Query Answering in Distributed Description Logics

Faisal Alkhateeb and Antoine Zimmermann
{faisal.alkhateeb,antoine.zimmermann}@inrialpes.fr

INRIA Rhône-Alpes/LIG, France

Abstract. This paper describes the notion of query answering in a distributed knowledge based system, and gives methods for computing these answers in certain cases. More precisely, given a distributed system (DS) of ontologies and ontology mappings (or bridge rules) written in Distributed Description Logics (DDL), distributed answers are defined for queries written in terms of one particular ontology. These answers may contain individuals from different ABoxes. To compute these answers, the paper provides an algorithm that reduce the problem of distributed query answering to local query answering. This algorithm is proved correct but not complete in the general case.

1 Introduction

The emergence of the Semantic Web has focused the attention on developing systems in which knowledge can be shared in a distributed environment. Besides, query answering in expressive knowledge base is a difficult task. Therefore, when knowledge description is separated into different knowledge bases, query answering becomes an even more tedious problem. The present paper investigates a new approach to the problem, where local knowledge bases (ontologies) may represent heterogeneous domains, but are related with directional mappings that express how one can interpret foreign knowledge from a given peer's point of view. Queries are posed in terms of one local ontology (the target), and answers are given in the context of the target ontology, while taking advantage of the overall distributed knowledge. For such an approach, Distributed Description Logics (DDL [1]) is an appropriate knowledge representation language, but it has currently no supports for queries with variables. So, this paper defines the notion of distributed answers to a query. Then, to evaluate queries, we propose to reduce the problem of distributed query answering to a local query answering problem, assuming that there already exist local query evaluation algorithms (*e.g.*, [2, 3]).

Several peer-to-peer (P2P) data management systems have been proposed recently, which are divided into two main categories: centralized systems (*e.g.*, [4]) and decentralized systems (*e.g.*, [5–7]). [5] presents a relational P2P data management system, and the mappings between relational peer schemas are inclusion and equivalence statements of conjunctive queries. The Piazza system [8] is a P2P data management system that relies on a tree based data model: data in XML and XQuery-based mapping language for mediating between peers. PEPSINT [4] supports interoperation of both XML and RDF data sources, using a hybrid architecture with a super peer containing the global ontology. EDUTELLA [6] provides an RDF-based metadata infrastructure for P2P networks. [9, 7] describe the SomeWhere semantic P2P data management system that promotes a small vision of the Semantic Web based on simple ontologies distributed at a large scale, and logical mappings between ontologies make possible the creation of a web of people.

Most of the existing systems are assumed to work with rather homogeneous data, and they might prove useless if different ontologies are developed for a different context of application. This is the particularity of our work: DDL can handle distributed knowledge where each ontology may provide different context, and mappings between domains are handled through the use of the so-called bridge-rules. Another peculiarity of our approach is that, even though the query is posed in terms of one particular local ontology, a single answer may contain individuals from several knowledge bases.

The paper is organized as follows: we present the syntax and the semantics of DDL in Section 2. Section 3 presents the syntax and the semantics of the query language that we consider. In Section 4, we prove a theorem that provides a guideline for a query evaluation procedure over DDL. The concluding remarks and further work are presented in Section 5.

2 Distributed Description Logics

Our work is based on DDL [1]. DDL serves to describe a distributed knowledge base (DKB) composed of several local KBs (written in standard DL) and of "bridge rules" that serve to connect terms from different local KBs.

2.1 Ontology Language:DL

Syntax. Our local KBs, that we will refer to as *ontologies*, are written in Description Logics (DL). The basic elements in DL are concepts, roles and individuals. Concepts (class of individuals) and roles (relations between individuals) are either primitive (named concepts or roles) or complex (recursively defined with constructors and other concepts or roles). Individuals can only be described by a name. Constructors are given in Table 1.

Ontologies are composed of axioms asserting truth about a knowledge domain. We distinguish terminological axioms, *i.e.*, statements about concepts and roles, and assertional axioms, *i.e.*, facts about individuals. All possible axioms are summarized in Table 1.

Definition 1 (Ontology) *An* ontology *(or local knowledge base) O is a pair $\langle T, A\rangle$ where T is a set of terminological axioms called TBox, and A is a set of assertional axioms called ABox.*

Given an ontology O, we denote by $\mathrm{Sig}(O)$ the set of all terms (primitive concept/role and individual names) appearing in the axioms of O and we call it the *signature* of O. Moreover, given a set of terms Σ, we denote by $\mathcal{A}(\Sigma)$ the set of all possible axioms inductively built out of terms of Σ and constructors from Table 1.

