mathcal{LO}_i$ built from its data. Data sources can be heterogeneous and/or stored in different formats (structured, semi-structured, not-structured). We don't discuss here how these local ontologies are built, for this we refer to works realized by [5, 7] in this field. Also, we consider that a local ontology is represented only by its conceptual (intentional) level, so $\mathcal{LO}_i = (\mathcal{C}, \mathcal{R})$. The extensional level $\mathcal{I}$ (instances) is represented in the data sources, similar to what have been presented in [20].

- **The mediator (or domain reference) ontology** ($\mathcal{MO}$). It provides a general intensional knowledge on the application domain. It is usually composed by several disjoint hierarchies. Here, we exploit only a simple part of these hierarchies by considering that each hierarchy is a subsumption hierarchy composed only by generic concepts (*IsA* relations between generic concepts). Each data source uses $\mathcal{MO}$ to compute its agreement.

- **Agreements** ($\mathcal{A}_i$). We call agreement and we denote it by $\mathcal{A}_i = \langle \mathcal{LO}'_i, \mathcal{M}_i \rangle$ an ontology built automatically from a local ontology $\mathcal{LO}_i$ according to the mediator ontology $\mathcal{MO}$. It is composed of $\mathcal{LO}'_i$, a subset of $\mathcal{LO}_i$ containing knowledge of $\mathcal{LO}$ that are relevant for the application domain, and $\mathcal{M}_i$, a set of mappings between concepts of $\mathcal{LO}_i$ and those of $\mathcal{MO}$.

- **The global ontology** ($\mathcal{GO}$=$\langle \{\mathcal{A}_i\}, \mathcal{MO}' \rangle$). It consists in the set of agreements $\{\mathcal{A}_i\}$ together with $\mathcal{MO}'$, which is the smallest subset of $\mathcal{MO}$ that conciliates every $\mathcal{A}_i$ in $\mathcal{GO}$.



(a) Agreement (b) Conciliation

Figure 1: General overview of our mediation-based process

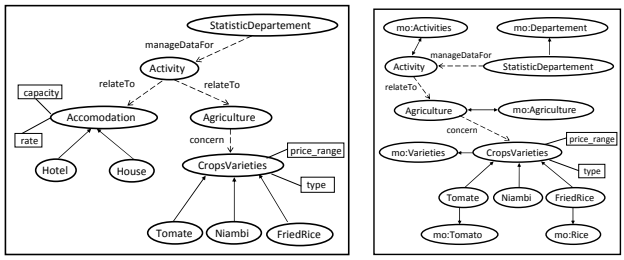
Figure 1 sketches the global ontology construction process: each source ($\mathcal{S}_i$) involved in the sharing process is represented by its local ontology ($\mathcal{LO}_i$) and the reference ontology ($\mathcal{MO}$) allows to find the portion of knowledge that each source can share with others. This portion is called agreement ($\mathcal{A}$ in Figure 1(a)). Then each agreement is incrementally integrated in the global ontology ($\mathcal{GO}$) via $\mathcal{MO}$ in what we call the conciliation phase (Figure 6(b)). Agreement and Conciliation processes will be detailed respectively in Sections 3.1 and 3.2.

3.1 Agreement process

Agreement process consists in the selection of knowledge fragments of $\mathcal{LO}$ to be included in the global ontology $\mathcal{GO}$. To identify such knowledge we proceed first by applying an *anchoring* process [2] to select from the local ontology relevant concepts for the application domain.

Anchoring consists in associating concepts of a local ontology, called *anchored concepts*, with concepts of the mediator ontology, called *anchor concepts*. Consider the example shown in Figure 2, where concepts are represented by ovals and attributes by rectangles. The single and double full arrows represent respectively subsumption and equivalence relationships between two concepts; simple binary relations are represented by dashed arrows. Figure 2(a) shows an excerpt of a local ontology $\mathcal{LO}$ that deals with both agricultural and accommodation knowledge. We assume that the application domain in which the source represented by $\mathcal{LO}$ shares its data is the agricultural domain: Figure 2(b) shows an excerpt of the agreement obtained after the anchoring process. Prefix "*mo* :" denotes anchor concepts from the mediator ontologie $\mathcal{MO}$. We can notice that only concepts related to agriculture are anchored because no anchor is found for accommodation knowledge. Anchor concepts generalize anchored concepts and will be used for finding semantic links between concepts in different local ontologies.

We perform two successive anchoring steps: a lexical anchoring process that selects relevant concepts to be anchored based on syntactic matching, followed by a semantic one that selects other concepts not-detected in the first step.



(a) Local Ontology (b) Agreement

Figure 2: An example of the agreement process

3.1.1 *Lexical anchoring process*

It consists in matching a local ontology $\mathcal{LO}$ with the mediator ontology $\mathcal{MO}$, *i.e.* in computing a set of mappings as defined in [25].

