

## View-Based Query Answering over Description Logic Ontologies

Diego Calvanese<sup>1</sup>, Giuseppe De Giacomo<sup>2</sup>, Maurizio Lenzerini<sup>2</sup>, Riccardo Rosati<sup>2</sup>

<sup>1</sup> Faculty of Computer Science  
Free University of Bozen-Bolzano  
Piazza Domenicani 3  
I-39100 Bolzano, Italy  
calvanese@inf.unibz.it

<sup>2</sup> Dip. di Informatica e Sistemistica  
Sapienza Università di Roma  
Via Ariosto 25  
I-00185 Roma, Italy  
lastname@dis.uniroma1.it

### Abstract

View-based query answering is the problem of answering a query based only on the answers precomputed for a set of views. While this problem has been widely investigated in databases, it is largely unexplored in the context of Description Logic ontologies. Differently from traditional databases, Description Logics may express several forms of incomplete information, and this poses challenging problems in characterizing the semantics of views. In this paper, we first present a general framework for view-based query answering, where we address the above semantical problems by defining a spectrum of notions of view-based query answering over ontologies, all based on the idea that the precomputed answers to views are the certain answers to the corresponding queries. We also relate such notions to relevant issues in ontology management, in particular ontology access authorization. Then, we provide decidability results, algorithms, and data complexity characterizations for view-based query answering in several Description Logics, ranging from the *DL-Lite* family to very expressive Description Logics.

### Introduction

View-based query answering is the problem of answering a query under the assumption that the only accessible extensional information consists of the precomputed answers to a set of queries, called views. Such a problem is considered fundamental in several scenarios of information management, including data integration, query optimization, maintenance of physical independence, data warehousing, and privacy-aware access to information (see (Halevy 2001) for a survey).

In particular, in the logical approach to privacy-aware access to data, each user (or, class of users) is associated with a set of views, called authorization views, which specify the information that the user is allowed to access (Zhang & Mendelzon 2005; Rizvi *et al.* 2004; Stouppa & Studer 2007). View-based query answering in this setting captures the requirement that only information deriving from such views can be revealed to the user.

A large number of results is reported on view-based query answering in the recent database literature, both for the case of relational databases (see, for instance, (Ullman 2000;

Halevy 2001)), and for the case of semistructured data (Calvanese *et al.* 2001; 2005). On the other hand, the problem is still largely unexplored in the context of Description Logics (DLs). Two exceptions are (Beeri, Levy, & Rousset 1997; Calvanese, De Giacomo, & Lenzerini 2000), which, however, formalize view definitions and view extensions essentially as additional assertions of a sophisticated form, rather than as knowledge about a set of queries whose precomputed answers are stored as view extensions.

In this paper, we present a new study on view-based query answering in DL ontologies. The idea at the basis of our work is that, differently from traditional databases, DLs may express several forms of incomplete information, and this should be taken into account in characterizing the semantics of the views. To address this issue, our approach considers the precomputed answers to views as the certain answers to the corresponding queries, and therefore differs from the previous work mentioned above. Our contributions can be summarized as follows.

- We present a general formal framework for view-based query answering in DL ontologies. Users pose queries to a system, whose knowledge is represented by an ontology expressed in a given DL. The system associates to each user (or, class of users) a set of views, whose extensions are computed as certain answers to the ontology. The system answers user queries coherently with the ontology, though hiding information not implied by the views. This idea is formalized based on the fundamental notion of *solution*. Given an ontology  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  expressed in a DL  $\mathcal{L}$ , a set of views  $V$  with extensions  $E$ , a solution for  $(\mathcal{T}, V, E)$  is a set of interpretations such that computing the certain answers of the views  $V$  over such interpretations yields exactly  $E$ . We show that different definitions can be provided for the semantics of view-based query answering, each one capturing additional properties for the notion of solution. In the model-centered semantics, we simply insist that a solution is a set of models of the TBox  $\mathcal{T}$ . In the language-centered semantics, we additionally require that the set of models constituting a solution is captured by an ontology expressed in  $\mathcal{L}$ . Finally, in the TBox-centered semantics, we require that such a set can be expressed in terms of an  $\mathcal{L}$  ABox paired to  $\mathcal{T}$ .
- We relate the framework to the problem of privacy-aware

access to ontologies, by illustrating how view-based query answering is able to conceal from the user the information that are not logical consequences of the associated authorization views. In particular, a fact  $q(\vec{t})$  that logically follows from a DL ontology  $\mathcal{K}$  is concealed from the user when there is a solution for  $(\mathcal{T}, V, E)$  that falsifies  $q(\vec{t})$  and cannot be distinguished from the models of  $\mathcal{K}$  by using  $V$  and  $\mathcal{T}$ . The various semantics defined in the framework correspond to imposing different notions of solution, and this in turn implies that the different semantics disclose different amounts of information to the user.

- We illustrate several decidability results, algorithms, and data complexity characterizations for view-based query answering, under different semantics, and for different DLs, ranging from tractable ones (the *DL-Lite* family (Calvanese *et al.* 2007; Poggi *et al.* 2008) and the  $\mathcal{EL}$  family (Baader, Brandt, & Lutz 2005; Krisnadhi & Lutz 2007; Rosati 2007)), to more expressive ones (*ALC*, *ALCQI* (Baader *et al.* 2003), *SHIQ* (Horrocks, Sattler, & Tobies 2000; Horrocks, Patel-Schneider, & van Harmelen 2003; Glimm *et al.* 2007)).

The paper is organized as follows. In the next section, we present basic notions for DL ontologies. Then, we illustrate the framework and its relationship with privacy-aware access to ontologies. The next two sections present general results on view-based query answering, and specific results for various DLs. The paper ends with a discussion on both the results presented here, and future directions for continuing our work.

## Preliminaries

In this section, we define some preliminary notions used in the rest of the paper. Let  $\mathcal{S}$  be a signature of unary predicates, binary predicates, and individual constants.

As usual, a Description Logic (DL) ontology  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  is a pair formed by a TBox  $\mathcal{T}$  and an ABox  $\mathcal{A}$  over  $\mathcal{S}$ , where  $\mathcal{T}$  contains universal assertions, and  $\mathcal{A}$  contains assertions about individuals.

Different DLs allow for different constructs in forming TBox and ABox assertions. In this paper we consider several well-known DLs, including

- expressive DLs, namely *ALC*, *ALCQI*, and *SHIQ*;
- DLs from the  $\mathcal{EL}$  family, namely,  $\mathcal{EL}$  and  $\mathcal{ELH}$ ;
- DLs from the *DL-Lite* family, namely, *DL-Lite<sub>F</sub>*, *DL-Lite<sub>R</sub>*, and *DL-Lite<sub>A</sub>*.

Below, for the sake of brevity, we give some details on the latter DLs only. We refer to (Baader *et al.* 2003; Horrocks, Sattler, & Tobies 2000; Baader, Brandt, & Lutz 2005; Krisnadhi & Lutz 2007) for the other ones.

The *DL-Lite* family (Calvanese *et al.* 2007; 2006) is a family of tractable DLs particularly suited for dealing with ontologies with very large ABoxes, which can be managed through relational database technology. One of the most expressive DLs in the family is *DL-Lite<sub>A</sub>*, which allows for representing both objects and values. Here, for simplicity, we concentrate on its fragment concerning objects only.

However, all the results presented in this paper hold for the full-fledged version of *DL-Lite<sub>A</sub>* as well.

In *DL-Lite<sub>A</sub>*, the syntax of concept and role expressions is as follows ( $A$  denotes a unary predicate name, i.e., concept,  $P$  denotes a binary predicate name, i.e., role, and  $a, b$  denote constants):

$$\begin{array}{l} B \longrightarrow A \mid \exists R \\ C \longrightarrow B \mid \neg B \\ R \longrightarrow P \mid P^- \\ Q \longrightarrow R \mid \neg R \end{array}$$

while TBox assertions and ABox assertions are formed according to the following syntax:

$$\begin{array}{l} \kappa_{tbox} \longrightarrow B \sqsubseteq C \mid R \sqsubseteq Q \mid (\text{funct } Q) \\ \kappa_{abox} \longrightarrow A(a) \mid P(a, b) \end{array}$$

with the proviso that *roles occurring in functionality assertions cannot be specialized*, i.e., they cannot appear in the right-hand side of inclusion assertions.

In other words, a *DL-Lite<sub>A</sub>* TBox consists of a finite set of:

- inclusion assertions between basic concepts (which are either atomic or unqualified existential restrictions) and general concepts (which include also negation);
- inclusion assertions between basic roles (which are either atomic or inverse roles) and general roles (which again include also negation);
- functional assertions on basic roles, such that roles occurring in functionality assertions cannot be specialized.