Construct name	Syntax	Semantics
individual	a	$a^I \in \Delta^I$
atomic concept	A	$A^I \subseteq \Delta^I$
universal concept	$\top$	$\top^I = \Delta^I$
empty concept	$\bot$	$\bot^I = \emptyset$
conjunction	$C \sqcap D$	$(C \sqcap D)^I = C^I \cap D^I$
disjunction	$C \sqcup D$	$(C \sqcup D)^I = C^I \cup D^I$
negation	$\neg C$	$\Delta^I \setminus C^I$
exists restriction	$\exists R.C$	$\{x \| \exists y.\langle x, y\rangle \in R^I \wedge y \in C^I\}$
value restriction	$\forall R.C$	$\{x \| \forall y.\langle x, y\rangle \in R^I \Rightarrow y \in C^I\}$
number	$\leq nR$	$\{x \| \sharp\{y.\langle x, y\rangle \in R^I\} \leq n\}$
restrictions	$\geq nR$	$\{x \| \sharp\{y.\langle x, y\rangle \in R^I\} \geq n\}$
nominals	$\{a_1, \ldots, a_n\}$	$\{a_1^I, \ldots, a_n^I\}$
atomic role	R	$R^I \subseteq \Delta^I \times \Delta^I$
role conjunction	$R \sqcap S$	$(R \sqcap S)^I = R^I \cap S^I$
role disjunction	$R \sqcup S$	$(R \sqcup S)^I = R^I \cup S^I$
role complement	$\neg R$	$(\Delta^I \times \Delta^I) \setminus R^I$
transitive closure	R^+	transitive closure of R^I
inverse role	R^-	$\{\langle y, x\rangle \| \langle x, y\rangle \in R^I\}$
role composition	$R \circ S$	$\{\langle x, z\rangle \| \exists y.\langle x, y\rangle \in R^I \wedge \langle y, z\rangle \in S^I\}$
TBox axioms	Syntax	Interpretation constraints
subsumption	$C \sqsubseteq D$	$C^I \subseteq D^I$
role inclusion	$R \sqsubseteq S$	$R^I \subseteq S^I$
role transitivity	$\mathrm{Trans}(R)$	$R^I = (R^+)^I$
ABox axioms	Syntax	Interpretation constraints
class membership	$C(a)$	$a^I \in C^I$
role membership	$R(a_1, a_2)$	$\langle a_1^I, a_2^I\rangle \in R^I$
idendity	$a_1 = a_2$	$a_1^I = a_2^I$

Table 1. Syntax and semantics of DL constructors

Semantics. The semantics of DL define the notion of interpretation, and specify when an interpretation satisfies an axiom or an ontology.

Definition 2 (Interpretation of a set of terms) *Given a set of terms Σ, an* interpretation *I of Σ is a pair $\langle \Delta^I, \cdot^I\rangle$ where Δ^I is a non-empty set called the domain of I, and $\cdot^I$ is an interpretation function that maps individuals of Σ to elements of Δ^I, concepts of Σ to subsets of Δ^I and roles of Σ to subsets of*

$\Delta^I \times \Delta^I$. *The interpretation function is extended to complex concepts or roles by applying interpretation rules of Table 1 recursively.*

Satisfiability allows one to identify interpretations that are consistent with the statements expressed in axioms.

Definition 3 (Axiom satisfiability) *Let Σ be a set of terms, and $\phi \in \mathcal{A}(\Sigma)$ an axiom in terms of Σ. An interpretation I of Σ satisfies ϕ iff it fulfills the constraint associated with ϕ in Table 1. In this case, we denote it $I \models \phi$.*

Given an ontology O, an interpretation I of $\mathrm{Sig}(O)$ satisfies O iff it satisfies all the axioms of O. In this case, we write $I \models O$ and call I a *model* of O. Moreover, we denote by $\mathrm{Mod}(O)$ the set of all models of O. Additionally, an ontology is *satisfiable* if it has at least one model.

The next section explains how knowledge from different ontologies can be related, and how distributed systems are interpreted and satisfied.