Let $\mathcal{LO}$ be a local ontology, $\mathcal{MO}$ be the mediator ontology, and $\mathcal{C}_l$, $\mathcal{C}_m$ the respective concept sets of $\mathcal{LO}$ and $\mathcal{MO}$. Lexical anchoring of $\mathcal{LO}$ w.r.t. $\mathcal{MO}$ consists in finding a set of mappings $\mathcal{M} = \langle m_1, ..., m_n \rangle$ such that each m_i is a relation of the form: $m_i = (c_l\ r\ c_m)$, where $c_l \in \mathcal{C}_l$, $c_m \in \mathcal{C}_m$, and r is a subsumption ($\sqsubseteq$) or equivalence ($\equiv$) relation between c_l and c_m. c_m is called the anchor of c_l and denoted by $anc(c_l)$. In what follows, we use the following notations: $Anc(\mathcal{C}_l)$, $Anc^-(\mathcal{C}_l)$, $\mathcal{C}_\mathcal{M}$, and $\mathcal{R}_\mathcal{M}$ for denoting respectively:

- anchored concept set of $\mathcal{LO}$, $Anc(\mathcal{C}_l) \subseteq \mathcal{C}_l$;
- anchor concept set of $\mathcal{LO}$, $Anc^-(\mathcal{C}_l) = \bigcup_{c_l \in \mathcal{C}_l} anc(c_l)$;
- concept set of $\mathcal{M}$, $\mathcal{C}_\mathcal{M} = (Anc(\mathcal{C}_l) \cup Anc^-(\mathcal{C}_l))$;
- relation set of $\mathcal{M}$, $\mathcal{R}_\mathcal{M} = \bigcup_{c_l \in \mathcal{C}_l, c_m \in \mathcal{C}_m} (c_l\ r\ c_m)$, where $r \in \{\sqsubseteq, \equiv\}$.

The key point in the lexical anchoring (or matching) process is to measure how much a concept c_l in a local ontology $\mathcal{LO}$ is related to a concept c_m in the mediator ontology $\mathcal{MO}$. This is done by syntactically comparing concepts names (labels). Many lexical similarity measures, proposed in the literature [15, 6, 25], may be used and, as noticed in [25], no similarity measure can give good results in all cases: it is still necessary to look for the best one for each specific application. However, whatever the application is, the relation between c_l and c_m is obtained as follows. If we consider that φ ($\varphi : \mathcal{C}_l \times \mathcal{C}_m \rightarrow [0, 1]$) is the chosen similarity measure, then if $\forall c_{mi} \in \mathcal{C}_l\ [\varphi(c_l, c_m) \geq \alpha\ \wedge \varphi(c_l, c_m) \geq \varphi(c_l, c_{mi})]$ holds, where α is the maximum threshold similarity, then the mapping $m = (c_l \equiv c_m)$ is established between c_l and c_m. If any anchor concept c_m is not found for a concept c_l, then we apply a partial matches as well, if c_m is a concept that has a label consisting of a superset of words of the label of the concept c_l, then we conclude that $m = (c_l \sqsubseteq c_m)$. In other words, we use the partial lexical matches following the intuition that additional words in a label additionally constrain the meaning of that concept. Doing so, one can conclude for example that *StatisticDepartement* $\sqsubseteq$ *Departemen* in Figure 2.

3.1.2 *Semantic anchoring process*

It consists in finding additional local concepts that may be relevant for the application domain and which have not been anchored during the lexical anchoring process. To identify

such concepts, we apply the following deduction rule: if c_l and c'_l are two concepts of $\mathcal{LO}$ such that

$$(c_l \in Anc(\mathcal{C}_l)) \wedge \ (c'_l \notin Anc(\mathcal{C}_l)) \wedge \ (c'_l \sqsubseteq c_l)$$

holds then we conclude that $c'_l \in Anc(\mathcal{C}_l) \wedge \text{anc}(c'_l) = anc(c_l)$.

The semantic meaning of this rule is well defined and corresponds to the subsumption reasoning mechanisms [22]. Indeed, $c'_l \in Anc(\mathcal{C}_l)$ means there is a subsumption relation $(c_l \sqsubseteq c_m)$ between c_l and its anchor c_m, so the subsumption relation $(c'_l \sqsubseteq c_l)$ that exists between c'_l and c_l in $\mathcal{LO}$ allows to deduce the subsumption relation $(c'_l \sqsubseteq c_m)$ between c'_l and c_m. Therefore, c'_l can be considered as an anchored concept, semantically selected, and its anchor is c_m. For example, there are no direct anchors found for the concept *Niambi* in Figure 2(a), but it appears in Figure 2(b) because *Niambi* is a sub-concept of *CropsVarieties*, which is anchored by the concept *mo:Varieties*.

3.1.3 From anchoring to agreement

After the anchoring process, we built the agreement $\mathcal{A}$ of each source. It is an ontology composed by $\mathcal{LO}'$, a subset of $\mathcal{LO}$ containing knowledge fragments of $\mathcal{LO}$ that are relevant for the application domain, and $\mathcal{M}$ the result of anchoring $\mathcal{LO}$ w.r.t. $\mathcal{OM}$. We compute $\mathcal{LO}'$ from anchored concepts of $\mathcal{LO}$, with the following purposes: $\mathcal{LO}'$ contains respectively the *(i)* maximum of relevant (*(ii)* the minimum of irrelevant) knowledge w.r.t. the application domain, and *(iii)* $\mathcal{LO}'$ is consistent if $\mathcal{LO}$ is consistent.

Thus, in addition to anchored concepts, $\mathcal{LO}'$ may contain unanchored concepts that we call *selected concepts*. A selected concept c_{l_1} is an unanchored concept that must be related to an anchored concept c_l in order to avoid loosing information about c_l and also to avoid inconsistency in $\mathcal{LO}'$. We consider that an unanchored concept c_{l_1} of $\mathcal{LO}$ must be a selected concept if:

– $(c_l \sqsubseteq c_{l_1})$, where c_l is an anchored concept of $\mathcal{LO}$.