A *DL-Lite<sub>A</sub>* ABox consists of a finite set of concept and role membership assertions.

In this paper, we also consider two other DLs of the *DL-Lite*-family, namely, *DL-Lite<sub>R</sub>*, obtained from *DL-Lite<sub>A</sub>* by dropping functional assertions ((*funct*  $Q$ )), and *DL-Lite<sub>F</sub>*, obtained from *DL-Lite<sub>A</sub>* by dropping inclusion assertions on roles ( $Q \sqsubseteq R$ ).

Below, we call *atomic ABox* an ABox containing only assertions of the form  $A(a)$  and  $P(a, b)$  where  $A$  is an atomic concept and  $P$  is an atomic role. The DLs *DL-Lite<sub>R</sub>*, *DL-Lite<sub>F</sub>*, and *DL-Lite<sub>A</sub>* allow only for atomic ABoxes, but this is not the case for every DL.

As for the semantics, as usual in DLs, the logics in the *DL-Lite* family are based on the standard notion of first-order interpretation, consisting of an interpretation domain and an interpretation function. In addition, as in several DLs, the *Unique Name Assumption* (UNA) is enforced: i.e., different constant names denote different objects of the interpretation domain.

An interpretation over  $\mathcal{S}$ , or simply interpretation, is a *model* of an ontology  $\mathcal{K}$  if it makes true (or satisfies) all TBox and ABox assertions in  $\mathcal{K}$ . We denote by  $MOD(\mathcal{K})$  the set of models of  $\mathcal{K}$ . An ontology  $\mathcal{K}$  is *satisfiable* if it has at least one model, i.e., if  $MOD(\mathcal{K}) \neq \emptyset$ .

A *conjunctive query* (CQ)  $q$  is an expression of the form

$$q(\vec{x}) \leftarrow \text{conj}(\vec{x}, \vec{y})$$

where  $\vec{x}$  are the so-called *distinguished variables*,  $\vec{y}$  are the *non-distinguished variables*, which are implicitly existentially quantified, and each *conj*( $\vec{x}, \vec{y}$ ) is a conjunction of

atoms of the form  $A(z)$ ,  $P(z, z')$ , where  $A$  is a unary predicate name,  $P$  is a binary predicate name, and  $z, z'$  are constants in  $\mathcal{S}$ , or variables in  $\vec{x}$  or  $\vec{y}$ . When  $\vec{x}$  is the tuple  $\langle \rangle$  of arity 0, then  $q$  is called a Boolean query. In this paper we use the so-called Datalog notation for CQs (Abiteboul, Hull, & Vianu 1995), i.e., we write conjunctions simply as sequences.

Given an interpretation  $\mathcal{I}$ ,  $q^{\mathcal{I}}$  is the set of tuples of domain elements that, when assigned to the distinguished variables  $\vec{x}$  of  $q$ , make the formula  $\exists \vec{y}. \text{conj}(\vec{x}, \vec{y})$  true (Abiteboul, Hull, & Vianu 1995). We denote  $\vec{t} \in q^{\mathcal{I}}$  also as  $\mathcal{I} \models q(\vec{t})$ .

Given a set  $\mathcal{W}$  of interpretations and a CQ  $q$ , the set of *certain answers*  $\text{cert}(q, \mathcal{W})$  to  $q$  over  $\mathcal{W}$  is defined as follows:

$$\text{cert}(q, \mathcal{W}) = \{\vec{t} \mid \vec{t} \text{ is a tuple of constants in } \mathcal{S} \\ \text{s.t. } \mathcal{I} \models q(\vec{t}) \text{ for every } \mathcal{I} \in \mathcal{W}\}$$

Given an ontology  $\mathcal{K}$ , the set of *certain answers to  $q$  over  $\mathcal{K}$*  is the set  $\text{cert}(q, \text{MOD}(\mathcal{K}))$ , also written simply as  $\text{cert}(q, \mathcal{K})$ . We denote  $\vec{t} \in \text{cert}(q, \mathcal{K})$  also as  $\mathcal{K} \models q(\vec{t})$ .

## Framework

As we said in the introduction, in view-based query answering, the answer to a query posed to  $\mathcal{K}$  is computed solely on the basis of the knowledge on a *finite sequence*  $V = \langle v_1, \dots, v_n \rangle$  of views over the ontology. In this paper we consider conjunctive views, and therefore the definition  $v_i$  of each view is a CQ of the form

$$v_i(\vec{x}_i) \leftarrow \text{conj}_i(\vec{x}_i, \vec{y}_i)$$

where each  $v_i$  is not a symbol of the signature  $\mathcal{S}$ , and  $v_i \neq v_j$  for  $i \neq j$ . In the following, we use  $v_i$  both for the symbol denoting the view, and for the CQ constituting its definition.

Given  $V = \langle v_1, \dots, v_n \rangle$ , we call  $V$ -*extension* any sequence  $E = \langle e_1, \dots, e_n \rangle$ , where each  $e_i$  is a finite set of tuples of the same arity as  $v_i$ .

Since an ontology is characterized by a set of models, we are interested in those extensions of the views  $V$  that correspond to the certain answers of the queries  $\langle v_1, \dots, v_n \rangle$  with respect to a set of interpretations. Formally, the *certain extension* of  $V$  with respect to a set of interpretations  $\mathcal{W}$ , denoted  $\text{cert}(V, \mathcal{W})$ , is the sequence  $\langle e_1, \dots, e_n \rangle$  where  $e_i = \text{cert}(v_i, \mathcal{W})$ . A prominent role is played by  $\text{cert}(V, \text{MOD}(\mathcal{K}))$ , i.e., the certain extension of  $V$  with respect to the set of models of the ontology  $\mathcal{K}$ , simply written  $\text{cert}(V, \mathcal{K})$  and called the  $V$ -*extension for  $\mathcal{K}$* .

We base our framework on the following characterizing elements.

- Users pose CQs to a system, whose knowledge is represented by an ontology  $K = \langle \mathcal{T}, \mathcal{A} \rangle$  expressed in a given DL  $\mathcal{L}$  over a signature  $\mathcal{S}$ .
- The system associates to each user (or, class of users) some views  $V$ , which are CQs over  $\mathcal{S}$ , whose extension  $E$  is computed as  $\text{cert}(V, \mathcal{K})$ .
- The system answers user queries *faithfully*, i.e., coherently with  $\mathcal{K}$ , though hiding information not implied by the views  $V$ . This idea is captured by grounding the

semantics of view-based query answering on the notion of *solution* for  $(\mathcal{T}, V, E)$ . Intuitively, a solution for  $(\mathcal{T}, V, E)$  is a set of interpretations  $\mathcal{W}$  over  $\mathcal{S}$  which can be “confused” with  $\mathcal{K}$  in the following sense:

- $\mathcal{W} \subseteq \text{MOD}(\mathcal{T})$ , and
- $\text{cert}(V, \mathcal{W}) = E$ .

Now, the set of answers that the system provides to a user query  $q$  is the set of tuples  $\vec{t}$  such that  $\vec{t} \in \text{cert}(q, \mathcal{W})$  for every solution  $\mathcal{W}$  for  $(\mathcal{T}, V, E)$ .

Observe that  $\mathcal{K}$ , or, more precisely,  $\text{MOD}(\mathcal{K})$ , is obviously a solution for  $(\mathcal{T}, V, E)$ , but many other ontologies, or, more precisely, set of models, are solutions as well. Observe also that if  $\mathcal{W}_i$  and  $\mathcal{W}_j$  are both solutions for  $(\mathcal{T}, V, E)$ , then they are indistinguishable by means of  $V$ , because  $\text{cert}(V, \mathcal{W}_i) = \text{cert}(V, \mathcal{W}_j) = E$ . In other words, a tuple  $\vec{t}$  such that  $q(\vec{t})$  is logically implied by  $\mathcal{K}$  is hidden to the user if there is a solution  $\mathcal{W}$  for  $(\mathcal{T}, V, E)$  where  $\vec{t}$  is not a certain answer to  $q$ , i.e.,  $\vec{t} \notin \text{cert}(V, \mathcal{W})$ . In this sense  $\mathcal{W}$  is an evidence for concealing  $q(\vec{t})$  from the user.

In this work, we refer to three notions of solutions. The first one is exactly the one described above. The other two semantics capture increasing requirements on the level of faithfulness of the system with respect to the ontology  $\mathcal{K}$ .