2.2 Distributed Systems

Basically, a distributed (knowledge-based) system (DS for short) is a structure composed of ontologies and ontology mappings interconnecting them. Since there are now several ontologies, it is convenient to identify the provenance of terms. To do so, we add a prefix to terms or axioms denoting the ontology whence they come. Additionally, we will consistently use K to denote a set of indexes.

Syntax. In DDL, ontology mappings are expressed by *bridge rules.*

Definition 4 (Bridge rule) *Let O_i and O_j be two ontologies. A* bridge rule *from O_i to O_j ($i \neq j$), is an expression of one of the following two forms:*

- $i\!:\!C \xrightarrow{\sqsubseteq} j\!:\!D$ *an into-bridge rule;*
- $i\!:\!C \xrightarrow{\sqsupseteq} j\!:\!D$ *an onto-bridge rule;*
- $i\!:\!x \mapsto j\!:\!y$ *a (partial) individual correspondence;*
- $i\!:\!x \overset{=}{\mapsto} j\!:\!\{y_1,\dots,y_n\}$ *a complete individual correspondence;*

where $i\!:\!C$ and $j\!:\!D$ are either two concepts or two roles of O_i and O_j respectively, $i\!:\!x$ is an individual of O_i and $j\!:\!y, j\!:\!y_1,\dots,j\!:\!y_n$ are individuals of O_j.

Informally, a distributed system is a set of ontologies interconnected with ontology mappings.

Definition 5 (Distributed System) *A* Distributed System *(DS) is a pair $S = \langle (O_i)_{i\in K}, (B_{ij})_{i\neq j}\rangle$ where for all $i,j \in K$, O_i is a DL ontology as defined in the previous section, and B_{ij} is a (possibly empty) set of bridge rules between O_i and O_j.*[1]

The notion of distributed system can capture P2P knowledge-based systems, the Semantic Web, or multi-agent systems.

[1] When there is no ambiguity, we will write $\langle \mathbf{O}, \mathbf{B}\rangle$ to denote $\langle (O_i)_{i\in K}, (B_{ij})_{i\neq j}\rangle$.

Semantics. In a distributed system in DDL, each ontology is interpreted according to the local DL semantics. Since interpretation domains of different ontologies can be heterogeneous, a distributed interpretation also describe how two different domains are interrelated. This done via domain relation, as shown in the following definition.

Definition 6 (Distributed interpretation) *Let $S = \langle \mathbf{O}, \mathbf{B} \rangle$ be a DS. A* distributed interpretation *$\mathcal{I} = \langle \mathbf{I}, \mathbf{r} \rangle$ of S assigns to each ontology O_i an interpretation $I_i = \langle \Delta^{I_i}, \cdot^{I_i} \rangle$ of* $\mathrm{Sig}(O_i)$ *and for all pairs $i \neq j$, a* domain relation *$r_{ij} \subseteq \Delta^{I_i} \times \Delta^{I_j}$.*[2]

Informally, a domain relation r_{ij} serves to express what objects from Δ^{I_i} represent from the O_j's point of view. A distributed interpretation satisfies a DS when (1) its local interpretations locally satisfy their respective ontology, and (2) satisfies constraints imposed by bridge rules as defined in the following definition.

Definition 7 (Distributed satisfiability) *Let $\mathcal{I} = \langle \mathbf{I}, \mathbf{r} \rangle$ be a distributed interpretation. $\mathcal{I}$ satisfies a bridge rule b_{ij} (written $\mathcal{I} \models_d b_{ij}$) when the following constraints hold:*

- *if b_{ij} is $i\!:\!C \xrightarrow{\sqsubseteq} j\!:\!D$ then $\mathcal{I} \models_d b_{ij} \Leftrightarrow r_{ij}(C^{I_i}) \subseteq D^{I_j}$;*
- *if b_{ij} is $i\!:\!C \xrightarrow{\sqsupseteq} j\!:\!D$ then $\mathcal{I} \models_d b_{ij} \Leftrightarrow r_{ij}(C^{I_i}) \supseteq D^{I_j}$;*
- *if b_{ij} is $i\!:\!x \mapsto j\!:\!y$ then $\mathcal{I} \models_d b_{ij} \Leftrightarrow y^{I_j} \in r_{ij}(x^{I_i})$;*
- *if b_{ij} is $i\!:\!x \overset{=}{\mapsto} j\!:\!\{y_1, \ldots, y_n\}$ then $\mathcal{I} \models_d b_{ij} \Leftrightarrow r_{ij}(x^{I_i}) = \{y_1^{I_j}, \ldots, y_n^{I_j}\}$.*

A distributed interpretation satisfying a DS is called a model of the DS.

