When *DL-Lite* met OWL...

(extended abstract)

Claudio Corona, Emma Di Pasquale, Antonella Poggi, Marco Ruzzi, and Domenico Fabio Savo

Dip. di Informatica e Sistemistica, SAPIENZA University of Rome
lastname@dis.uniroma1.it

1 Introduction

DL-Lite is a family of Description Logics (DLs) whose aim is to capture some of the most popular conceptual modeling formalisms, such as Entity-Relationship model [4] and UML class diagrams[1], while preserving the tractability of the most important reasoning tasks, such as ontology satisfiability and query answering of arbitrary (union of) conjunctive queries (ground and not ground). More specifically, reasoning over ontologies of the *DL-Lite* family, is LOGSPACE in data complexity, and can be entirely delegated to a standard DBMS technology. We refer here to *DL-Lite*$_A$, the DL of the *DL-Lite* family that is closest to OWL 2.0. Note that *DL-Lite*$_A$ is in fact a fragment of OWL [2]. As such, it is possible to restrict the syntax of OWL so that it captures *DL-Lite*$_A$. Because of the lack of the space, we will not discuss further this topic.

We provide an expressive query language for *DL-Lite*$_A$ ontologies, named *SparSQL*, whose syntax is inspired by both SQL and SPARQL[3], one of the most popular W3C proposals as standard query language for OWL. Actually, *SparSQL* is an epistemic query language that implements the language *EQL-Lite(UCQ)* presented in [2]. Thus, query answering is in LOGSPACE and can be reduced to evaluate first-order logic queries over the ABox. Moreover, we stress the expressive power of *DL-Lite*$_A$ by presenting a series of features, that are beyond OWL and can be added to the language. Specifically, these features allow one to express *data properties of object properties*, *denial* and *identification constraints*, and they include a new ontology component, called the ECBox, which is a set of general form of constraints called *EQL constraints*, based on epistemic logics. Note that this is not the first attempt to add constraints to OWL. However, previous related work, e.g. [5], are technically and semantically incomparable with the approach presented in this paper.

Notably, all the contributions discussed in this work are currently implemented within the MASTRO system, a tool for ontology representation and reasoning that has *DL-Lite*$_A$ as proprietary core language. The main feature of MASTRO is to reduce all reasoning tasks, such as consistency checking and query answering to the evaluation of standard SQL queries over a DBMS. Note that one of the major benefits of using

[1] http://www.omg.org/uml/

[2] From now on we will refer to OWL 2.0, see http://www.webont.org/owl/1.1/.

[3] We refer to http://www.w3.org/TR/2008/REC-rdf-sparql-query-20080115/ for details about SPARQL.

DBMS technologies, is to allow for using huge amounts of possibly pre-existing data, to populate the ontology instances [7].

2 How to pose expressive queries over *DL-Lite$_A$* ontologies: *SparSQL*

It is well-known that open-world semantics, typically adopted to interpret DL ontologies, are essential for representing incomplete information, but make FOL queries over DL ontologies undecidable. To the best of our knowledge, the most expressive FOL fragment for which decidability of query answering has been proved in DLs, is the class of union of conjunctive queries (UCQs), which unfortunately, have limited expressive power.

Hence, a non monotonic epistemic query language, named *EQL-Lite($\mathcal{Q}$)*, was introduced in [2] to query arbitrary DL ontologies. Intuitively, this language is based on the idea that "we have complete information on what we know", which allows to query "what we know" by adopting a closed-world semantics. Thus, *EQL-Lite($\mathcal{Q}$)* queries are FOL queries, whose atoms, expressed in the *embedded query language* $\mathcal{Q}$, are epistemic formulas that extract, from an ontology, "what is known to hold", i.e. the *certain answers*. According to the results of [2], *EQL-Lite(UCQ)* is particularly suitable to express complicated queries over *DL-Lite$_A$* ontologies which can be answered in LOGSPACE w.r.t.the number of assertions in the ontology.

In the following example, we illustrate *SparSQL*, the query language actually implementing *EQL-Lite(UCQ)* in MASTRO.

Example 1. Consider the following fragment of *DL-Lite$_A$* ontology:

SubClassOf($Female\ Person$)
DisjointClasses($Female\ Male$)
ObjectPropertyDomain(MARRIES $Person$)
SubClassOf($Person$ DataMinCardinality($1\ SSN$))
SubClassOf($Male\ Person$)
ObjectPropertyRange(MARRIES $Person$)
SymmetricObjectProperty(MARRIES)

The intensional level of the ontology asserts that males and females are disjoint sets of persons, where all persons have at least one social security number and can marry other persons. Moreover, it says that if a person x marries a person y , then also y marries x.

Suppose that we want to know if there is not any person that is not *known* be a male or a female. This query, not expressible through a UCQ, can be caught by the following boolean *SparSQL* query:

```
VERIFY not exists (SELECT persons.x
                   FROM SparqlTable(SELECT ?x
                   WHERE{?x rdf:type 'Person'}) persons
                   EXCEPT( SELECT males.x
                           FROM SparqlTable(SELECT ?x
                                            WHERE{?x rdf:type 'Male'}) males
                           UNION
                           SELECT females.x
                           FROM SparqlTable(SELECT ?x
                                            WHERE{?x rdf:type 'Female'})
                                            females))
```

3 Extending *DL-Lite$_A$* beyond OWL

We now present some new features that are absent in OWL and can be added to *DL-Lite$_A$*.