**Model-centered semantics.** A *model-centered solution*, or simply  $M$ -solution, for  $(\mathcal{T}, V, E)$  is a solution  $\mathcal{W}$  for  $(\mathcal{T}, V, E)$  as specified above. In other words, given a user query  $q$ , the evidence for concealing  $q(\vec{t})$  from the user can be simply any subset of the models of the TBox  $\mathcal{T}$  of  $\mathcal{K}$ .

**Language-centered semantics.** This semantics imposes the further condition that a set of interpretations  $\mathcal{W}$  can be considered a solution if it is expressible in terms of an ontology in the same DL  $\mathcal{L}$  as  $\mathcal{K}$ . More precisely, a *language-centered solution*, or simply  $L$ -solution, for  $(\mathcal{T}, V, E)$  is an  $M$ -solution  $\mathcal{W}$  for  $(\mathcal{T}, V, E)$  such that there exists an ontology  $\langle \mathcal{T}', \mathcal{A}' \rangle$  expressed in  $\mathcal{L}$  with  $\mathcal{W} = \text{MOD}(\langle \mathcal{T}', \mathcal{A}' \rangle)$ . So, any evidence for concealing  $q(\vec{t})$  from the user must be expressible in  $\mathcal{L}$ .

**TBox-centered semantics.** Finally, this semantics characterizes a set of interpretations as a solution if it can be captured by the original TBox  $\mathcal{T}$  and an ABox expressed in  $\mathcal{L}$ : a *TBox-centered solution*, or simply  $T$ -solution, for  $(\mathcal{T}, V, E)$  is an  $M$ -solution  $\mathcal{W}$  for  $(\mathcal{T}, V, E)$  such that there exists an ABox  $\mathcal{A}'$  expressed in  $\mathcal{L}$  with  $\mathcal{W} = \text{MOD}(\langle \mathcal{T}, \mathcal{A}' \rangle)$ . In other words, any evidence for concealing  $q(\vec{t})$  from the user must be expressible in terms of an ABox in  $\mathcal{L}$  paired to  $\mathcal{T}$ .

Observe that, as an immediate consequence of the above definitions, every  $T$ -solution for  $(\mathcal{T}, V, E)$  is also an  $L$ -solution for  $(\mathcal{T}, V, E)$ , and every  $L$ -solution for  $(\mathcal{T}, V, E)$  is also an  $M$ -solution for  $(\mathcal{T}, V, E)$ .

We are now ready to formally introduce the notion of view-based query answering. In the following definition,  $\sigma$

stands for either  $M$  (for model-centered),  $L$  (for language-centered), or  $T$  (for TBox-centered), thus referring to one of the three semantics defined above.

**View-based answer.** Let  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  be an ontology expressed in a DL  $\mathcal{L}$ ,  $q$  a CQ,  $V$  a set of views, and  $E = \text{cert}(V, \mathcal{K})$  the  $V$ -extension for  $\mathcal{K}$ . The set of *view-based answers* to  $q$  with respect to  $(\mathcal{T}, V, E)$  under the  $\sigma$ -centered semantics, denoted by  $\text{vba}_\sigma(q, \mathcal{T}, V, E)$ , is the set of tuples  $\vec{t}$  such that  $\vec{t} \in \text{cert}(q, \mathcal{W})$  for every  $\sigma$ -solution  $\mathcal{W}$  for  $(\mathcal{T}, V, E)$ .

In this paper, we study the decision problem associated to computing view-based answers: given a TBox  $\mathcal{T}$ , views  $V$ , a  $V$ -extension  $E$ , a query  $q$ , and a tuple  $\vec{t}$ , check whether  $\vec{t} \in \text{vba}_\sigma(q, \mathcal{T}, V, E)$ , i.e.,  $\vec{t}$  is a view-based answer to  $q$  with respect to  $(\mathcal{T}, V, E)$  under the  $\sigma$ -centered semantics.

Since in our framework  $E$  represents the  $V$ -extensions for  $\mathcal{K}$ , we are guaranteed that at least one  $\sigma$ -solution (for any  $\sigma$ ) always exists, namely  $\text{MOD}(\mathcal{K})$ . Also, since we are interested in data complexity, we will actually measure the complexity of such decision problem only with respect to the size of the  $V$ -extension  $E$ .

We end this section by relating the framework presented here to the privacy scenario illustrated in the introduction. In such a scenario:

- The ontology  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  expressed in a given DL  $\mathcal{L}$  over a signature  $S$  represents the knowledge that the system has on the domain of interest.
- The views  $V$  associated to a user (or, class of users) are its *authorization views*, i.e., the  $V$ -extensions  $E$  for  $\mathcal{K}$  represent the knowledge that the system is authorized to disclose to the user.
- The user posing queries to the system is aware of the signature  $S$ , but is in principle unaware of all other aspects managed by the system ( $\mathcal{K}$ ,  $\mathcal{L}$ ,  $V$ , and  $E$ ).
- The answers provided by the system to a user query  $q$  are those logically implied by its authorization views, i.e., the view-based answers to  $q$ .
- The different semantics described above, ranging from the model-centered to the TBox-centered semantics, allow for increasing levels of information disclosure to the user. In particular, let  $q$  be a user query, and let  $q(\vec{t})$  be logically implied by  $\mathcal{K}$ :
  - In the model-centered semantics,  $q(\vec{t})$  is returned as an answer if there is no set of models  $\mathcal{W} \subseteq \text{MOD}(\mathcal{K})$  that is indistinguishable from  $\text{MOD}(\mathcal{K})$  itself with respect to the authorization views  $V$ , and such that  $\vec{t} \notin \text{cert}(q, \mathcal{W})$ .
  - In the language-centered semantics,  $\mathcal{W}$  must further satisfy the condition of being expressible in the DL  $\mathcal{L}$  of  $\mathcal{K}$ . Intuitively, this is the semantics to adopt in order to capture the case where the user is aware of the language  $\mathcal{L}$  used to express  $\mathcal{K}$ .
  - In the TBox-centered semantics,  $\mathcal{W}$  must be characterized by some ABox expressed in  $\mathcal{L}$  paired to  $\mathcal{T}$ . Intuitively, this semantics captures the situation where the

user is aware not only of the language  $\mathcal{L}$ , but also of the TBox  $\mathcal{T}$  of  $\mathcal{K}$ .

**Example 1** Consider a  $DL\text{-Lite}_R$  ontology  $\mathcal{K}$  with roles  $\text{Owns}$  and  $\text{Located}$ , where  $\text{Owns}(a, b)$  means that person  $a$  owns house  $b$ , and  $\text{Located}(x, y)$  means that  $x$  is located in  $y$ . Specifically, let  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  with  $\mathcal{T}$  empty, and  $\mathcal{A}$  be:

$$\text{Owns}(\text{john}, h55), \quad \text{Located}(h55, \text{london})$$

Now, suppose that the system can only disclose to a user  $u$  information about the location of houses owned by persons, and information about where houses are located. This can be formalized by associating to  $u$  the following authorization views  $V$ :

$$\begin{aligned} v_1(x, y) &\leftarrow \text{Owns}(x, z), \text{Located}(z, y) \\ v_2(x, y) &\leftarrow \text{Located}(x, y) \end{aligned}$$

The resulting  $V$ -extension  $E = \langle e_1, e_2 \rangle$  is

$$\begin{aligned} e_1 &= \{ \langle \text{john}, \text{london} \rangle \} \\ e_2 &= \{ \langle h55, \text{london} \rangle \} \end{aligned}$$

It is not hard to see that  $\text{Owns}(\text{john}, h55)$  is concealed from user  $u$  under the model-centered semantics, while it is not concealed under the other semantics. Indeed, since  $DL\text{-Lite}_R$  ABoxes only allow for atomic membership assertions, it follows that, in the TBox-centered semantics, every substitution  $t$  of the join variable  $z$  in  $v_1$  must be a constant explicitly mentioned in the ABox, thus the pair  $\langle t, \text{london} \rangle$  must also belong to the extension  $e_2$ , and hence  $t = h55$ . As for the language-centered semantics, we observe that  $DL\text{-Lite}_R$ , being a subset of  $\text{SHIQ}$ , enjoys the tree-model property (Baader *et al.* 2003), which again implies that every substitution  $t$  of the join variable  $z$  in  $v_1$  must be a constant explicitly mentioned in the ABox. Instead, such a conclusion cannot be derived in the model-centered semantics, which is language-independent.