Definition 8 (Model of a DS) *A* model *of a DS $\langle \mathbf{O}, \mathbf{B} \rangle$ is a distributed interpretation $\mathcal{I} = \langle \mathbf{I}, \mathbf{r} \rangle$ such that for all $i \in K$, $I_i \models O_i$ and for all $i, j \in K, b_{ij} \in B_{ij}$, $\mathcal{I} \models_d b_{ij}$. This is denoted by $\mathcal{I} \models_d S$.*

An axiom (resp. a bridge rule) α (resp. β) is a semantic consequence of S if all models of S satisfy α (resp. β).

Next section defines the notion of targeted query over a DS, and specify the problem we address in this paper: answering queries targeted on a specific ontology within a DS.

3 Query answering in DDL

We are interested, in this section, in defining answers to queries posed in terms of one ontology in DDL. So the query is specified in one specific context, and answers are given *w.r.t.* this context, even though the overall DS is used to determine these answers. We first define the syntax of such queries, and then define the set of possible answers to a given query over a DDL knowledge base.

[2] For all $d \in \Delta^{I_i}$, we use $r_{ij}(d)$ to denote $\{d' \in \Delta^{I_j} | \langle d, d' \rangle \in r_{ij}\}$, for any $D \subseteq \Delta^{I_i}$, we use $r_{ij}(D)$ to denote $\bigcup_{d \in D} r_{ij}(d)$, and for any $R \subseteq \Delta^{I_i} \times \Delta^{I_j}$ to denote $\bigcup_{\langle d,e \rangle \in R} r_{ij}(d) \times r_{ij}(e)$.

3.1 Syntax

Informally, a query as we consider here is simply a set of axioms that can be constructed from the terms of one specific local ontology as well as variables. So, we will consider the following formal definition of a query.

Definition 9 (Query) *A* query *Q is a tuple of the form $\langle X, Y, \Sigma, F\rangle$, where X and Y are sets of variables, Σ is a set of terms, X, Y and Σ are pairwise disjoint and $F \subseteq \mathcal{A}(\Sigma \cup X \cup Y)$ is a set of DL axioms where variables from X and Y are used as terms.*

In the above definition, variables of X are called *distinguished variables* and variables of Y are existentially quantified and called *non-distinguished variables.* A very common notation for queries is $q(\overline{x}) \leftarrow body(\overline{x}, \overline{y})$, where $\overline{x}$ represents the distinguished variables, $\overline{y}$ the non-distinguished ones, and *body* represents the axioms. The set of terms upon which the axioms are defined is made implicit. We will use this more convenient notation in our practical examples.

Example 1. The query $q(?x) \leftarrow \texttt{Student}(?y) \wedge \texttt{hastStudent}(?x, ?y)$ corresponds to the tuple $\langle X, Y, \Sigma, F\rangle$, s.t. $X = \{?x\}$, $Y = \{?y\}$, $\Sigma = \{\texttt{Students}, \texttt{hasStudent}\}$, $F = \{\texttt{Student}(?y), \texttt{hasStudent}(?x, ?y)\}$.

Definition 10 (Targeted query on an ontology) *Given an ontology O, a* targeted query *Q on O is a tuple $\langle X, Y, \mathrm{Sig}(O), F\rangle$.*

In this paper, we are only interested in answering targeted queries, which means we only want to get answers that are related to the knowledge domain of the target ontology, yet in accordance to all the knowledge described in every ontologies and bridge rules.

3.2 Semantics

As in distributed system semantics, there are two levels of query satisfiability: local and global. Local satisfiability corresponds to the usual query satisfiability over an ontology language. Global satisfiability necessitates a more elaborate definition since it has to deal with the presence of bridge-rules.

Definition 11 (Assignment) *An* assignment *is a mapping $\alpha : X \to \Sigma$ from a set of variables X to a set of terms Σ.*

Given a query Q and an assignment α, $\alpha(Q)$ denotes a the axioms of Q in which all variables x are replaced by $\alpha(x)$. Consequently, $\alpha(Q)$ is a set of axioms with no distinguished variables.

