July 2017

The Mastro Protégé plug-in for OBDA

Giacomo RONCONI [a,b], Marco RUZZI [a,b], Valerio SANTARELLI [a,b] and Domenico Fabio SAVO [a,b]

[a] *Sapienza University of Rome*
⟨lastname⟩@dis.uniroma1.it
[b] *OBDA Systems*
⟨lastname⟩@obdasystems.com

Abstract. Ontology-based data access (OBDA) is a recent approach for accessing data where an ontology is connected to autonomous, and generally pre-existing, data repositories through mappings, so as to provide a high-level, conceptual view over such data. Mastro is a Java tool for OBDA developed at Sapienza University of Rome and at the startup OBDA Systems, which is able to manage an OBDA specification where the ontology is specified in *DL-Lite*. In this work, we present the Mastro plug-in for the popular ontology editor Protégé. By means of this plug-in, the users can specify and manage full OBDA specifications and execute SPARQL queries posed over the ontology level to access data stored in the underlying data sources.

Keywords. Ontology-based Data Access, *DL-Lite*, Mastro, Protégé

1. Introduction

Ontology-based Data Access (OBDA) is a recent paradigm for accessing data sources through the mediation of a concentual domain view, given in terms of an ontology [12]. OBDA features a three-level architecture composed of the ontology, which provides a formal description of the domain of interest, the data sources used in organizations for storing their information, and the mapping used to specify the semantic relationships between the ontology layer and the data sources. Hence, OBDA can be seen as a form of information integration, where the ontology, which is expressed in a logic-based language, replaces the global schema [11]. Currently the two most popular systems for OBDA are Mastro [3] and Ontop [2].

Mastro is developed by OBDA Systems[1] and Sapienza University of Rome, and has been used in recent years in numerous projects with important business partners from the private and public sectors, for example, see [1,13]. Mastro comes with its own commercial application, the Mastro Studio system [5], and now, in this paper, we introduce the Mastro plug-in for the popular Protégé [10] editor for OWL ontologies.

Ontologies in Mastro are specified through languages belonging to the *DL-Lite* [4] family of lightweight Description Logics, and support mappings in the standard R2RML format [8], and also in their native mapping language. Data sources are seen as relational

[1] `www.obdasystems.com`

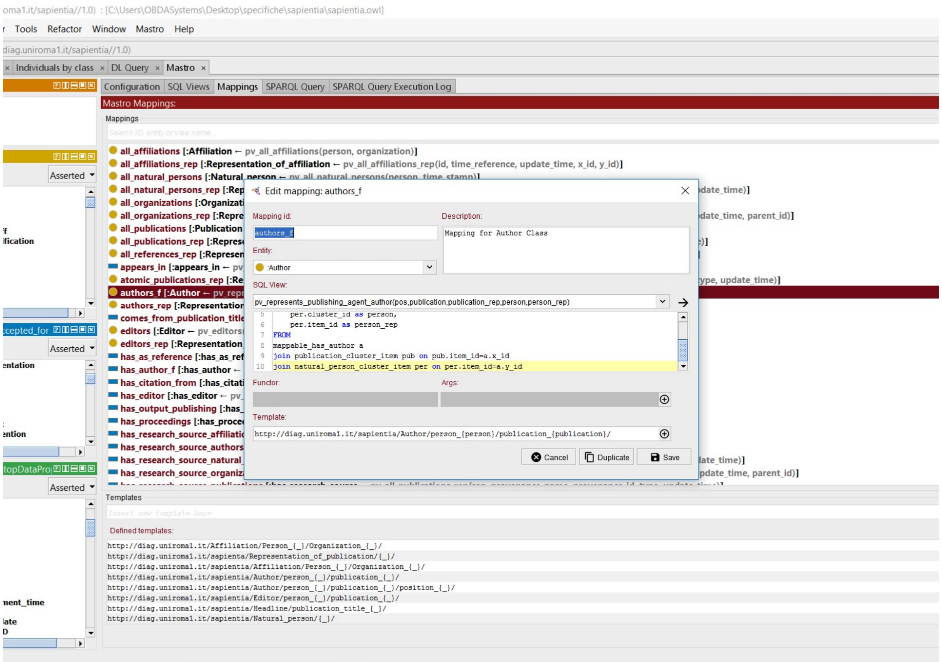

Figure 1. The Mappings sub-tab.

databases and can be accessed through SPARQL queries over the ontology, exploiting the query answering services provided by Mastro.