Suppose now that the language  $\mathcal{L}$  used by the system to express  $\mathcal{K}$  is the DL obtained from  $DL\text{-Lite}_R$  by allowing for role composition in TBox assertions. In this case,  $\text{Owns}(\text{john}, h55)$  is concealed from  $u$  not only in the model-centered semantics, but also in the language-centered semantics, because the set of models of the following ontology

$$\{ \{ P \sqsubseteq \text{Owns} \circ \text{Located} \}, \{ A \cup \{ P(\text{john}, \text{london}) \} \} \}$$

is indistinguishable from  $\text{MOD}(\langle \mathcal{T}, \mathcal{A} \rangle)$  by  $V$ . ■

In the remaining sections we present several results on the problem of view-based query answering, by concentrating our attention on the model-centered and the TBox-centered semantics. The language-centered semantics will be addressed in an extended version of this paper.

## General results

We start with a preliminary definition. Given views  $V = \langle v_1, \dots, v_n \rangle$ , where each  $v_i$  is defined as  $v_i(\vec{x}_i) \leftarrow \text{conj}_i(\vec{x}_i, \vec{y}_i)$ , and  $V$ -extension  $E = \langle e_1, \dots, e_n \rangle$ , where each  $e_i$  is of the form  $\{ \vec{t}_i^1, \dots, \vec{t}_i^{m_i} \}$ , we denote by  $\alpha(V, E)$  the Boolean CQ

$$\begin{aligned} \alpha(V, E) &= \text{conj}_1(\vec{t}_1^1, \vec{y}_1^1), \dots, \text{conj}_1(\vec{t}_1^{m_1}, \vec{y}_1^{m_1}), \dots \\ &\quad \dots, \text{conj}_n(\vec{t}_n^1, \vec{y}_n^1), \dots, \text{conj}_n(\vec{t}_n^{m_n}, \vec{y}_n^{m_n}) \end{aligned}$$

**Theorem 2** *If  $\text{MOD}(\mathcal{T} \cup \alpha(V, E))$  is a  $T$ -solution for  $(\mathcal{T}, V, E)$ , then view-based query answering under the three semantics coincides for CQs.*

*Proof (sketch).* First, observe that, since  $\alpha(V, E)$  is a Boolean query, it corresponds to a first-order sentence. Now, if  $\text{MOD}(\mathcal{T} \cup \alpha(V, E))$  is a  $T$ -solution for  $(\mathcal{T}, V, E)$ , then  $\alpha(V, E)$  can be equivalently expressed in terms of an ABox. Moreover, it is immediate to see that every solution for  $(\mathcal{T}, V, E)$  under each of the three semantics logically implies  $\alpha(V, E)$ .  $\square$

We then provide a general undecidability result for view-based query answering. First, we give an auxiliary lemma.

**Lemma 3** *If an ABox  $\mathcal{A}$  is such that  $\text{MOD}(\langle \mathcal{T}, \mathcal{A} \rangle)$  is a  $T$ -solution for  $(\mathcal{T}, V, E)$ , then  $\langle \mathcal{T}, \mathcal{A} \rangle \models \alpha(V, E)$ .*

**Theorem 4** *If answering conjunctive queries in a DL  $\mathcal{L}$  is undecidable, then view-based query answering in  $\mathcal{L}$  is undecidable under all the three semantics.*

*Proof (sketch).* Under model-centered semantics, the theorem is a direct consequence of Theorem 7. For TBox-centered semantics (and in an analogous way for language-centered semantics), we can easily reduce conjunctive query answering in  $\mathcal{L}$  to view-based query answering in  $\mathcal{L}$ . Given  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ , we build a view definition  $V$  (with one-to-one views) and a  $V$ -extension  $E$  (isomorphic to  $\mathcal{A}$ ) such that  $\alpha(V, E) = \mathcal{A}$ . Thus, by Lemma 3, it follows that  $\langle \mathcal{T}, \mathcal{A} \rangle \models q(\vec{t})$  iff  $\vec{t} \in \text{vba}_T(q, \mathcal{T}, V, E)$ .  $\square$

**Results for model-centered semantics** With respect to view-based query answering under model-centered semantics, we start by stating the following properties, whose proof is straightforward.

**Proposition 5** *If  $\text{MOD}(\mathcal{T} \cup \alpha(V, E))$  is not an  $M$ -solution for  $(\mathcal{T}, V, E)$ , then there are no  $M$ -solutions for  $(\mathcal{T}, V, E)$ .*

**Proposition 6** *If  $\text{MOD}(\mathcal{T} \cup \alpha(V, E))$  is an  $M$ -solution for  $(\mathcal{T}, V, E)$ , then  $\vec{t} \in \text{vba}_M(q, \mathcal{T}, V, E)$  iff  $\vec{t} \in \text{cert}(q, \text{MOD}(\mathcal{T} \cup \alpha(V, E)))$ .*

The next theorem gives a general results on how to transfer decidability and complexity results from conjunctive query answering to view-based query answering under model-centered semantics for DLs without UNA.

**Theorem 7** *For every DL  $\mathcal{L}$  without UNA, view-based conjunctive query answering under model-centered semantics and conjunctive query answering are mutually reducible to each other in LOGSPACE.*

*Proof (sketch).* First, we define a reduction from view-based query answering in  $\mathcal{L}$  under model-centered semantics to query answering in  $\mathcal{L}$ : the crucial point is the definition of an ABox obtained by freezing  $\alpha(V, E)$  using one new constant for every variable (which is correct due to Proposition 6, and the fact that  $\mathcal{L}$  is without UNA). Vice-versa, it is easy to define a reduction from query answering in  $\mathcal{L}$  to view-based query answering under model-centered semantics, through one-to-one views  $V$  (one view for every concept and role name) and a  $V$ -extension isomorphic to the ABox.  $\square$

**Results for TBox-centered semantics** We start our analysis of view-based query answering under TBox-centered semantics by considering DLs admitting only atomic ABoxes (as defined in the preliminaries).

**Theorem 8** *If a DL  $\mathcal{L}$  only allows for atomic ABoxes, and answering CQs in  $\mathcal{L}$  is decidable, then view-based conjunctive query answering in  $\mathcal{L}$  under TBox-centered semantics is decidable.*

*Proof (sketch).* To reason about all the  $T$ -solutions, it is sufficient to try all the ABoxes that can be built by the predicates in the signature  $\mathcal{S}$  and constants occurring in  $E$  (plus a finite number of new constant names). The number of such ABoxes is finite.  $\square$

By extending the proof of the above theorem, we obtain the following property.

**Theorem 9** *If a DL  $\mathcal{L}$  only allows for atomic ABoxes, and conjunctive query answering in  $\mathcal{L}$  is in CONP (respectively, in PTIME) with respect to data complexity, then view-based conjunctive query answering in  $\mathcal{L}$  under TBox-centered semantics is in  $\Pi_2^P$  (respectively, CONP) with respect to data complexity.*

Finally, we show the following, very general, hardness result for view-based conjunctive query answering under TBox-centered semantics. A DL  $\mathcal{L}$  has the tree-model property if every satisfiable TBox has a model that has the structure of a tree, when viewing each object as a node, and each pair of nodes connected by a direct or inverse atomic role as an edge.

**Theorem 10** *If a DL  $\mathcal{L}$  has the tree-model property, then view-based conjunctive query answering in  $\mathcal{L}$  under TBox-centered semantics is CONP-hard with respect to data complexity.*

*Proof (sketch).* Reduction from 3-colorability. Let  $G = (V_G, E_G)$  be a graph. The view definitions  $V$  are the following:

$$\begin{aligned} v_1(x) &\leftarrow \text{vertexCol}(x, y), \text{vertexColAux}(x, z), \text{col}(y, z) \\ v_2(x, y) &\leftarrow \text{edge}(x, y) \\ v_3(x) &\leftarrow \text{col}(x, x) \end{aligned}$$

Intuitively,  $v_1$  is used to assign a color to  $x$ , but it does so through a query that includes a loop (which is not expressible in  $\mathcal{L}$ , by the tree-model property);  $v_2$  denotes the presence of an edge between  $x$  and  $y$ ; and  $v_3$  denotes colors. Then consider the following  $V$ -extension  $E$ , populating  $v_1$  with the nodes of  $G$ ,  $v_2$  with the edges of  $G$ , and  $v_3$  with 3 colors:

$$E = \langle V_G, E_G, \{r, g, b\} \rangle$$

and the Boolean query  $q$ , denoting two adjacent vertices (notice that the query is symmetric with respect to edges) colored with the same color:

$$q \leftarrow \text{edge}(x, y), \text{vertexCol}(x, z), \text{vertexCol}(y, z)$$

It can be shown that  $G$  is 3-colorable iff  $\langle \rangle \notin \text{vba}_T(q, \emptyset, V, E)$ . Observe that in this construction the TBox is empty.  $\square$

## Algorithms and complexity for specific DLs

In this section we first provide a general algorithm for view-based query answering under TBox-centered semantics, then we present a set of complexity results for view-based query answering (under both model-centered semantics and TBox-centered semantics) for several Description Logics, ranging from tractable DLs ( $DL-Lite_A$  and  $\mathcal{EL}$ ) to very expressive ones ( $\mathcal{SHIQ}$ ).