– there exists in $\mathcal{LO}$ a relation $r(c_l, c_{l_1})$ between c_l and c_{l_1}, where c_l is an anchored concept of $\mathcal{LO}$.

– $(c_{l_1} \sqsubseteq c_{l_2})$, where c_{l_2} is a selected concept of $\mathcal{LO}$.

For instance, consider the local ontology $\mathcal{LO}$ shown in Figure 3(a) and assume that the concept *Agriculture* is an unanchored concept, as shown in Figure 3(b). Because we have in $\mathcal{LO}$ the indirect relation: *(Activity, relateTo, Agriculture)* and *(Agriculture, concern, CropsVarieties)* between the two anchored concepts *Activity* and *CropsVarieties*, it is necessary to select the concept *Agriculture* in order to keep it in $\mathcal{LO}'$. In what follows, we denote by $Select(\mathcal{C}_l)$ the selected-concepts set from $\mathcal{LO}$.

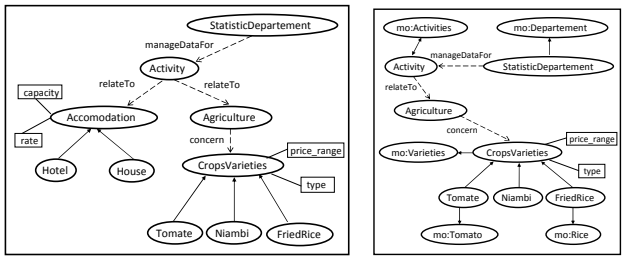
Let $\mathcal{LO}$ be a local ontology and $\mathcal{MO}$ be the mediator ontology, the agreement $\mathcal{A} = \langle \mathcal{LO}', \mathcal{M} \rangle$ of $\mathcal{LO}$ w.r.t $\mathcal{MO}$ is such that *(i)* $\mathcal{M} = \langle m_1, ..., m_n \rangle$ is the result of the anchoring of $\mathcal{LO}$ w.r.t $\mathcal{MO}$, and *(ii)* $\mathcal{LO}' = (\mathcal{C}'_l, \mathcal{R}'_l)$ is inductively defined as follows:

– $[\mathcal{C}_\mathcal{M} \cup Select(\mathcal{C}_l)] \subseteq \mathcal{C}'_l$

– $\mathcal{R}_\mathcal{M} \subseteq \mathcal{R}'_l$

– If $\exists\ c_{l_1}, c_{l_2}\ /\ c_{l_1} \in Anc(\mathcal{C}_l) \wedge c_{l_2} \in \mathcal{C}_l \wedge (c_{l_2} \sqsubseteq c_{l_1})$ then $(c_{l_2} \sqsubseteq c_{l_1}) \in \mathcal{R}'_l$.

– If $\exists\ c_{l_1}, c_{l_2}\ /\ c_{l_1} \in [Anc(\mathcal{C}_l) \cup Select(\mathcal{C}_l)] \wedge c_{l_2} \in \mathcal{C}_l \wedge (c_{l_1} \sqsubseteq c_{l_2})$ then $(c_{l_1} \sqsubseteq c_{l_2}) \in \mathcal{R}'_l$.



(a) Local Ontology (b) Agreement

Figure 3: Agreement process – the case of selected concepts

– If $\exists\ c_{l_1}, c_{l_2}, r\ /\ c_{l_1}, c_{l_2} \in [Anc(\mathcal{C}_l) \cup Select(\mathcal{C}_l)] \wedge\ r(c_{l_1}, c_{l_2}) \in \mathcal{R}_l$ then $r(c_{l_1}, c_{l_2}) \in \mathcal{R}'_l$.

Notice that all the above rules are designed to keep in $\mathcal{LO}'$ as much semantic information contained in $\mathcal{LO}$ as possible.

3.2 Conciliation process

We can now build the global ontology $\mathcal{GO}$ by conciliating the different agreements $\mathcal{A}_i = \langle \mathcal{LO}'_i, \mathcal{M}_i \rangle$ obtained above. The conciliation is achieved incrementally by integrating the agreements into $\mathcal{GO}$, one after another. Integrating an agreement $\mathcal{A}$ in $\mathcal{GO}$ consists in linking its concepts with the ones of other agreements already conciliated in $\mathcal{GO}$. Links between concepts in $\mathcal{A}$ are established through anchor concepts contained in $\mathcal{M}_i$ for every agreement $\mathcal{A}_i$. Let us recall that all anchor concepts are part of the mediator ontology $\mathcal{MO}$. Thus, we search for links between anchor concepts in $\mathcal{MO}$ in order to use them to conciliate concepts in $\mathcal{GO}$. In this way, our global ontology $\mathcal{GO}$ contains the following components:

– the set of agreements $\mathcal{A}_i = \langle \mathcal{LO}'_i, \mathcal{M}_i \rangle$. They represent the shared part of local ontologies $(\mathcal{LO}'_i)$, together with the mappings between their local concepts and the mediator ones $(\mathcal{M}_i)$.

– an as small as possible subset $\mathcal{MO}'$ of $\mathcal{MO}$ containing only the part of the hierarchy which is usefull to link local concepts.

To illustrate this process in the context of agricultural domain, consider the example in Figure 4. In this example the concepts *Tomate* of the agreement $\mathcal{A}_1$ and *FriedRice* of the agreement $\mathcal{A}_2$ are respectively anchored by the concepts *Tomato* and *Rice* of the mediator ontology $\mathcal{MO}$. The structure of the mediator ontology reveals that *Tomato* and *Rice* have a common ancestor which is the concept *Plan_products*. We reproduce this relation to conciliate the concepts *Tomate* and *FriedRice* in the global ontology $\mathcal{GO}$.