We demonstrate the Mastro Protégé plug-in through the Ontology of Multi-dimensional Research Assessment, or *Sapientia* [7], an OBDA specification developed and currently used within a project funded by Sapienza University of Rome which models aspects of assessing research activities and their impact on human knowledge and the economic system. For example, it deals with inter-relationships between research activities, between research activities and people's personal knowledge, and between research activities and other missions of individuals and institutions. The ontology is composed of fourteen modules, which formalize a wide array of activities, ranging from teaching, publishing, and research, to funding and preservation.

Users of the plug-in can inspect, query and edit the Sapientia OBDA specification through the plug-in. The ontology is managed through the features offered by Protégé, while the plug-in's Mappings and SQL Views tabs illustrate how the ontology is linked to the data sources through the mapping assertions, and how to edit or create new mappings and views; the SPARQL query tab and its execution log allow various querying functionalities; and lastly, the plugin allows to approximate OWL 2 into *DL-Lite*.

2. Overview of the Plug-in

Mastro's Protégé plug-in provides a full-fledged environment to define an OBDA specification for Mastro, and to access its query answering service. Along with these core features, the plug-in is equipped with other functionalities among which are: *(i)* the ap-

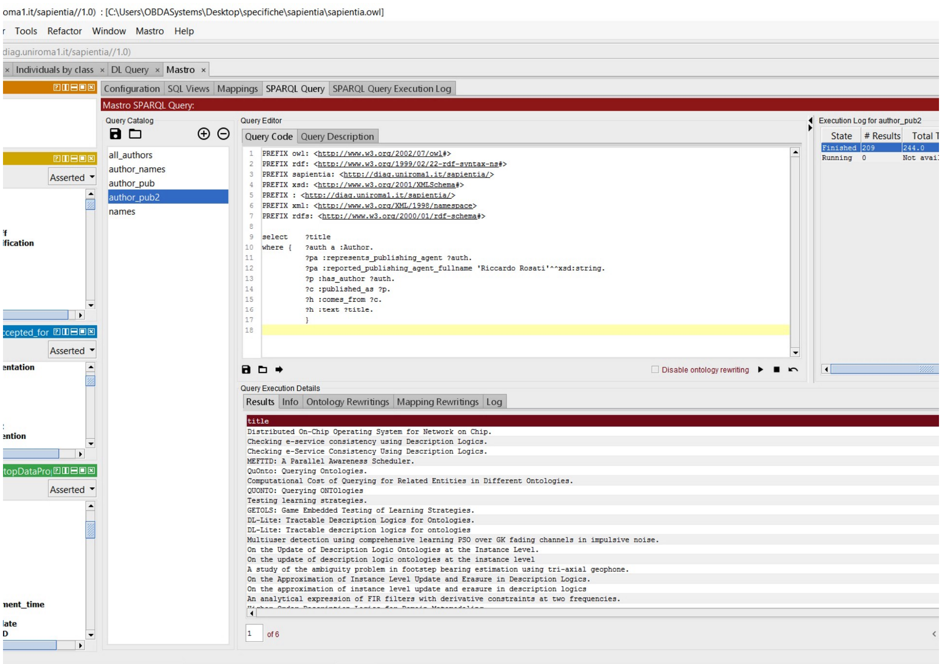

Figure 2. The SPARQL Query sub-tab.

proximation module that is based on the semantic approach presented in [6] for approximating OWL 2 ontologies in *DL-Lite*, the language supported by Mastro. This allows to load OWL 2 ontologies in Protégé, while using its *DL-Lite* representation for query answering with Mastro; *(ii)* in order to comply with the current recognized standards of the Semantic Web, Mastro is able to import R2RML mapping into its proprietary format and viceversa, and to export the results of the queries over the ontology into RDF.

The components of Mastro's plug-in are organized in five sub-tabs under the main Mastro tab, one item in the "Reasoner" menu, and one "Mastro" menu item which provides the features described above. Below we briefly describe each component.

Configuration. The Configuration sub-tab is used to create, open, and save a mapping specification, and to define the jdbc parameter connections to the source relational database, and to a database which Mastro uses handle query executions and to store all their information, e.g., results, execution time, ontology and mapping rewritings.

SQL Views. This sub-tab is used to create and inspect the views in the specification, and to specify assertions over them, i.e., inclusion assertions, disjointness assertions, and key dependency assertions. Each view is defined by its name and the SQL query code.