We start by providing some preliminary definitions.

An  $\mathcal{L}$ -UCQ is a union of conjunctive queries whose atoms are ABox assertions in  $\mathcal{L}$  involving either constants or variables. More precisely  $\mathcal{L}$ -UCQ atoms are of the following forms: (i)  $C(z)$  with  $z$  either a constant or a variable, where  $C(a)$ , with  $a$  constant, is an ABox assertion in  $\mathcal{L}$ ; (ii)  $R(z_1, z_2)$  with  $z_1, z_2$  either constants or variables, where  $R(a, b)$ , with  $a, b$  constants, is an ABox assertion in  $\mathcal{L}$ .

We now define the notion of roll-up of a CQ.

**Roll-up.** Given a DL  $\mathcal{L}$ , a CQ  $q$  and a TBox  $\mathcal{T}$ , an  $\mathcal{L}$ -UCQ  $Q$  is a *roll-up of  $q$  under  $\mathcal{T}$  in  $\mathcal{L}$*  if, for every ABox  $\mathcal{A}$  and for every tuple  $\vec{t}$ ,  $\langle \mathcal{T}, \mathcal{A} \rangle \models q(\vec{t})$  iff  $\langle \mathcal{T}, \mathcal{A} \rangle \models \text{grounding}(Q, \mathcal{T}, \mathcal{A})$ , where  $\text{grounding}(Q, \mathcal{T}, \mathcal{A})$  is the grounding of  $Q$  with respect to the constants occurring in  $\mathcal{T}$  and  $\mathcal{A}$  (notice that  $\text{grounding}(Q, \mathcal{T}, \mathcal{A})$  is a Boolean ABox in  $\mathcal{L}$ , in particular, a finite disjunction of a finite number of ABoxes in  $\mathcal{L}$ ).

Given an  $\mathcal{L}$ -UCQ  $Q$ , we denote by  $\text{freeze}(Q, \mathcal{T}, V, E)$  the expression obtained from  $Q$  replacing every variable symbol occurring in  $Q$  with a new distinct constant not already occurring in  $Q, \mathcal{T}, V$ , or  $E$ .

Notice that  $\text{freeze}(Q, \mathcal{T}, V, E)$  is a disjunction of ABoxes, in particular the disjunction of a finite number of finite ABoxes.

The following crucial property establishes a precise relationship between the roll-up of  $\alpha(V, E)$  and view-based query answering under TBox-centered semantics.

**Lemma 11** *Let  $Q^\alpha$  be a roll-up of  $\alpha(V, E)$  under  $\mathcal{T}$ . Then, for every CQ  $q$ , we have that  $\text{vba}_T(q, \mathcal{T}, V, E) = \text{cert}(q, \langle \mathcal{T}, \text{freeze}(Q^\alpha, \mathcal{T}, V, E) \rangle)$ .*

*Proof (sketch).* We focus on the case when the DL  $\mathcal{L}$  is without UNA. (If  $\mathcal{L}$  enforces UNA we can still follow the same line of the proof presented here, although some details require more care.) Suppose  $\vec{t} \in \text{cert}(q, \langle \mathcal{T}, \text{freeze}(Q^\alpha, \mathcal{T}, V, E) \rangle)$ , and suppose there exists an ABox  $\mathcal{A}'$  such that  $\text{MOD}(\langle \mathcal{T}, \mathcal{A}' \rangle)$  is a T-solution for  $(\mathcal{T}, V, E)$  and such that  $\langle \mathcal{T}, \mathcal{A}' \rangle \not\models q(\vec{t})$ . Now, by Lemma 3 we have  $\langle \mathcal{T}, \mathcal{A}' \rangle \models \alpha(V, E)$  and since  $Q^\alpha$  is a roll-up of  $\alpha(V, E)$  under  $\mathcal{T}$  in  $\mathcal{L}$ , it follows that  $\langle \mathcal{T}, \mathcal{A}' \rangle \models \text{grounding}(Q^\alpha, \mathcal{T}, \mathcal{A}')$ . Let  $h_1$  be the homomorphism from  $Q^\alpha$  to  $\text{grounding}(Q^\alpha, \mathcal{T}, \mathcal{A}')$ , let  $h_2$  be the isomorphism from  $Q^\alpha$  to  $\text{freeze}(Q^\alpha, \mathcal{T}, V, E)$ , and let  $h$  be the homomorphism  $h_2^{-1} \cdot h_1$  (i.e.,  $h$  is the homomorphism which maps  $\text{freeze}(Q^\alpha, \mathcal{T}, V, E)$  to  $\text{grounding}(Q^\alpha, \mathcal{T}, \mathcal{A}')$ ). Now let  $\mathcal{I}$  be a model of  $\langle \mathcal{T}, \mathcal{A}' \rangle$  such that  $\mathcal{I} \not\models q(\vec{t})$ . Let  $\mathcal{I}'$  be the interpretation obtained from  $\mathcal{I}$  by modifying the interpretation of constants as follows: for every constant  $a$ ,  $a^{\mathcal{I}'} = (h(a))^{\mathcal{I}}$ . It is immediate to see that: (i)  $\mathcal{I}'$  is

**Algorithm View-based-answer $_T$** ( $\vec{t}, q, \mathcal{L}, \mathcal{T}, V, E$ )

**Input:** tuple  $t$ , CQ  $q$ , DL  $\mathcal{L}$ ,  $\mathcal{L}$ -TBox  $\mathcal{T}$ , views  $V$ ,  $V$ -extension  $E$

**Output:** true if  $\vec{t} \in \text{vba}_T(q, \mathcal{T}, V, E)$ , false otherwise

**begin**

$Q := \text{rollUp}(\mathcal{L}, \mathcal{T}, \alpha(V, E));$

$Q := \text{freeze}(Q, \mathcal{T}, V, E);$

let  $Q$  be the disjunction of ABoxes  $\mathcal{A}_1 \vee \dots \vee \mathcal{A}_n$ ;

**if** there exists  $i \in \{1, \dots, n\}$

**such that**  $\langle \mathcal{T}, \mathcal{A}_i \rangle$  is satisfiable

**and**  $\text{cert}(V, \langle \mathcal{T}, \mathcal{A}_i \rangle) = E$

**and**  $\langle \mathcal{T}, \mathcal{A}_i \rangle \not\models q(\vec{t})$

**then return** false

**else return** true

**end**

Figure 1: Algorithm View-based-answer $_T$

a model of  $\mathcal{T}$  (since  $\mathcal{I}'$  does not modify the interpretation of the symbols occurring in  $\mathcal{T}$  with respect to  $\mathcal{D}$ ); (ii)  $\mathcal{I}'$  is a model of  $\text{freeze}(Q^\alpha, \mathcal{T}, V, E)$  (since  $\mathcal{I}$  is a model of  $\text{grounding}(Q^\alpha, \mathcal{T}, \mathcal{A}')$ ); (iii)  $\mathcal{I}' \not\models q(\vec{t})$  (since  $\mathcal{I} \not\models q(\vec{t})$  and  $\mathcal{I}'$  does not modify the interpretation of the symbols occurring in  $\mathcal{T}$  with respect to  $\mathcal{D}$ ). Now, by hypothesis  $\vec{t} \in \text{cert}(q, \langle \mathcal{T}, \text{freeze}(Q^\alpha, \mathcal{T}, V, E) \rangle)$ , and since  $\mathcal{I}'$  is a model of  $\langle \mathcal{T}, \text{freeze}(Q^\alpha, \mathcal{T}, V, E) \rangle$ , it follows that  $\mathcal{I}' \models q(\vec{t})$ . Contradiction. Therefore, there exists no ABox  $\mathcal{A}'$  such that  $\text{MOD}(\langle \mathcal{T}, \mathcal{A}' \rangle)$  is a T-solution for  $(\mathcal{T}, V, E)$  and such that  $\langle \mathcal{T}, \mathcal{A}' \rangle \not\models q(\vec{t})$ . Hence,  $\text{cert}(q, \langle \mathcal{T}, \text{freeze}(Q^\alpha, \mathcal{T}, V, E) \rangle) \subseteq \text{vba}_T(q, \mathcal{T}, V, E)$ . Moreover, it is immediate to see that  $\text{vba}_T(q, \mathcal{T}, V, E) \subseteq \text{cert}(q, \langle \mathcal{T}, \text{freeze}(Q^\alpha, \mathcal{T}, V, E) \rangle)$ .  $\square$

Based on the above lemma, we define the general algorithm View-based-answer $_T$  for deciding whether a tuple  $\vec{t}$  is a view-based answer to a query  $q$  with respect to  $(\mathcal{T}, V, E)$  in a DL  $\mathcal{L}$  under the TBox-centered semantics. The algorithm, displayed in Figure 1, assumes the existence of a procedure  $\text{rollUp}(\mathcal{L}, \mathcal{T}, q)$  for computing a roll-up of a CQ  $q$  under  $\mathcal{T}$  in the DL  $\mathcal{L}$ . The algorithm allows us to prove the following general result about decidability of view-based query answering under TBox-centered semantics.