Definition 12 (Interpretation extended to variables) *Let Σ be a set of terms, Y be a set of variables disjoint from Σ and I be an interpretation of Σ. Then, an extension I' of I to a set of variables X is an interpretation of $\Sigma \cup X$ such that $\forall x \in \Sigma\ x^{I'} = x^I$.*

This extension of interpretations serves to define satisfaction of axioms with variables, as shown below.

Definition 13 (Satisfied query) *Let $Q = \langle X, Y, \Sigma, F\rangle$ be a query, $\alpha : X \to \Sigma$ an assignment and I be an interpretation of Σ. Then, I satisfies $\alpha(Q)$ if there exist an extension I' of I to Y such that $I' \models \alpha(Q)$. In this case, we simply write $I \models \alpha(Q)$.*

This definition makes sense since $\alpha(Q)$ is a set of axioms built out of $\Sigma \cup Y$, and I' interprets it.

Definition 14 (Local answer to a query) *Let $Q = \langle X, Y, \mathrm{Sig}(O), F\rangle$ be a query targeted on ontology O. An answer to Q over O is an assignment $\alpha : X \to \mathrm{Sig}(O)$ such that $\forall I \in \mathrm{Mod}(O)$, $I \models \alpha(Q)$. We denote by $\mathrm{Ans}(Q, O)$ the set of all local answers to a query Q targeted on O.*

Now we must define a distributed answer to a targeted query. The difficulty lies in the fact that there might be answers with terms from different ontologies. In this case, the resulting assignment α may be such that $\alpha(Q)$ has terms from different ontologies. Yet, none of our currently defined interpretations can interpret axioms where terms from different ontologies appear. So, we will provide a new definition for that matter.

The idea of the following definition is to extend a local interpretation to an interpretation of all the terms that appear in a DS. However, since we only want to answer the query in the target ontology's context, we want this extended interpretation to stay within the same domain of interpretation, hence, to be local. To this extent, we first define the extension of an ontology signature *w.r.t.* a DS.

Definition 15 (Extended signature) *Let $S = \langle \mathbf{O}, \mathbf{B}\rangle$ be a DS, with O_t a particular ontology of S. The extended signature $\mathrm{eSig}_S(O_t)$ of O_t w.r.t. S is the smallest set such that:*

- $\mathrm{Sig}(O_t) \subseteq e\mathrm{Sig}_S(O_t)$;
- *for all $e \in \mathrm{Sig}(O_i)$, there exists a distinct term $e^{i \to t} \in e\mathrm{Sig}_S(O_t)$.*

Now our goal is to be able to interpret the extended signature in the local domain of the target ontology. Such an interpretation is defined by taking advantage of a given distributed interpretation.

Definition 16 (Combined interpretation) *Let $\mathcal{I} = \langle \mathbf{I}, \mathbf{r}\rangle$ be an interpretation of a DS, O_t be an ontology of the DS. The* combined interpretation *of I targeted on O_t is an interpretation $\mathcal{I}_t^c$ of the extended vocabulary $\mathrm{eSig}_S(O_t)$ defined by:*

- $\mathcal{I}_t^c = \langle \Delta_t, \cdot^{\mathcal{I}_t^c}\rangle$;
- $\forall e \in \mathrm{Sig}(O_t)$, $e^{\mathcal{I}_t^c} = e^{I_t}$; *and*
- $\forall i \neq t, \forall e \in \mathrm{Sig}(O_i)$, $(e^{i \to t})^{\mathcal{I}_t^c} = r_{it}(e^{I_i})$.

It must be remarked that when e is an individual of ontology $O_i \neq O_t$, then $r_{it}(e^{I_i})$ is a set of elements of the domain of interpretation. So, foreign individuals are interpreted as subset of Δ_t, not as elements. However, our goal is to obtain a standard DL interpretation of foreign terms. We envisage two possibilities to overcome this problem: either we restrict the combined interpretation to foreign concepts and roles and ignore foreign individuals; or we add an explicit constraint on individual terms in order to make their interpretation a singleton. More precisely, this second choice imposes that a bridge rule $i\!:\!x \overset{=}{\mapsto} t\!:\!\{x^{i\to t}\}$ is added to the DS for each foreign individual term $x \in \mathrm{Sig}(O_i)$. While this does not fully solve the problem, it at least ensures that if $\mathcal{I}$ is a model of the DS, then $\mathcal{I}_t^c$ is a well defined DL interpretation. This is actually all we need to have a valid definition of query answer.

Our approach works for these two options, and we will present the differences that are implied by the two approaches.