Mappings. The Mappings sub-tab, shown in Figure 1, is used to create and inspect the ontology predicate mappings in the specification, and to define the IRI templates that Mastro uses to build the answers from the data in the data source. Each ontology predicate mapping is identified by an ID, and contains the ontology predicate that is being mapped, a conjunctive query over the SQL views, and, optionally, a description.

SPARQL Query. The SPARQL Query sub-tab, shown in Figure 2, allows to define and execute a SPARQL query through Mastro over the ontology. Once the query has been executed, the user can consult its results and its ontology and mapping rewritings, and export the results in RDF or CSV formats. The user can also save the query into a catalog, which is managed through a file that can be loaded, exported, and edited. Additionally, Mastro saves all query executions into a log, allowing the user to visualize those of the selected query.

SPARQL Query Execution Log. This sub-tab provides information regarding all executions of SPARQL queries over an OBDA specification. The data shown in the log can be exported, and the queries re-executed.

The items in the Mastro menu show the details of the approximation performed by the semantic approximation module (*Approximation details*), allow to access a configuration panel (*Mastro properties*), to export Mastro's mappings in R2RML (*Export mappings to R2RML*), and to consult a tutorial of the Mastro plug-in (*Help*).

Acknowledgments. Work supported by MIUR under the SIR project "MODEUS" - grant n. RBSI14TQHQ.

References

[1] N. Antonioli, F. Castanò, C. Civili, S. Coletta, S. Grossi, D. Lembo, M. Lenzerini, A. Poggi, D. F. Savo, and E. Virardi. Ontology-based data access: The experience at the italian department of treasury. In *Proceedings of CAISE 2013, València, Spain*, volume 1017 of *CEUR Workshop Proceedings*, pages 9–16, 2013.

[2] D. Calvanese, B. Cogrel, S. Komla-Ebri, R. Kontchakov, D. Lanti, M. Rezk, M. Rodriguez-Muro, and G. Xiao. Ontop: Answering SPARQL queries over relational databases. *Semantic Web*, 8(3):471–487, 2017.

[3] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, A. Poggi, M. Rodriguez-Muro, R. Rosati, M. Ruzzi, and D. F. Savo. The Mastro system for ontology-based data access. *Semantic Web J.*, 2(1):43–53, 2011.

[4] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, and R. Rosati. Tractable reasoning and efficient query answering in description logics: The *DL-Lite* family. *JAR*, 39(3):385–429, 2007.

[5] C. Civili, M. Console, G. De Giacomo, D. Lembo, M. Lenzerini, L. Lepore, R. Mancini, A. Poggi, R. Rosati, M. Ruzzi, V. Santarelli, and D. F. Savo. MASTRO STUDIO: Managing ontology-based data access applications. *PVLDB*, 6:1314–1317, 2013.

[6] M. Console, J. Mora, R. Rosati, V. Santarelli, and D. F. Savo. Effective computation of maximal sound approximations of description logic ontologies. In *Proc. of ISWC*, volume 8797 of *LNCS*, pages 164–179. Springer, 2014.

[7] C. Daraio, M. Lenzerini, C. Leporelli, P. Naggar, A. Bonaccorsi, and A. Bartolucci. The advantages of an ontology-based data management approach: openness, interoperability and data quality. *Scientometrics*, 108(1):441–455, 2016.

[8] S. Das, S. Sundara, and R. Cyganiak. R2RML: RDB to RDF mapping language. W3C Recommendation, W3C, Sept. 2012. Available at `http://www.w3.org/TR/r2rml/`.

[9] F. Di Pinto, D. Lembo, M. Lenzerini, R. Mancini, A. Poggi, R. Rosati, M. Ruzzi, and D. F. Savo. Optimizing query rewriting in ontology-based data access. In *Proc. of EDBT*, pages 561–572. ACM Press, 2013.

[10] J. H. Gennari, M. A. Musen, R. W. Fergerson, W. E. Grosso, M. Crubézy, H. Eriksson, N. F. Noy, and S. W. Tu. The evolution of Protégé: an environment for knowledge-based systems development. *International Journal of Human-computer studies*, 58(1):89–123, 2003.

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

[12] 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.

[13] D. F. Savo, D. Lembo, M. Lenzerini, A. Poggi, M. Rodríguez-Muro, V. Romagnoli, M. Ruzzi, and G. Stella. MASTRO at work: Experiences on ontology-based data access. In *Proc. of DL*, volume 573 of *CEUR*, `ceur-ws.org`, pages 20–31, 2010.