**Theorem 12** *Let  $\mathcal{L}$  be a DL without UNA and such that the following problems are decidable/computable in  $\mathcal{L}$ : (i) KB satisfiability; (ii) conjunctive query answering; (iii) computing a roll-up of a CQ. Then, view-based conjunctive query answering in  $\mathcal{L}$  under TBox-centered semantics is decidable.*

*Proof.* The proof follows from correctness of the algorithm View-based-answer $_T$ , which is a direct consequence of Lemma 11.  $\square$

**Results for the DL-Lite family** We now consider view-based query answering for the DLs of the  $DL-Lite$  family, for which the UNA holds. We first analyze model-centered semantics.

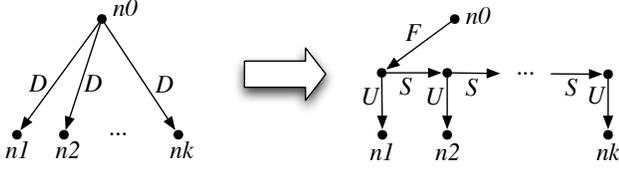


Figure 2: Representation of a graph

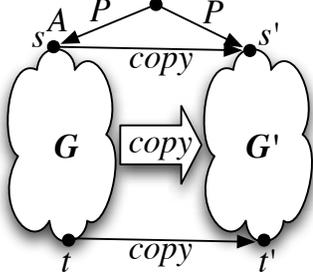


Figure 3: View instance used in Theorem 14

**Theorem 13** *View-based conjunctive query answering under model-centered semantics in  $DL-Lite_R$  is in LOGSPACE with respect to data complexity.*

*Proof.* Follows from the fact that in  $DL-Lite_R$  there are no functionality assertions that introduce equalities that could affect the freezing of  $\alpha(V, E)$ . Thus, we can apply Theorem 7 even with UNA, and reduce in LOGSPACE view-based query answering under model-centered semantics to query answering. Indeed, in  $DL-Lite_R$  adopting or not UNA does not affect conjunctive query answering.  $\square$

**Theorem 14** *View-based conjunctive query answering under model-centered semantics in  $DL-Lite_F$  is NLOGSPACE-hard with respect to data complexity.*

*Proof.* We exhibit a reduction from reachability in directed graphs, which is NLOGSPACE-hard. Let  $G = \langle N, D \rangle$  be a directed graph, where  $N$  is the set of nodes and  $D$  the set of directed edges. Reachability is the problem of deciding, given two nodes  $s, t \in N$  whether there is an oriented path formed by edges in  $D$  that, starting from  $s$  allows to reach  $t$ . We consider the graph represented through the functional relations first-child  $F$ , next-sibling  $S$ , and  $U$  (cf. Figure 2).

Given a directed graph  $G = \langle N, F, S, U \rangle$ , we define the following triple  $(\mathcal{T}, V, E)$ :

- $\mathcal{T}$  is the  $DL-Lite_F$  TBox whose alphabet consists of the atomic concept  $A$  and the atomic roles  $F, S, U, copy, P$ , and containing the assertions  $\{(\text{funct } \mathcal{R}) \mid \mathcal{R} \in \{F, S, U, copy, P\}\}$ .
- $V = \langle v_F, v_S, v_U, v_A \rangle$  with (cf. Figure 3):

$$v_{\mathcal{R}}(x, y) \leftarrow \mathcal{R}(x, y), copy(x, x'), copy(y, y'), \mathcal{R}(x', y'), \\ \text{for } \mathcal{R} \in \{F, S, U\}$$

$$v_A(x) \leftarrow A(x), copy(x, x'), P(z, x), P(z, x')$$

- $E = \langle E_F, E_S, E_U, E_A \rangle$ , with:

$$E_{\mathcal{R}} = \mathcal{R}, \quad \text{for } \mathcal{R} \in \{F, S, U\}$$

$$E_A = \{s\}$$

Notice that, for each  $M$ -solution  $\mathcal{W}$  for  $(\mathcal{T}, V, E)$  and each interpretation  $\mathcal{I} \in \mathcal{W}$ , by the assertions in  $\mathcal{T}$ , we have that  $\mathcal{R}^{\mathcal{I}}$  is a function, for  $\mathcal{R} \in \{F, S, U, copy, P\}$ . Consider the query

$$q(x) \leftarrow copy(x, x).$$

We show that  $t$  is reachable from  $s$  in  $G$  iff  $t \in vba_M(q, \mathcal{T}, V, E)$ .

“ $\Rightarrow$ ” Assume that  $t$  is reachable from  $s$  in  $G$ . We show by induction on the length  $\ell$  of a path from  $s$  to a node  $n$  reachable from  $s$  in  $G$  that  $n \in vba_M(q, \mathcal{T}, V, E)$ . Base case:  $\ell = 0$ . Then  $n = s$ . Let  $\mathcal{W}$  be an  $M$ -solution for  $(\mathcal{T}, V, E)$ , and let  $\mathcal{I} \in \mathcal{W}$ . Since  $s \in E(v_A) = cert(v_A, \mathcal{W})$ , there exist objects  $o, o' \in \Delta^{\mathcal{I}}$  such that  $s^{\mathcal{I}} \in A^{\mathcal{I}}$ ,  $(s^{\mathcal{I}}, o') \in copy^{\mathcal{I}}$ ,  $(o, s^{\mathcal{I}}) \in P^{\mathcal{I}}$ , and  $(o, o') \in P^{\mathcal{I}}$ . Since  $P^{\mathcal{I}}$  is a function, we have that  $s^{\mathcal{I}} = o'$ , and  $(s^{\mathcal{I}}, s^{\mathcal{I}}) \in copy^{\mathcal{I}}$ . Hence  $s = n \in vba_M(q, \mathcal{T}, V, E)$ . Inductive case:  $\ell = k$ . Let  $n'$  be a node in  $G$  with  $(n, n') \in F$  (the cases for  $S$  and  $U$  are analogous), and assume towards a contradiction that  $n' \notin vba_M(q, \mathcal{T}, V, E)$ . Then there is an  $M$ -solution  $\mathcal{W}$  for  $(\mathcal{T}, V, E)$  and an  $\mathcal{I} \in \mathcal{W}$  such that  $(n'^{\mathcal{I}}, n'^{\mathcal{I}}) \notin copy^{\mathcal{I}}$ . By inductive hypothesis,  $n \in vba_M(q, \mathcal{T}, V, E)$ , hence  $(n^{\mathcal{I}}, n^{\mathcal{I}}) \in copy^{\mathcal{I}}$ . Since  $(n, n') \in E(v_F) = cert(v_F, \mathcal{W})$ , there exist objects  $o, o' \in \Delta^{\mathcal{I}}$  such that  $(n^{\mathcal{I}}, n'^{\mathcal{I}}) \in F^{\mathcal{I}}$ ,  $(n^{\mathcal{I}}, o) \in copy^{\mathcal{I}}$ ,  $(n'^{\mathcal{I}}, o') \in copy^{\mathcal{I}}$ , and  $(o, o') \in F^{\mathcal{I}}$ . Since  $copy^{\mathcal{I}}$  is a function,  $o = n^{\mathcal{I}}$ , and since  $F^{\mathcal{I}}$  is a function,  $o' = n'^{\mathcal{I}}$ . Hence  $(n'^{\mathcal{I}}, n'^{\mathcal{I}}) \in copy^{\mathcal{I}}$ , which is a contradiction.

“ $\Leftarrow$ ” If  $t$  is not reachable from  $s$  in  $G$ , it is easy to construct an  $M$ -solution  $\mathcal{W}$  for  $(\mathcal{T}, V, E)$  with an interpretation  $\mathcal{I} \in \mathcal{W}$  such that  $(t^{\mathcal{I}}, t^{\mathcal{I}}) \notin copy^{\mathcal{I}}$ .  $\square$

**Theorem 15** *View-based conjunctive query answering under model-centered semantics in  $DL-Lite_A$  is NLOGSPACE-complete with respect to data complexity.*

*Proof.* NLOGSPACE-hardness follows directly from Theorem 14. To show membership in NLOGSPACE, we have to first propagate equalities due to functionalities in the freezing of  $\alpha(V, E)$  (or, equivalently, we add the definition of the equality predicate in the perfect rewriting of the query) and then we perform standard query answering in  $DL-Lite_A$ .  $\square$

Then, we turn our attention to TBox-centered semantics. Since in the  $DL-Lite$  family we consider only atomic ABoxes, we get directly the following result.