Let $\{\mathcal{LO}_i\}$ be a set of local ontologies and $\mathcal{MO}$ be the mediator ontology. The corresponding global ontology $\mathcal{GO}$ is $\langle \{\mathcal{A}_i\}, \mathcal{MO}' \rangle$, where *(i)* $\{\mathcal{A}_i\}$ is the set of agreements built from local ontologies, and *(ii)* $\mathcal{MO}'$ the smallest subset of $\mathcal{MO}$ that conciliates every $\mathcal{A}_i$ in $\mathcal{GO}$, built by the algorithm that we will present in the sequel of this section.

As suggested by the Figure 4 example, one particular interest in our approach is the use of the hierarchy of the me-

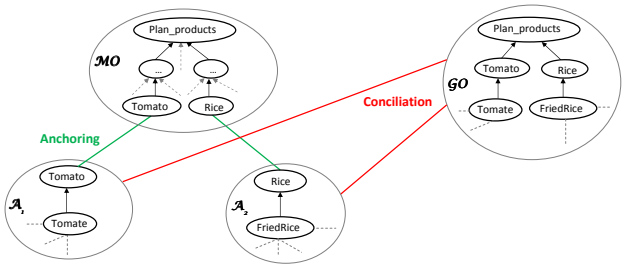


Figure 4: Illustration of the conciliation process

diator ontology $\mathcal{MO}$ in order to find links between anchor concepts. These links are reproduced in the global ontology $\mathcal{GO}$ for conciliating agreements.

The relation that we are looking for within the hierarchy $\mathcal{H}$ of $\mathcal{MO}$ is the *least common subsumer* (*lcs*) of two anchor concepts. It is important to notice that in our first experiments we have only considered tree taxonomies, we plan as a future work to generalize this point. We can follow the algorithm proposed in [3] to compute the *lcs* of two concepts c_1 and c_2 in $\mathcal{MO}$, according to what follows.

Let $\mathcal{MO}$ be the mediator ontology, c_1 and c_2 two given concepts in $\mathcal{MO}$, the concept c of $\mathcal{MO}$ is the lcs of c_1 and c_2 in $\mathcal{MO}$ (noted $c = lcs_{\mathcal{H}}(c_1, c_2)$) *iff* *(i)* $\forall i/i = 1, 2 \;\; (c_i \sqsubseteq c)$, and *(ii)* if $\exists c' \in \mathcal{MO}$ / $(c_1 \sqsubseteq c') \wedge (c_2 \sqsubseteq c')$ *then* $c \sqsubseteq c'$.

Based on *lcs* computation in [3], $\mathcal{MO}'$ consists in a subsumption hierarchy between all anchor concepts of all $\mathcal{A}_i$ and their *lcs* in $\mathcal{MO}$. The algorithm that we propose to achieve this uses the hierarchical proximity measure proposed in [29], that we recall hereafter.

Let $\mathcal{MO}$ be the mediator ontology, c_1 and c_2 two concepts of $\mathcal{MO}$. The hierarchical proximity measure between c_1 and c_2 in $\mathcal{MO}$ is such that:

$$sim_{\mathcal{H}}(c_1, c_2) = \frac{2 * depthOf(lcs_{\mathcal{H}}(c_1, c_2))}{depthOf(c_1) + depthOf(c_2)},$$

where $depthOf(c)$ returns the number of subsumers of c in $\mathcal{MO}$.

Moreover, let $c, c' \in \mathcal{MO}$. We say that c' is the closest concept of c in $\mathcal{MO}$ and we denote it by $closest_{\mathcal{H}}(c)$ *iff* $\forall i/c_i \in \mathcal{MO}$ $[sim_{\mathcal{H}}(c, c') \geq sim_{\mathcal{H}}(c, c_i)]$ holds. However, if $c' \in \mathcal{MO}'$ $(\mathcal{MO}' \subseteq \mathcal{MO})$ and $\forall i/c_i \in \mathcal{MO}'$ $[sim_{\mathcal{H}}(c, c') \geq sim_{\mathcal{H}}(c, c_i)]$ holds, then we say that c' is the closest concept of c in $\mathcal{MO}'$ w.r.t. $\mathcal{MO}$ and we denote it by $closest_{\mathcal{H}'/\mathcal{H}}(c)$.

The conciliation of an agreement $\mathcal{A}_k = \langle \mathcal{LO}'_k, \mathcal{M}_k \rangle$ with other agreements already conciliated in $\mathcal{GO} = \langle \{\mathcal{A}_i\}_{i \neq k}, \mathcal{MO}' \rangle$ consists in integrating each anchor concept c_m of $\mathcal{M}_k$ within the hierarchy of $\mathcal{MO}'$. To integrate a concept c_m within the hierarchy of $\mathcal{MO}'$ we have to compute the *lcs* in $\mathcal{MO}$ between c_m and the closest concept of c_m in $\mathcal{MO}$ among the anchor concepts already present in the hierarchy of $\mathcal{MO}'$.

However, when the mediator ontology $\mathcal{MO}$ is very large, which is frequently the case, this process can be hard and costly. In addition, as said before, our mediation ontology has a structure composed by disjoint hierarchies, so two different anchor concepts do not have necessarily the same root. This configuration can make the *lcs* computation very difficult. To take into account all these constraints, we propose to partition the mediator ontology into blocks in order to limit *lcs* computation within the block that contains the anchors concepts to be connected. To realize this, a number of methods proposed for ontology partitioning may be used [12, 13]. Their general objective is to improve the effectiveness of automatic alignment methods by reducing the number of concepts the alignment tool has to deal with.

