Journal of Computer and System Sciences 78 (2012) 26–46

Contents lists available at ScienceDirect

Journal of Computer and System Sciences

www.elsevier.com/locate/jcss

View-based query answering in Description Logics: Semantics and complexity

Diego Calvanese [a,*], Giuseppe De Giacomo [b], Maurizio Lenzerini [b], Riccardo Rosati [b]

[a] KRDB Research Centre for Knowledge and Data, Free University of Bozen-Bolzano, Piazza Domenicani 3, I-39100 Bolzano, Italy
[b] Dipartimento di Informatica e Sistemistica, Sapienza Università di Roma, Via Ariosto 25, I-00185 Roma, Italy

ARTICLE INFO

Article history:
Received 1 October 2009
Received in revised form 3 September 2010
Accepted 10 February 2011
Available online 15 March 2011

Keywords:
View-based query processing
Description Logics
Ontologies
Conjunctive queries
Incomplete information
Certain answers
Privacy-aware data management
Data complexity
DL-Lite family
EL-family

ABSTRACT

View-based query answering is the problem of answering a query based only on the precomputed answers to 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 providing two 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 privacy-aware access to ontologies. Then, we provide decidability results, algorithms, and data complexity characterizations for view-based query answering in several Description Logics, ranging from those with limited modeling capability to highly expressive ones.

1. Introduction

View-based query processing is the problem of processing a query under the assumption that the only accessible extensional information consists of the precomputed answers to a set of queries, called views. Several articles in the literature point out that this problem [1–3] is relevant in many aspects of information management, including query optimization, data warehousing, data integration, and query answering with incomplete information. In all these contexts, the problem arises of answering a query posed to a database only on the basis of the information about a set of views, which are again queries over the same database. In query optimization, the problem is relevant because using the views may speed up query processing. In data integration, the views represent the only information sources accessible to answer a query. A data warehouse can be seen as a set of materialized views, and, therefore, query processing reduces to view-based query answering. Finally, since the views provide partial knowledge on the database, view-based query processing can be seen as a special case of query answering with incomplete information.

In this article, the context that we are most interested in is the one of privacy-aware access to information. In the logical approach to privacy-aware data access, each user (or, class of users) is associated with a set of views, called authorization

* Corresponding author. Fax: +39 0471016009.
E-mail addresses: calvanese@inf.unibz.it (D. Calvanese), degiacomo@dis.uniroma1.it (G. De Giacomo), lenzerini@dis.uniroma1.it (M. Lenzerini), rosati@dis.uniroma1.it (R. Rosati).
URLs: http://www.inf.unibz.it/~calvanese/ (D. Calvanese), http://www.dis.uniroma1.it/~degiacomo/ (G. De Giacomo), http://www.dis.uniroma1.it/~lenzerini/ (M. Lenzerini), http://www.dis.uniroma1.it/~rosati/ (R. Rosati).

0022-0000/$ – see front matter
doi:10.1016/j.jcss.2011.02.011

views, which specify the information that the user is allowed to access [4–6]. View-based query answering in this setting captures the requirement that only information deriving from such views can be revealed to the user.

There are two approaches to view-based query processing, called *query rewriting* and *query answering*, respectively. In the former approach, originated with [7], we are given a query q and a set of view definitions, and the goal is to reformulate the query into an expression that refers to the views (or only to the views), and provides the answer to q when evaluated over the view extension. Typically, the rewriting is expressed in the same language used for both the query and the views. The latter approach, originated in [2], takes a more direct route: based on the view definitions and the view extensions, it tries to compute the so-called *certain answers*, i.e., the tuples satisfying the query in all databases consistent with the views. The relationship between the two approaches has been discussed, e.g., in [8,9]. In the rest of this article, we only deal with view-based query answering.

A large number of results is reported on view-based query answering in the recent database literature, both for the case of relational databases and for the case of semistructured and XML data. On the other hand, the problem is still largely unexplored in the context of Description Logics. We discuss related work in detail in Section 6.

Description Logics [10] (DLs) were introduced in the early 80s in the attempt to provide a formal ground to Semantic Networks and Frames. Since then, they have evolved into knowledge representation languages that are able to capture virtually all class-based representation formalisms used in Artificial Intelligence, Software Engineering, and Databases. In particular, DLs have proved adequate as logic-based formalisms for expressing ontologies [11]. Here, by ontology we mean a formal representation of a domain of interest, expressed in terms of both intensional (concepts, attributes, and relations) and extensional (instances of concepts, attributes and relations) properties. When ontologies are expressed in DLs, the intensional and the extensional assertions form the so-called TBox, and ABox, respectively. In DLs, the domain of interest is modeled by means of concepts and roles, which denote classes of objects and binary relations between classes of objects, respectively. Concepts and roles can be denoted using expressions of a specified language, and the various DLs differ in the expressive power of such a language. One of the distinguishing features of the work on these logics is the detailed computational complexity analysis both of the associated reasoning algorithms, and of the logical implication problem that the algorithms are supposed to solve. By virtue of this analysis, most of these logics have optimal reasoning algorithms, and practical systems implementing such algorithms are now used in several projects.

In this article, we present a first 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. In particular, while a database can be considered as a single interpretation structure, an ontology is characterized by a set of models, and the answers to a query are those that are *certain*, i.e., those that satisfy the query in every model of this set. To address this issue, we follow the novel approach to consider the precomputed answers to views as the certain answers to the corresponding queries. 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*. Roughly speaking, given an ontology $\mathcal{K}$ with TBox $\mathcal{T}$ and ABox $\mathcal{A}$ expressed in a DL $\mathcal{L}$, and a set of views V with extensions E, a solution for $\langle \mathcal{T}, V, E\rangle$ is a set of interpretations for $\mathcal{K}$ which satisfy $\mathcal{T}$ and 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 specific properties for the notion of solution. In particular, we refer to two notions of solutions, yielding two different semantics for view-based query answering, called *model-centered* and *TBox-centered semantics*, respectively. In the model-centered semantics, we simply insist that a solution is a set of models of the TBox $\mathcal{T}$. In the TBox-centered semantics, we additionally require that such a set can be expressed in terms of an 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 $\langle \mathcal{T}, V, E\rangle$ that falsifies $q(\vec{t})$ and cannot be distinguished from the models of $\mathcal{K}$ by using V and $\mathcal{T}$. The two semantics defined in the framework correspond to adopting 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 the two semantics, and for different DLs, ranging from tractable ones (the *DL-Lite* family [12,13] and the $\mathcal{EL}$ family [14–16]), to more expressive ones ($\mathcal{ALC}$, and $\mathcal{SHIQ}$ [17–19]).

Generally speaking, our work shows that, for all DLs considered, the complexity of view-based query answering under the model-centered semantics is essentially the same as the complexity of computing the certain answers over an ontology. On the other hand, the complexity of view-based query answering under the TBox-centered semantics is consistently higher than under the model-centered semantics. This is an indication that the model-centered semantics may represent a good trade-off between the requirement of answering queries based on a set of views, and computing such answers efficiently.