**Theorem 16** *View-based conjunctive query answering under TBox-centered semantics in  $DL-Lite_R, DL-Lite_F,$  and  $DL-Lite_A$  is CONP-complete w.r.t. data complexity.*

*Proof.* Membership in CONP follows from the fact that answering CQs for all three DLs is LOGSPACE with respect to data complexity and from Theorem 9, while CONP-hardness follows from Theorem 10.  $\square$

**Results for the  $\mathcal{EL}$  family** We now consider view-based query answering for the DLs  $\mathcal{EL}$  and  $\mathcal{ELH}$ . The following results hold both with and without UNA. First, we analyze model-centered semantics.

**Theorem 17** *View-based conjunctive query answering under model-centered semantics in  $\mathcal{EL}$  and  $\mathcal{ELH}$  is PTIME-complete with respect to data complexity.*

*Proof.* Without UNA, the result follows from Theorem 7 and from the results of (Calvanese *et al.* 2006; Krisnadhi & Lutz 2007; Rosati 2007). For the case with UNA, we observe again that there are no number restrictions that could affect the freezing of  $\alpha(V, E)$ .  $\square$

Then, we turn our attention to TBox-centered semantics.

**Theorem 18** *View-based conjunctive query answering under TBox-centered semantics in  $\mathcal{EL}$  and  $\mathcal{ELH}$  is CONP-complete with respect to data complexity.*

*Proof.* Membership in CONP follows from the algorithm **View-based-answer<sub>T</sub>** and from a modification of the roll-up procedures for  $\mathcal{EL}$  and  $\mathcal{ELH}$  presented in (Rosati 2007). Hardness follows from Theorem 10.  $\square$

**Results for expressive DLs** We now examine view-based query answering for ontologies expressed as general TBoxes in expressive DLs. First, we take into account the model-centered semantics.

**Theorem 19** *View-based conjunctive query answering under model-centered semantics in  $\mathcal{ALC}$  is CONP-hard with respect to data complexity.*

*Proof.* Without UNA, the result follows from Theorem 7 and from the fact that conjunctive query answering in  $\mathcal{ALC}$  is CONP-hard with respect to data complexity (Schaerf 1993). For the case with UNA, we observe again that number restrictions are not present.  $\square$

The above CONP-hardness result actually holds already for  $\mathcal{AL}$ , for which also conjunctive query answering is CONP-hard (Calvanese *et al.* 2006).

**Theorem 20** *View-based conjunctive query answering under model-centered semantics in  $\mathcal{SHIQ}$ , and hence in  $\mathcal{ALCQI}$ , is in CONP with respect to data complexity.*

*Proof.* The result follows from Propositions 5 and 6, and from the fact that conjunctive query answering in  $\mathcal{ALCQI}$  and in  $\mathcal{SHIQ}$  is in CONP with respect to data complexity (Levy & Rousset 1998; Ortiz, Calvanese, & Eiter 2006; Glimm *et al.* 2007).  $\square$

Then, we analyze the complexity of view-based query answering under TBox-centered semantics.

**Theorem 21** *View-based conjunctive query answering under TBox-centered semantics in  $\mathcal{ALC}$  is  $\Pi_2^p$ -hard with respect to data complexity.*

*Proof (sketch).* The reduction is from 2-QBF validity. Given a 2-QBF formula of the form  $\phi = \forall \bar{x}. \exists \bar{y}. f(\bar{x}, \bar{y})$ , where  $\bar{x} = x_1 \dots, x_n$ ,  $\bar{y} = y_1, \dots, y_m$  and  $f$  is a 3-CNF over the propositional variables  $\bar{x}$  and  $\bar{y}$ , i.e., a formula of

the form  $f = c_1 \wedge \dots \wedge c_k$ , where each  $c_i$  is of the form  $c_i = l_1^i \vee l_2^i \vee l_3^i$  and each  $l_j^i$  is a literal over the variables  $\bar{x}$  and  $\bar{y}$ . We define the following triple  $(\mathcal{T}, V, E)$ :

- $\mathcal{T}$  is the following  $\mathcal{ALC}$ -TBox:

$$\begin{aligned} \top &\sqsubseteq C_t \sqcup C_f \\ C_t &\sqsubseteq \neg C_f \\ D_t &\sqsubseteq C_t \\ D_f &\sqsubseteq C_f \\ \exists Val.C_t &\sqsubseteq \forall Val.C_t \\ \exists Val.C_f &\sqsubseteq \forall Val.C_f \\ \exists Val.C_t &\sqsubseteq \forall Comp.\forall Val.C_f \\ \exists Val.C_f &\sqsubseteq \forall Comp.\forall Val.C_t \\ C_{ex} &\sqsubseteq \exists Val.\top \end{aligned}$$

- $V$  is constituted by the following view definitions:

$$\begin{aligned} v_{ex}(x) &\leftarrow C_{ex}(x) \\ v_{comp}(x, y) &\leftarrow Comp(x, y) \\ v_{Lit_1}(x, y) &\leftarrow Lit_1(x, y) \\ v_{Lit_2}(x, y) &\leftarrow Lit_2(x, y) \\ v_{Lit_3}(x, y) &\leftarrow Lit_3(x, y) \\ v_{univ}(x) &\leftarrow C_{univ}(x) \\ v_{GX}(x) &\leftarrow C_{univ}(x), Val(x, y), TV(y, y) \\ v_{EX}(y) &\leftarrow C_{univ}(x), Val(x, y), TV(y, y) \\ v_{TV}(x) &\leftarrow TV(x, x) \\ v_{D_t}(x) &\leftarrow D_t(x) \\ v_{D_f}(x) &\leftarrow D_f(x) \\ v_\phi &\leftarrow Lit_1(x, y_1), Lit_2(x, y_2), Lit_3(x, y_3), \\ &\quad Val(y_1, z_1), Val(y_2, z_2), Val(y_3, z_3), \\ &\quad C_f(z_1), C_f(z_2), C_f(z_3) \end{aligned}$$

- $E$  is the following  $V$ -extension, where we denote with  $E(v)$  the extension associated to view  $v$ :

$$\begin{aligned} E(v_{ex}) &= \{y_1, \dots, y_m, \bar{y}_1, \dots, \bar{y}_n\} \\ E(v_{comp}) &= \{\langle x_1, \bar{x}_1 \rangle, \dots, \langle x_n, \bar{x}_n \rangle, \\ &\quad \langle y_1, \bar{y}_1 \rangle, \dots, \langle y_m, \bar{y}_m \rangle\} \\ E(v_{Lit_1}) &= \{\langle c_i, l_i^1 \rangle \mid 1 \leq i \leq n\} \\ E(v_{Lit_2}) &= \{\langle c_i, l_i^2 \rangle \mid 1 \leq i \leq n\} \\ E(v_{Lit_3}) &= \{\langle c_i, l_i^3 \rangle \mid 1 \leq i \leq n\} \\ E(v_{univ}) &= \{x_1, \dots, x_m, \bar{x}_1, \dots, \bar{x}_n\} \\ E(v_{GX}) &= \{x_1, \dots, x_m, \bar{x}_1, \dots, \bar{x}_n\} \\ E(v_{EX}) &= \{0, 1\} \\ E(v_{TV}) &= \{0, 1\} \\ E(v_{D_t}) &= \{1\} \\ E(v_{D_f}) &= \{0\} \\ E(v_\phi) &= \{\langle \rangle\} \end{aligned}$$

It can be shown that the formula  $\phi$  is satisfiable iff  $\langle \mathcal{T}, V, E \rangle$  has no  $T$ -solution.

Then, we define a query  $q \leftarrow C(a)$  where  $C$  is a concept that does not occur in  $\mathcal{T}$  and  $V$ . It is immediate to see that  $\langle \rangle \in vba_{\mathcal{T}}(q, \mathcal{T}, V, E)$  iff  $\langle \mathcal{T}, V, E \rangle$  has no  $T$ -solution. Consequently, the claim holds. Finally notice that, if a  $T$ -solution exists, then there exists one obtained from an ABox that is atomic.  $\square$

Notice that the reduction given in the above proof can be immediately rephrased in  $\mathcal{AL}$ , by avoiding the explicit use of concept disjunction and qualified existential concepts

(which is possible due to the fact that general concept inclusion assertions are allowed in the TBox). Hence, the above  $\Pi_2^p$ -hardness result holds also for  $\mathcal{AL}$ .