Definition 17 (Distributed answer to a query) *Let $S = \langle \mathbf{O}, \mathbf{B} \rangle$ be a distributed system. Let $Q_t = \langle X, Y, \mathrm{Sig}(O_t), F \rangle$ be a query targeted on O_t in S. A* distributed answer *to Q_t is an assignment $\alpha : X \to \mathrm{eSig}_S(O_t)$ such that for all models $\mathcal{I}$ of S, $\mathcal{I}_t^c \models \alpha(Q_t)$. We denote by $\mathrm{dAns}(Q_t, O_t, S)$ the set of all distributed answers to a query Q_t targeted on O_t in S.*

Example 1 *To illustrate our approach, we consider the following DDL system that contains only two ontologies $\boldsymbol{O}_1$ and $\boldsymbol{O}_2$, and the bridge-rules $\boldsymbol{B}_{21}$. In this system `Teacher` and `Student` are concepts, and the roles are `teachesTo` and `hasStudent`:*

$\boldsymbol{O}_1$	$1\!:\!\exists \mathtt{teachesTo} \sqsubseteq \mathtt{Teacher}$ $1\!:\!\mathtt{funct}(\mathtt{teachesTo})$
	$1\!:\!\mathtt{teachesTo}(\mathtt{John}, \mathtt{Bob})$
$\boldsymbol{B}_{21}$	$2\!:\!\mathtt{hasStudent} \overset{\sqsubseteq}{\to} 1\!:\!\mathtt{teachesTo}^-$ $2\!:\!\mathtt{John\text{-}Doe} \overset{=}{\mapsto} 1\!:\!\mathtt{John}$
$\boldsymbol{O}_2$	$2\!:\!\mathtt{Student} \sqsubseteq \exists \mathtt{hasStudent}^-$
	$2\!:\!\mathtt{hasStudent}(\mathtt{John\text{-}Doe}, \mathtt{Robert})$ $2\!:\!\mathtt{Student}(\mathtt{Larry})$

Consider now the following query targeted on $\boldsymbol{O}_1$:

$$Q_1(?x, ?y) \leftarrow \mathtt{Teacher}(?x), \mathtt{teachesTo}(?x, ?y)$$

the answer to the above query is:

$$\langle ?x, ?y \rangle \in \{\langle \mathtt{John}, \mathtt{Bob} \rangle, \langle \mathtt{John\text{-}Doe}, \mathtt{Robert} \rangle, \langle \mathtt{John\text{-}Doe}, \mathtt{Bob} \rangle, \langle \mathtt{John}, \mathtt{Robert} \rangle\}$$

Note that our approach is capable of assigning individuals in the same answer from different domains (e.g., $\langle \mathtt{John}, \mathtt{Robert} \rangle$), which is not provided by previous approaches.
Consider now the following query with the same body as the previous one:

$$Q_2(?y) \leftarrow \mathtt{Teacher}(?x), \mathtt{teachesTo}(?x, ?y)$$

the answer to the above query is:

$$?y \in \{\texttt{Bob}, \texttt{Robert}, \texttt{Larry}\}$$

Although both of the above two queries have the same body, `Larry` *appears in the answer of the second query while it does not appear in the answer of the first one. This is because we know that there is someone who teaches to* `Larry` *but we do not know whom, so there is no assignment to the variable $?x$ in the first query.*

4 Evaluating a targeted query over DDL

Though our definitions for local and distributed answers are given for general queries, we are interested in answering conjunctive queries over DDL knowledge base. Conjunctive queries only allow variables in replacement of individuals, and only have axioms of the form $C(\tau)$ and $R(\tau_1, \tau_2)$ where C is a concept, R is a role and τ, τ_1, τ_2 are either individual terms or variables.

In order to present our method for computing answers, we first show an important theorem that includes properties satisfied by the combined interpretation.