Formally, let $\mathcal{O}$ be an ontology and $\mathcal{C}$ the set of all concepts in $\mathcal{O}$. A partitioning $\mathcal{G}$ of $\mathcal{O}$ breaks $\mathcal{C}$ into a set of blocks $\{\mathcal{B}_1, ..., \mathcal{B}_n\}$, which satisfies the following conditions:

- *(i)* $\forall \mathcal{B}_i, \mathcal{B}_j / i, j \in \{1, ..., n\}$, if $i \neq j$ then $\mathcal{B}_i \cap \mathcal{B}_j = \oslash$; and
- *(ii)* $\mathcal{B}_1 \cup \mathcal{B}_2 \cup ... \cup \mathcal{B}_n = \mathcal{C}$.

In our application context, we use the partitioning method integrated into the ontology matching system Falcon-AO [14, 13]. This method is convenient for our context because it deals well with subsumption relationships and allows efficient partitioning of large-size ontologies. Indeed, using the clustering ROCK algorithm [10] and by introducing the notion of *weighted links* mainly based on a structural similarity measure between concepts, Falcon-AO allows to partition an ontology into blocks by classifying in each block *the most semantically-related concepts*.

Our algorithm uses this closest semantic concepts notion in order to limit *lcs* computation within only one block. A set of blocks, denoted $\mathcal{B}$ is generated by applying this algorithm to $\mathcal{MO}$ and each one is saved to be treated latter as a sub-ontology of $\mathcal{MO}$. Weighted links are kept for each block and this process is done only once. The algorithm that we present below uses these blocks to conciliate every agreement $\mathcal{A}_i$ in the global ontology $\mathcal{GO} = \langle \{\mathcal{A}_i\}, \mathcal{MO}' \rangle$, by incrementally computing the hierarchy $\mathcal{H}'$ of $\mathcal{OM}'$.

Algorithm 1 Conciliation

Input: $\mathcal{A}_k = \langle \mathcal{LO}'_k, \mathcal{M}_k \rangle$, $\mathcal{GO} = \langle \{\mathcal{A}_i\}_{i \neq k}, \mathcal{MO}' \rangle$, $\mathcal{B}$
Output: The new $\mathcal{GO}$
begin
  foreach $(m = (c_l, r, c_m)$ *in* $\mathcal{R}_{\mathcal{M}_k})$ **do**
    if $(\mathcal{MO}' \neq \oslash)$ *and* $(c_m \notin \mathcal{C}_{m'})$ **then**
      Identify the block $\mathcal{B}_x \in \mathcal{B}$ that contains c_m
      $\mathcal{C}_{\mathcal{B}_{xm'}} \longleftarrow \mathcal{C}_{m'} \cap \mathcal{C}_{\mathcal{B}_x}$ /*concepts of $\mathcal{B}_x$ already inserted in $\mathcal{MO}'$*/
      if $(\mathcal{C}_{\mathcal{B}_{xm'}} \neq \oslash)$ **then**
        $c_{cl} \longleftarrow closest_{\mathcal{H}'/\mathcal{H}}(c_m)$ /* $\mathcal{H}$ is the hierarchy of $\mathcal{B}_x$*/
        $c_{lcs} \longleftarrow lcs_{\mathcal{H}}(c_m, c_{cl})$ /* the lcs is found in $\mathcal{B}_x$*/
        if $(c_{lcs} = c_{cl})$ **then**
          $\mathcal{MO}' \leftarrow \mathcal{MO}' \cup \{c_m \sqsubseteq c_{cl}\}$
        else if $(c_{lcs} = c_{cm})$ **then**
          $\mathcal{MO}' \leftarrow \mathcal{MO}' \cup \{c_{cl} \sqsubseteq c_m\}$
          if $(\exists c \in \mathcal{MO}' \ / \ c = lcs_{\mathcal{H}'}(c_{cl}, c))$ **then**
            $\mathcal{MO}' \leftarrow \mathcal{MO}' \cup \{c_m \sqsubseteq c\}$
          end
        else
          $\mathcal{MO}' \leftarrow \mathcal{MO}' \cup \{c_m \sqsubseteq c_{lcs}, c_{cl} \sqsubseteq c_{lcs}\}$
          if $(\exists c \in \mathcal{MO}' \ / \ c = lcs_{\mathcal{H}'}(c_{cl}, c))$ **then**
            $\mathcal{MO}' \leftarrow \mathcal{MO}' \cup \{c_{lcs} \sqsubseteq c\}$
          end
        end
      else
        $\mathcal{MO}' \leftarrow \mathcal{MO}' \cup \{c_m \sqsubseteq \top\}$ /* $\top$ is the universal concept*/
      end
    else
      $\mathcal{MO}' \leftarrow \mathcal{MO}' \cup \{c_m \sqsubseteq \top\}$
    end
  end
end

To illustrate our algorithm, we consider the two agreements $\mathcal{A}_1 = \langle \mathcal{OL}'_1, \mathcal{M}_1 \rangle$ and $\mathcal{A}_2 = \langle \mathcal{OL}'_2, \mathcal{M}_2 \rangle$ such that:
– $\mathcal{M}_1 = \langle (FrideRice \sqsubseteq mo : Rice), (Onion \equiv mo : Onion) \rangle$; and
– $\mathcal{M}_2 = \langle (Sorgho \equiv mo : Sorgho), (Tomate \equiv mo : Tomato) \rangle$
Results obtained by conciliating $\mathcal{A}_1$ and $\mathcal{A}_2$ are as follows:

1- **Conciliate $\mathcal{A}_1$ in $\mathcal{GO}$**
Input: $\mathcal{A}_1$, $GO = \langle \{\}, \mathcal{MO}' = \{\} \rangle$

iteration 1 – $m_{1.1} = (FrideRice \sqsubseteq mo : Rice)$
$\mathcal{MO}' = \{Rice \sqsubseteq \top\}$
itération 2 – $m_{1.2} = (Onion \equiv mo : Onion)$
$B_x = Block\_23079$
$c_{cl} = Rice$; $c_{lcs} = PlanProducts$
$\mathcal{MO}' = \{Rice \sqsubseteq PlanProducts, Onion \sqsubseteq PlanProducts\}$

2- **Conciliate $\mathcal{A}_1$ and $\mathcal{A}_2$ in $\mathcal{GO}$**
Input: $\mathcal{A}_2$, $\mathcal{GO} = \langle \{\mathcal{A}_1\}, \mathcal{MO}' = \{Rice \sqsubseteq PlanProducts, Onion \sqsubseteq PlanProducts\}$

itération 1 – $m_{2.1} = (Sorgho \equiv mo : Sorgho)$
$B_x = Block\_23079$
$c_{cl} = Rice$; $c_{lcs} = Cereals$
$\mathcal{MO}' = \{Rice \sqsubseteq Cereals, Sorgho \sqsubseteq Cereals, Cereals \sqsubseteq PlanProducts, Onion \sqsubseteq PlanProducts\}$
itération 2 – $m_{2.2} = (Tomate \equiv mo : Tomato)$
$B_x = Block\_23079$
$c_{cl} = Onion$; $c_{lcs} = Vegetables$
$\mathcal{MO}' = \{Rice \sqsubseteq Cereals, Sorgho \sqsubseteq Cereals, Cereals \sqsubseteq PlanProducts, \sqsubseteq PlanProducts, Tomato \sqsubseteq Vegetables, Vegetables \sqsubseteq PlanProducts\}$

Figure 5 illustrates the global ontology $\mathcal{GO}$ resulting from the conciliation of $\mathcal{A}_1$ and $\mathcal{A}_2$. We have distinguished the hierarchy $\mathcal{MO}'$, composed of all anchor concepts in $\mathcal{M} = \mathcal{M}_1 \cup \mathcal{M}_2$, linked to each other by their *lcs* found in $\mathcal{MO}$.

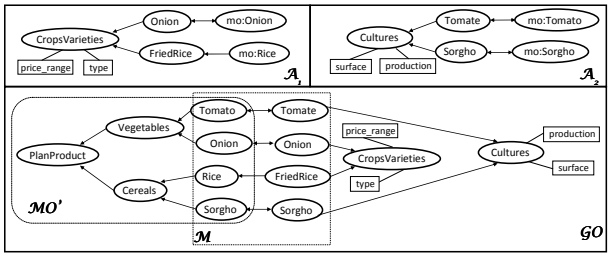


Figure 5: An excerpt of the global ontology that conciliates the two agreements $\mathcal{A}_1$ and $\mathcal{A}_2$

4. A CASE STUDY

We have implemented our solution, i.e., a global ontology construction method based on a domain reference ontology, in the context of the SIC-Senegal project. This project aims to enable several partners (agricultural agencies) to share their agricultural data. Each partner usually collects and stores its data in different formats (spreadsheets, relational databases, etc.). Previous works in this project [24] have already allowed the provision of local ontologies that semantically describe partner's data. We address the last phase of the integration process in this paper, i.e., the construction of the intended global ontology to be shared between all the partners. It represents the shared understanding of the application domain and provides a structured vocabulary for querying all the partners data.

In our experiments, we have considered the local ontologies of three partners' (P_1, P_1, P_3) and the domain reference ontology AGROVOC as a mediator one:
– The partners P_1 and P_2 treat mainly information about agricultural-crops varieties and the cultivated surfaces ; P_3 deals with prices of agricultural products.
– The local ontologies $\mathcal{LO}_1(P_1)$, $\mathcal{LO}_2(P_2)$ and $\mathcal{LO}_3(P_3)$ have respectively 323, 387 and 224 concepts.
– AGROVOC is an ontology proposed by the FAO (*Food and Agriculture Organization*) for agriculture, forestry, fisheries, food and related domains (e.g. environment). It contains up to 28 439 concepts.

After a brief explanation of our global ontology construction in this context, especially agreement and conciliation, we will present some evaluation results of our method.

Agreement. The first step of the agreement phase is the lexical anchoring process. We performed our experiments with different terminological-similarity-measures (Jaron-Winkler, Levenshtein, etc.)[4]. We have then compared the mapping

[4]We have done our experiments with 8 similarity-measures implemented in the *SimMetrics* API : `http://staffwww.dcs.shef.ac.uk/people/S.Chapman/simmetrics`

results of the lexical anchoring process with a manual anchoring process performed by an expert of the agricultural domain. We note that the *string metrics* measure proposed in [27] is the one that gave the best results. We have then performed the last step of the agreement phase, i.e., the semantic anchoring process, according to the methods described in Section 3.1.2. The result being three agreements $\mathcal{A}_1$, $\mathcal{A}_2$ and $\mathcal{A}_3$, respectively for the partners P_1, P_2 and P_3.