As for the upper bound, we are able to provide results for those DLs in which CQs can be rolled up while staying in the same language, in particular for  $\mathcal{ALCQI}$ .

**Theorem 22** *View-based conjunctive query answering under TBox-centered semantics in  $\mathcal{ALCQI}$  is  $\Pi_2^p$ -complete with respect to data complexity.*

*Proof.* The lower bound follows from the proof of Theorem 21. The upper bound follows from the roll-up-based algorithm  $\text{View-based-answer}_T$  for view-based query answering under ABox-centered semantics presented above, and from the roll-up procedure for  $\mathcal{DLR}$  presented in (Calvanese, De Giacomo, & Lenzerini 2008), which constitutes also a correct roll-up of CQs for  $\mathcal{ALCQI}$ .  $\square$

Notice that, for  $\mathcal{SHIQ}$ , we cannot directly apply the roll-up procedure of (Glimm *et al.* 2007), since it produces a roll-up expressed in  $\mathcal{SHIQ}^\square$ . Similarly, for  $\mathcal{ALC}$  we cannot directly apply the roll-up procedure for  $\mathcal{ALCQI}$ . It remains open to determine roll-up procedures for  $\mathcal{SHIQ}$  and  $\mathcal{ALC}$  that stay in the same language.

## Discussion and conclusions

In this paper we have presented a first study on view-based query answering in DLs. The framework we have introduced distinguishes between different semantics for the problem, corresponding to different refinements of the notion of solution. We have related view-based query answering to privacy-aware information access, and we have presented several algorithms and complexity results for various DLs, both in the model-centered and in the TBox-centered semantics. Figure 4 summarizes the complexity results measured with respect to the size of view extensions.

The work presented in this paper will be continued in different directions. In particular, we will address view-based query answering under the language-centered semantics, and we will study the impact of varying (in particular, restricting) the language used to express the views on the complexity of view-based query answering.

## Acknowledgments

This research has been partially supported by FET project TONES (Thinking ONtologiES), funded by the EU under contract number FP6-7603, by project HYPER, funded by IBM through a Shared University Research (SUR) Award grant, and by MIUR FIRB 2005 project “Tecnologie Orientate alla Conoscenza per Aggregazioni di Imprese in Internet” (TOCALIT).

## References

- Abiteboul, S.; Hull, R.; and Vianu, V. 1995. *Foundations of Databases*. Addison Wesley Publ. Co.
- Baader, F.; Calvanese, D.; McGuinness, D.; Nardi, D.; and Patel-Schneider, P. F., eds. 2003. *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press.
- Baader, F.; Brandt, S.; and Lutz, C. 2005. Pushing the  $\mathcal{EL}$  envelope. In *Proc. of the 19th Int. Joint Conf. on Artificial Intelligence (IJCAI 2005)*, 364–369.
- Beeri, C.; Levy, A. Y.; and Rousset, M.-C. 1997. Rewriting queries using views in description logics. In *Proc. of the 16th ACM SIGACT SIGMOD SIGART Symp. on Principles of Database Systems (PODS'97)*, 99–108.
- Calvanese, D.; De Giacomo, G.; Lenzerini, M.; and Vardi, M. Y. 2001. View-based query answering and query containment over semistructured data. In *Proc. of the 8th Int. Workshop on Database Programming Languages (DBPL 2001)*.
- Calvanese, D.; De Giacomo, G.; Lenzerini, M.; and Vardi, M. Y. 2005. View-based query processing: On the relationship between rewriting, answering and losslessness. In *Proc. of the 10th Int. Conf. on Database Theory (ICDT 2005)*, volume 3363 of *Lecture Notes in Computer Science*, 321–336. Springer.
- Calvanese, D.; De Giacomo, G.; Lembo, D.; Lenzerini, M.; and Rosati, R. 2006. Data complexity of query answering in description logics. In *Proc. of the 10th Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR 2006)*, 260–270.
- Calvanese, D.; De Giacomo, G.; Lembo, D.; Lenzerini, M.; and Rosati, R. 2007. Tractable reasoning and efficient query answering in description logics: The *DL-Lite* family. *J. of Automated Reasoning* 39(3):385–429.
- Calvanese, D.; De Giacomo, G.; and Lenzerini, M. 2000. Answering queries using views over description logics knowledge bases. In *Proc. of the 17th Nat. Conf. on Artificial Intelligence (AAAI 2000)*, 386–391.
- Calvanese, D.; De Giacomo, G.; and Lenzerini, M. 2008. Conjunctive query containment and answering under description logics constraints. *ACM Trans. on Computational Logic* 9(3). Available at <http://tocl.acm.org/accepted/282calvanese.pdf>.
- Glimm, B.; Horrocks, I.; Lutz, C.; and Sattler, U. 2007. Conjunctive query answering for the description logic  $\mathcal{SHIQ}$ . In *Proc. of the 20th Int. Joint Conf. on Artificial Intelligence (IJCAI 2007)*, 399–404.
- Halevy, A. Y. 2001. Answering queries using views: A survey. *Very Large Database J.* 10(4):270–294.
- Horrocks, I.; Patel-Schneider, P. F.; and van Harmelen, F. 2003. From  $\mathcal{SHIQ}$  and RDF to OWL: The making of a web ontology language. *J. of Web Semantics* 1(1):7–26.
- Horrocks, I.; Sattler, U.; and Tobies, S. 2000. Reasoning with individuals for the description logic  $\mathcal{SHIQ}$ . In McAllester, D., ed., *Proc. of the 17th Int. Conf. on Automated Deduction (CADE 2000)*, volume 1831 of *Lecture Notes in Computer Science*, 482–496. Springer.
- Krisnadhi, A., and Lutz, C. 2007. Data complexity in the  $\mathcal{EL}$  family of description logics. In *LPAR*, 333–347.
- Levy, A. Y., and Rousset, M.-C. 1998. Combining Horn rules and description logics in CARIN. *Artificial Intelligence* 104(1–2):165–209.

DL	model-centered	TBox-centered with atomic ABox	TBox-centered with general ABox
$DL\text{-Lite}_R$	in LOGSPACE	CONP-complete	
$DL\text{-Lite}_F, DL\text{-Lite}_A$	NLOGSPACE-complete	CONP-complete	
$\mathcal{EL}, \mathcal{ELH}$	PTIME-complete	CONP-complete	CONP-complete
$\mathcal{ALCQI}$	CONP-complete	$\Pi_2^P$ -complete	$\Pi_2^P$ -complete
$\mathcal{ALC}, \mathcal{SHIQ}$	CONP-complete	$\Pi_2^P$ -complete	$\Pi_2^P$ -hard

Figure 4: Data complexity of view-based query answering

Ortiz, M. M.; Calvanese, D.; and Eiter, T. 2006. Characterizing data complexity for conjunctive query answering in expressive description logics. In *Proc. of the 21st Nat. Conf. on Artificial Intelligence (AAAI 2006)*, 275–280.

Poggi, A.; Lembo, D.; Calvanese, D.; De Giacomo, G.; Lenzerini, M.; and Rosati, R. 2008. Linking data to ontologies. *J. on Data Semantics X*:133–173.

Rizvi, S.; Mendelzon, A. O.; Sudarshan, S.; and Roy, P. 2004. Rewriting techniques for fine-grained access control. In *Proc. of the ACM SIGMOD Int. Conf. on Management of Data*, 551–562.

Rosati, R. 2007. On conjunctive query answering in  $\mathcal{EL}$ . In *Proc. of the 2007 Description Logic Workshop (DL 2007)*, volume 250 of *CEUR Electronic Workshop Proceedings*, <http://ceur-ws.org/Vol-250/>.

Schaerf, A. 1993. On the complexity of the instance checking problem in concept languages with existential quantification. *J. of Intelligent Information Systems* 2:265–278.

Stouppa, P., and Studer, T. 2007. A formal model of data privacy. In Virbitskaite, I., and Voronkov, A., eds., *Ershov Memorial Conference*, volume 4378 of *Lecture Notes in Computer Science*, 400–408. Springer.

Ullman, J. D. 2000. Information integration using logical views. *Theoretical Computer Science* 239(2):189–210.

Zhang, Z., and Mendelzon, A. 2005. Authorization views and conditional query containment. In *Proc. of the 10th Int. Conf. on Database Theory (ICDT 2005)*, volume 3363 of *Lecture Notes in Computer Science*, 259–273. Springer.