Theorem 1 *Let $\mathcal{I} = \langle \mathbf{I}, \mathbf{r} \rangle$ be a distributed interpretation of a DS, and O_t a target ontology. The following properties holds:*

- *if $\mathcal{I} \models_d i : B \xrightarrow{\sqsubseteq} t : C$ then $\mathcal{I}_t^c \models t : B^{i \to t} \sqsubseteq C$;*
- *if $\mathcal{I} \models_d i : S \xrightarrow{\sqsubseteq} t : R$ then $\mathcal{I}_t^c \models t : S^{i \to t} \sqsubseteq R$;*
- *if $\mathcal{I} \models_d i : D \xrightarrow{\sqsupseteq} t : C$ then $\mathcal{I}_t^c \models t : C \sqsubseteq D^{i \to t}$;*
- *if $\mathcal{I} \models_d i : T \xrightarrow{\sqsupseteq} t : R$ then $\mathcal{I}_t^c \models t : R \sqsubseteq T^{i \to t}$;*
- *if $I_i \models i : B \sqsubseteq D$ then $\mathcal{I}_t^c \models t : B^{i \to t} \sqsubseteq D^{i \to t}$.*

Moreover, if we consider the second choice mentioned after Definition 16, the following holds too:

- *if $I_i \models i : B(a)$ then $\mathcal{I}_t^c \models t : B^{i \to t}(a^{i \to t})$;*
- *if $I_i \models i : S(a, b)$ then $\mathcal{I}_t^c \models t : S^{i \to t}(a^{i \to t}, b^{i \to t})$;*
- *if $I_i \models i : a = b$ then $\mathcal{I}_t^c \models t : a^{i \to t} = b^{i \to t}$;*

where C is any concept, R is any role and B, D, S, T are defined according to the following grammar:

$$\begin{aligned} B &::= A \,|\, \exists S.\top \,|\, B \sqcup B \,|\, [\{a_1, \ldots, a_n\}] \\ D &::= A \,|\, D \sqcap D \,|\, D \sqcup D \,|\, [\{a_1, \ldots, a_n\}] \\ S &::= P \,|\, S^- \,|\, S \sqcup S \\ T &::= P \,|\, T^- \,|\, T \sqcup T \,|\, T \sqcap T \end{aligned}$$

and A is a primitive concept, P a primitive role and $a_1, \ldots, a_n$ are individuals[3]*.*

[3] Individuals are allowed only if the second choice mentioned after Definition 16 is taken.

The proof for this theorem is not very difficult but long and may be hard to follow because of the many variables and notations involved.[4]

When using DL constructs in axioms or bridge rules other than the ones used in Theorem 1, there is no guarantee that a corresponding axiom is satisfied by the combined interpretation $\mathcal{I}_t^c$. Counter examples for all constructs not in the list above are found in the online appendix.

4.1 Query Evaluation

In order to compute the distributed answers to a query targeted on an ontology, we build a new ontology which extends the targeted one with terms from foreign ontologies and axioms deduced from foreign axioms and bridge rules, according to Theorem 1. This ontology corresponds to a kind of deductive closure of the target ontology with respect to a DS. This new ontology is thus called the "targeted distributed closure". More precisely, the target ontology signature is extended to $e\text{Sig}_S(O_t)$ and all axioms satisfied by all combined interpretations are added to it. Building the closure ontology O'_t serves to transform the distributed query answering problem into a local query answering problem. Indeed, the following property holds:

Algorithm 1: Closure(O_t, S)

Data: A distributed system $S = \langle \mathbf{O}, \mathbf{B} \rangle$, and an ontology $O_t \in S$.
Result: a new ontology O'_t.
begin
  $O'_t ::= O_t$;
  for *each bridge rule* b_{ti} *such that* $S \models_d b_{ti}$ **do**
    if b_{ti} *is of one of the forms given in Theorem 1* **then**
      add the corresponding axiom to O'_t
  for *each axiom* ϕ_i *in* O_i **do**
    if ϕ_i *is of one of the forms given in Theorem 1* **then**
      add the corresponding axiom to O'_t
  return O'_t
end

$$\text{Ans}(Q_t, O_t) \subseteq \text{Ans}(Q_t, \text{Closure}(O_t, S)) \subseteq \text{dAns}(Q_t, O_t, S)$$

So this gives a correct algorithm for computing distributed answers, on the condition that we have (1) a correct and complete deduction procedure for DDL, computing all semantic consequences of a DS in finite time; (2) local query answering facilities. The first point is of course very optimistic, but it can give a criteria on the type of DL and/or ontologies that permits computable answers. Moreover, it is not needed to have the complete closure in order to have the

[4] The reader can refer to the online appendix at the following url: `http://www.inrialpes.fr/exmo/people/zimmer/NTMS2007proof.pdf`.

above property. Any ontology extending O_t with axioms obtained by applying the rules in Theorem 1 satisfies this property. So, it is possible to extend the ontology progressively, while computing new answers step by step. This method, which corresponds to a lazy evaluation, permits to give answers at anytime during the process. Finally, in the favorable case when the closure is computable and the query has only distinguished variables, the set of local answers to the query over the closure is most likely equal to the set of distributed answers over the DS.