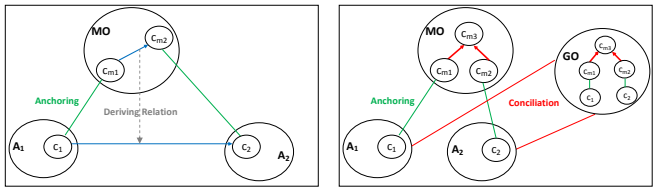
Conciliation. In this phase we have built the global ontology according to the method described in Section 3.2. An excerpt is given in Figure 5.

Semantic expressiveness of the resulted global ontology. To evaluate the semantic expressiveness[5] of the constructed global ontology w.r.t. the considered application domain, we have compared our approach to classical ontology-matching techniques [25], in particular those sharing our context and using a domain reference ontology as a mediation support [1, 2, 22]. These methods don't build a global ontology in the integration process, they are only based on ontologies alignment. They have however, as the ours, the objective of establishing links between different local ontologies for an integration purpose. They start by an anchoring phase as described in Section 3.1, our approach follows then with the conciliation process (please refer to Section 3.2) while classical ontology-matching techniques are based on derivation process [23]. Derivation is just the deduction of proximity relations (isClose, narrower-than, broader-than, is-equiv, etc.) between two anchoring concepts through the structural relations that may exist between their corresponding anchors in the domain reference ontology. Figure 6 briefly sketches the conciliation and derivation processes while Figure 7 summarizes the results obtained by applying them in our case study (blue line for derivation and red line for conciliation).

By analyzing Figures 7(a) and 7(b), we can notice that the number of discovered relations linking concepts in different local ontologies (partners) is more important when using conciliation (cf. Figure 6). In addition, the expert analysis of our results showed that the semantics of that relations is more explicit w.r.t. the application domain (Agricultural Domain) than those discovered using derivation. This can be justified by the fact that relationships discovered with derivation are mainly proximity relationships. This kind of relations have a weak semantics w.r.t. the considered application domain. On the contrary, relations established by conciliation have a well explicit semantics w.r.t. the considered application domain. As an example, the relation discovered between the concepts *Tomato* and *Onion* through the concept *Vegetable* in Figure 5, which means that both the concepts *Tomato* and *Onion* are vegetables (sub-concepts of *Vegetable*).

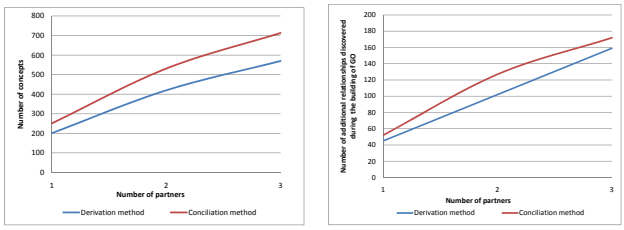
We can thus conclude that our integration-approach offers, by using a global ontology for the mediation process, a more interesting semantic expressiveness compared to classical approaches based on ontology-matching techniques.

[5]Semantic expressiveness is considered here as the number of relations discovered between the concepts of the local ontologies to be integrated and their relevance to the application domain.



(a) Classical ontology-matching techniques – derivation process (b) Our approach – conciliation process

Figure 6: A general illustration of derivation and conciliation processes



(a) The number of concepts in the integration process (b) The number of additional discovered relations in the integration process

Figure 7: Global ontology building result plots

5. RELATED WORKS

In data integration process, when designing an integration system $I =< G, S, M >$, where G is the expected global schema, S the set of sources' schemas, and M the set of mappings between G, one starts by building G and S before having the choice between several approaches for the construction of M: Local As View (LAV), Global As View (GAV), Global-Local As View (GLAV) or Both As View (BAV). Theses last ones have largely been studied in the literature [4, 8, 9, 21].

The automation in building G (for automation purpose, G is in most cases an ontology) is however not much studied in the literature. On the one hand, because building an ontology is considered as a difficult task in itself (see for example [26]), and, on the other hand, ontologies-integration task is also known to be difficult. This last task is generally ensured by computing correspondences, or mappings, between ontologies (please see [25] for a survey).

In this article we have tackled the problem of data integration in the context of ontologies, i.e., ontology-based data integration (please see [28] for a survey). Our main contribution consists in the introduction of an automatic solution for a global ontology construction problem. Our proposal is based on a domain reference ontology as a background knowledge. One the one hand, links between local ontologies are obtained from the taxonomical relationships of the reference ontology and, on the other hand, mappings between the global ontology and local ones are obtained, by syntactic matching, from the names of the local-ontologies concepts and those of the reference-ontology-concepts. For that reason, our algorithm depends on the performance of ontology-matching techniques (cf. Section 3).

The use of reference ontologies for data integration has been investigated in the literature (see for instance [2, 1, 22]). It was shown that the reference ontology can significantly improve the performance of the matching process. Our contribution in this context is to show that the reference ontology also allows to enrich the semantics of links discovered in the matching process.

6. CONCLUSION

Our proposition in this article brings a solution to the problem of automatic construction of an appropriate global ontology. We have tackled this issue using a background-knowledge (i.e., a domain-reference ontology) as a mediation support. This global ontology can offer interesting properties, especially an appropriate conceptualization and easy resource-adding and querying processes.