Extending the intensional level It is possible to extend *DL-Lite$_A$* of *object property data*, *denial constraints* and *identification constraints*, as formally introduced in [3, 2].

- Object property data: to be closer to ER and UML formalisms, *DL-Lite$_A$* has been enriched with *object property data*, representing binary relations between pairs of objects and values [1], allowing to handle data properties of object properties;
- Denial constraints: they are intensional axioms used to guarantee that a certain *condition* is not satisfied by the ontology; specifically, the *condition* is expressed through a (boolean) UCQ;
- Identification constraints: they are intensional axioms used to identify a concept of an ontology by means of relationships with other objects or values. Informally, the relationships mentioned above can be simple properties of the ontology (i.e., paths of length 1), as an object property or a data property, or chains of simple properties (i.e., paths with a length ≥ 1).

Example 1 (cont.) The ontology of Example 1 can be enriched with the following axioms:

ObjectPropertyDataDomain($WeddingDate$ MARRIES)	KeyFor(SSN $Person$)
ObjectPropertyDataRange($WeddingDate$ `rdf:date`)	deny($q() \leftarrow$ MARRIES(X,X))

The new intensional axioms state that MARRIES is the domain of the object property data $WeddingDate$, while its range is `date`. Moreover, they state that SSN is an identifier for $Person$, and that nobody is married with himself.

Adding epistemic constraints : the ECBox We present a new component of a *DL-Lite$_A$* ontology, called ECBox, consisting of a set of *epistemic constraints*, named EQL constraints, formally introduced in [2]. It is worth noting that, as opposed to rules and other kinds of axioms, constraints are not interpreted as axioms allowing to infer the set of models of the ontology. On the contrary, constraints are interpreted as simple "checks" over the ontology set of models.

An EQL constraint is a ECBox axiom used to guarantee that a certain *condition* is satisfied by the ontology. Specifically, the *condition* is expressed through a (boolean) *SparSQL* query.

Let us explain the need for introducing the ECBox. It is well-known that *DL-Lite$_A$*, being conceived to keep data complexity of the main reasoning tasks within LOGSPACE, has renounced to express few constructs typically used in the ER (or UML) formalism, such as complete generalization and minimal cardinality (greater than 1) on relations. Hence, for example, it is not possible to assert in a *DL-Lite$_A$* ontology that a person is either a male or a female (and nothing else). However, by weakening the semantics and using a EQLC with the boolean *SparSQL* query showed in Example 1, it is possible to guarantee that if the ontology with constraints is satisfiable, then a person is either a male or a female, in all the models of the ontology.

4 Conclusion

We presented a new query language, called *SparSQL*, that is an implementation of the *EQL-Lite(UCQ)* epistemic query language presented in [2], whose syntax is inspired by both SQL and SPARQL, one of the most popular proposals for the standard query language for OWL. Moreover, we enriched the ontology language mentioned so far with a new set of constructs, beyond OWL, to handle data properties of object properties, and to express constraints that might be particularly useful for modeling ontologies of practical interest. It is worth noting that all along this paper we considered $\textit{DL-Lite}_A$ ontologies, having traditional ABoxes as for data layer. Actually, MASTRO is able to access through an ontology and an appropriate set of *mappings* [6] any data layer, provided that it is accessible through a standard SQL engine. Notably, both the fragment of OWL corresponding to $\textit{DL-Lite}_A$, the *SparSQL* query language and the expressive $\textit{DL-Lite}_A$ constraints that were presented, are currently implemented within the MASTRO system, and keep working when MASTRO is used to access a general data layer. Furthermore, the first experiments with the overall system are very encouraging.

As future work we plan to follow two main directions. On one hand, we plan to investigate the use of *SparSQL* to query ontologies written in DLs, other than $\textit{DL-Lite}_A$, e.g. OWL-DL and RDFS. On the other hand, we plan to run experiments with actual users, in order to compare the usability of MASTRO with the other tools for ontology management reasoning that are currently available.

References

1. D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, A. Poggi, and R. Rosati. Linking data to ontologies: The description logic $\textit{DL-Lite}_A$. In *Proc. of OWLED 2006*, volume 216 of *CEUR Electronic Workshop Proceedings,* `http://ceur-ws.org/Vol-216/`, 2006.
2. D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, and R. Rosati. EQL-Lite: Effective first-order query processing in description logics. In *Proc. of IJCAI 2007*, pages 274–279, 2007.
3. D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, and R. Rosati. Path-based identification constraints in description logics. In *Proc. of KR 2008*, 2008. To appear.
4. P. P. Chen. The Entity-Relationship model: Toward a unified view of data. *ACM Trans. on Database Systems*, 1(1):9–36, Mar. 1976.
5. B. Motik, I. Horrocks, and U. Sattler. Adding integrity constraints to OWL. In *Proceedings of the Workshop OWLED*, 2007.
6. A. Poggi, D. Lembo, D. Calvanese, G. De Giacomo, M. Lenzerini, and R. Rosati. Linking data to ontologies. *J. on Data Semantics*, X:133–173, 2008.
7. A. Poggi and M. Ruzzi. Ontology-based data access with MASTRO (demo). In *Proceedings of the Workshop OWLED*, 2007.