Example 2 *According to Algorithm 1, the closure of ontology* $\boldsymbol{O}_1$*, which includes all possible deductions that can be made by the bridge-rules and foreign axioms, is:*

$1{:}\exists \mathtt{teachesTo} \sqsubseteq \mathtt{Teacher}$
$1{:}\mathtt{funct}(\mathtt{teachesTo})$
$1{:}\mathtt{Student}^{2\to 1} \sqsubseteq \exists(\mathtt{hasStudent}^{2\to 1})^-$
$1{:}\mathtt{hasStudent}^{2\to 1} \sqsubseteq \mathtt{teachesTo}$
$1{:}\mathtt{teachesTo}(\mathtt{John}, \mathtt{Bob})$
$1{:}\mathtt{Student}^{2\to 1}(\mathtt{Larry}^{2\to 1})$
$1{:}\mathtt{hasStudent}^{2\to 1}(\mathtt{John\text{-}Doe}^{2\to 1}, \mathtt{Robert}^{2\to 1})$
$1{:}\mathtt{John\text{-}Doe}^{2\to 1} = \mathtt{John}$

5 Conclusion and future work

We have investigated the query answering problem over DDL, and showed that, in some specific cases, the problem can be reduced to answering query over local DL ontology by constructing the closure ontology. The closure ontology extends a local ontology with all possible axioms that can be deduced from other ontologies in the system using bridge-rules between different domains, which allow to deduce additional information from other ontologies, and thus find answers that were not given by the sole local knowledge base.

Our approach guarantees correctness as soon as there exists a sound algorithm for querying DL ontologies. However, it does not guarantee completeness in the general case. Nonetheless, we conjecture and strongly believe that with ontologies in DL-Lite and minimal constraints on bridge rules, the algorithm we propose would prove to be complete, and it is part of our future investigations to do so.

Future work will also regard four main directions: optimization, implementation of this approach, test of the performance of our approach in terms of scalability (*i.e.*, the number of peers in the system) and study of the algorithmic complexity. Comparison with existing approach in terms of expressivity and completeness of results is also envisaged. Especially, a possible optimization would consist in taking into account the query in the construction of the closure, such that it would not be needed to deduce all possible axioms, and in particular, reduce the set of needed axioms to a finite one.

References

1. Borgida, A., Serafini, L.: Distributed Description Logics: Assimilating information from peer sources. Journal of Data Semantics **1** (2003) 153–184
2. Calvanese, D., De Giacomo, G., Lembo, D., Lenzerini, M., Rosati, R.: DL-Lite: Tractable description logics for ontologies. In: Proceedings of the 20th National Conference on Artificial Intelligence (AAAI 2005). (2005) 602–607
3. Calvanese, D., De Giacomo, G., Lembo, D., Lenzerini, M., Rosati, R.: Data complexity of query answering in description logics. In: Proceedings of the Tenth International Conference on Principles of Knowledge Representation and Reasoning (KR 2006). (2006) 260–270
4. Cruz, I.F., Xiao, H., Hsu, F.: Peer-to-Peer Semantic Integration of XML and RDF Data Sources. In: Third International Workshop on Agents and Peer-to-Peer Computing (AP2PC 2004). (2004)
5. Halevy, A., Ives, Z., Suciu, D., Tatarinov, I.: Schema mediation in peer data management systems. In: Proceedings of ICDE. (2003)
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: Proceedings of the 11th international conference on World Wide Web (WWW2002). (2002)
7. Goasdoue, F., Rousset, M.C.: Querying Distributed Data through Distributed Ontologies: A Simple but Scalable Approach. IEEE Intelligent Systems **18** (2003) 60–65
8. Halevy, A.Y., Ives, Z.G., Mork, P., Tatarinov, I.: Piazza: data management infrastructure for semantic web applications. In: Proceedings of the 12th international conference on World Wide Web (WWW2003), New York, NY, USA, ACM Press (2003) 556–567
9. Adjiman, P., Chatalic, P., Goasdou, F., Rousset, M.C., Simon, L.: SomeWhere in the Semantic Web . In: International Workshop on Principles and Practice of Semantic Web Reasoning. (2005)