We are now working on the complexity evaluation of our algorithm as well as the proof of its correctness and soundness. We aim also to turn our proposition into a robust software for ontology-based data integration.

7. REFERENCES

[1] Z. Aleksovski, M. C. A. Klein, W. ten Kate, and F. van Harmelen. Matching Unstructured Vocabularies Using a Background Ontology. In *EKAW*, pages 182–197, 2006.

[2] Z. Aleksovski, W. ten Kate, and F. van Harmelen. Exploiting the Structure of Background Knowledge Used in Ontology Matching. In *Ontology Matching*, 2006.

[3] F. Baader, B. Sertkaya, and A.-Y. Turhan. Computing the least common subsumer w.r.t. a background terminology. *J. Applied Logic*, 5(3):392–420, 2007.

[4] D. Beneventano, S. Bergamaschi, F. Guerra, and M. Vincini. The momis approach to information integration. In *ICEIS (1)*, pages 194–198, 2001.

[5] F. Cerbah. Learning highly structured semantic repositories from relational databases:. In *ESWC*, pages 777–781, 2008.

[6] N. Choi, I.-Y. Song, and H. Han. A survey on ontology mapping. *SIGMOD Record*, 35(3):34–41, 2006.

[7] I. F. Cruz, H. Xiao, and F. Hsu. Peer-to-peer semantic integration of XML and RDF data sources. In *AP2PC*, pages 108–119, 2004.

[8] I. Fundulaki, B. Amann, C. Beeri, M. Scholl, and A.-M. Vercoustre. Stʏx: Connecting the xml web to the world of semantics. In *EDBT*, pages 759–761, 2002.

[9] H. Garcia-Molina, Y. Papakonstantinou, D. Quass, A. Rajaraman, Y. Sagiv, J. D. Ullman, V. Vassalos, and J. Widom. The tsimmis approach to mediation: Data models and languages. *J. Intell. Inf. Syst.*, 8(2):117–132, 1997.

[10] S. Guha, R. Rastogi, and K. Shim. ROCK: A Robust Clustering Algorithm for Categorical Attributes. *International Conference on Data Engineering*, pages 512–521, 1999.

[11] A. Y. Halevy. Answering queries using views: A survey. *VLDB J.*, 10(4):270–294, 2001.

[12] F. Hamdi, B. Safar, H. Zargayouna, and C. Reynaud. Partitionnement d'ontologies pour le passage à l'échelle des techniques d'alignement. In *EGC*, pages 409–420, 2009.

[13] W. Hu, Y. Qu, and G. Cheng. Matching large ontologies: A divide-and-conquer approach. *Data Knowl. Eng.*, 67(1):140–160, 2008.

[14] W. Hu, Y. Zhao, and Y. Qu. Partition-Based Block Matching of Large Class Hierarchies. In *ASWC*, pages 72–83, 2006.

[15] Y. Kalfoglou and M. Schorlemmer. Ontology Mapping: The State of the Art. *The Knowledge Engineering Review*, 18:2003, 2003.

[16] M. Lenzerini. Data integration: A theoretical perspective. In *PODS*, pages 233–246, 2002.

[17] B. McBride. Rdf vocabulary description language 1.0: Rdf schema, 2004.

[18] D. L. McGuinness and F. van Harmelen. Owl web ontology language overview. Technical Report REC-owl-features-20040210, W3C, 2004.

[19] N. T. Nguyen. A method for ontology conflict resolution and integration on relation level. *Cybernetics and Systems*, 38(8):781–797, 2007.

[20] A. Poggi, D. Lembo, D. Calvanese, G. D. Giacomo, M. Lenzerini, and R. Rosati. Linking Data to Ontologies. *J. Data Semantics*, 10:133–173, 2008.

[21] M.-C. Rousset and C. Reynaud. Picsel and xyleme: Two illustrative information integration agents. In *AgentLink*, pages 50–78, 2003.

[22] M. Sabou, M. d'Aquin, and E. Motta. Using the Semantic Web as Background Knowledge for Ontology Mapping. In *Ontology Matching*, 2006.

[23] M. Sabou, M. dAquin, and E. Motta. Scarlet: Semantic relation discovery by harvesting online ontologies. In *Proceedings of the 5th European Semantic Web Conference*, Tenerife, Spain, 2008.

[24] O. Sall, M. Lo, F. Gandon, C. Niang, and I. Diop. Using XML data integration and ontology reuse to share agricultural data. *IJMSO*, 4(1/2):93–105, 2009.

[25] P. Shvaiko and J. Euzenat. Ten Challenges For Ontology Matching. In *Proceedings of The 7th International Conference on Ontologies, DataBases, and Applications of Semantics (ODBASE)*, 2008.

[26] G. Steve, A. Gangemi, and D. M. Pisanelli. Integrating medical terminologies with onions methodology. In *Information Modelling and Knowledge Bases VIII (IOS*. Press, 1997.

[27] G. Stoilos, G. B. Stamou, and S. D. Kollias. A String Metric for Ontology Alignment. In *International Semantic Web Conference*, pages 624–637, 2005.

[28] H. Wache, T. Vï£¡gele, U. Visser, H. Stuckenschmidt, G. Schuster, H. Neumann, and S. Hï£¡bner. Ontology-Based Integration of Information - A Survey of Existing Approaches. In *IJCAI'01 Workshop on Ontologies and Informations Sharing*, pages 108–117, 2001.

[29] Z. Wu and M. S. Palmer. Verb Semantics and Lexical Selection. In *ACL*, pages 133–138, 1994.