

MappingPedia: A Collaborative Environment for R2RML Mappings

Freddy Priyatna¹, Edna Ruckhaus¹, Nandana Mihindikulasooriya¹,
Oscar Corcho¹, and Nelson Saturno²(✉)

¹ Ontology Engineering Group, Universidad Politécnica de Madrid, Madrid, Spain
{fpriyatna,eruckhaus,nmihindu,ocorcho}@fi.upm.es

² Dep. de Computación y T. I., Universidad Simón Bolívar, Caracas, Venezuela
nelsonsaturno@gmail.com

Abstract. Most of Semantic Web data is being generated from legacy datasets with the help of mappings, some of which may have been specified declaratively in languages such as R2RML or its extensions: RML and xR2RML. Most of these mappings are kept locally in each organization, and to the best to our knowledge, a shared repository that would facilitate the discovery, registration, execution, request and analysis of mappings doesn't exist. Additionally, many R2RML users do not have sufficient knowledge of the mapping language, and would probably benefit from collaborating with others. We present a demo of MappingPedia, a collaborative environment for storing and sharing R2RML mappings. It is comprised of five main functionalities: (1) Discover, (2) Share, (3) Execute, (4) Request, and (5) Analyze.

1 Introduction

Most Semantic Web data is being generated from existing legacy datasets with the help of R2RML mappings (<https://www.w3.org/TR/r2rml/>) or extensions, such as RML [2] and xR2RML [4].

Currently, a significant number of users have developed R2RML mappings but they have been kept locally in each organization, and to the best to our knowledge, a shared repository that would facilitate the registration, discovery, exploration and execution of mappings doesn't exist. Additionally, many R2RML users do not have sufficient knowledge of the mapping language, and would probably benefit from collaborative work with users who have experience in the development and specification of mappings in similar contexts. MappingPedia¹ provides such an environment where users may browse mappings that have used concepts in their domain of interest or request their development by other users.

This research is supported by the MobileAge (H2020/693319) project & the BES-2014-068449 FPI grant. We'd like to thank Boris and Ahmad for fruitful discussions.

¹ MappingPedia is available at <http://demo.mappingpedia.linkeddata.es/> and is currently in MobileAge (<http://www.mobile-age.eu/>) and Retele (<http://retele.linkeddata.es/>).

The work in [6] points out the difficulties encountered during the development of R2RML mappings, and presents two approaches for mapping construction: Ontology-driven and Database-driven; [3] extends them with the Model-driven and Result-driven approaches. MappingPedia is a tool to support collaborative mapping development and is independent of a specific editing approach.

MappingPedia integrates several tools developed for the exploitation of RDF data. The core of the integrated tools is morph-RDB [7], an RDB2RDF engine that follows the R2RML specification. It supports two operational modes: (1) RDF data generation from a relational database or CSV file, and (2) SPARQL to SQL Query translation according to R2RML mapping descriptions. MIRROR [1] is a system that generates two sets of R2RML mappings: First, it creates a set of mappings similar to the W3C Direct Mappings (<https://www.w3.org/TR/rdb-direct-mapping/>), and second, a set of R2RML mappings that result from the implicit knowledge encoded in relational database schemas. Loupe [5] is a tool that is aimed at the exploration of data sources that have been annotated, and their ontologies. Loupe conducts an inspection of the classes, properties and triples to gather explicit vocabulary, classes and property usage, and to discover implicit data patterns through a fine grained set of metrics.

We demonstrate the capabilities of MappingPedia for a set of use cases that take into account three key elements: the dataset, the ontologies used for the mappings and the mapping files themselves.

2 Architecture

Figure 1 presents the MappingPedia architecture. MappingPedia consists of two components: the Engine and the Interface. The MappingPedia Engine is responsible for storing mappings as RDF graphs in a Virtuoso server, executing mappings by connecting to morph-RDB, and taking care of the evolution of mappings through their storage in a GitHub repository. The MappingPedia Interface provides a web interface for end-users and REST interfaces for external applications. Additionally, it stores user data in a MySQL database and calls the Loupe API

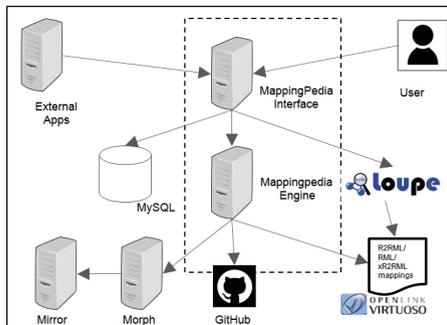


Fig. 1. Architecture of MappingPedia

to gather statistics. Loupe has been extended in order to analyze not only classes and properties, but also the value distribution of properties. This is the case of the R2RML properties `rr:class` and `rr:predicate`, where their values are the classes and properties used in mappings.

3 Use Cases

Table 1 presents eight use cases for MappingPedia and the actions that can be performed. A description of the functionalities follow:

Table 1. Scenarios in MappingPedia

Functionality	Use case	What the user has			MappingPedia allows to
		Dataset info	Mappings	Ontologies to map to	
Browse	UC1	No	No	No	Browse/Search mappings by metadata and ontology classes/properties
	UC2	No	No	Yes	
	UC3	No	Yes	Yes	<i>Mappings without dataset info not considered currently</i>
	UC4	No	Yes	Yes	
Request	UC5	Yes	No	No	Request the development of mappings
	UC6	Yes	No	Yes	
Share/Execute	UC7	Yes	Yes	No	Register Mappings and Generate RDF or SPARQL→SQL query translation
	UC8	Yes	Yes	Yes	

- **Browse.** Users may search for metadata defined in the MappingPedia ontology². The ontology includes properties for datasets and mapping files and reuses the Data Catalog (dcat) and Dublin Core (dc) ontologies. The user may also search by a class or property in a mapping; for example a user may search the mappings that contain `foaf:Person`. A screenshot of this functionality can be seen in Fig. 2.
- **Execute.** A user may execute R2RML mappings on MappingPedia using morph-RDB. If the mappings have not been developed, MIRROR can be used to generate the initial mappings; the output will be an RDF dataset. Additionally she may specify a SPARQL query using the concepts in the

² <http://ontology.linkeddata.es/publish/MappingpediaVocabulary/index-en.html>).

ontologies mapped, and the query will be rewritten into SQL according to the R2RML mapping descriptions.

- **Share.** A user may register mappings and dataset information.
- **Request.** A user may request the development of mappings. This resembles a ticket system where a request may be Open, In progress, Resolved (Fixed, Incomplete, Not Fixed) and Closed.
- **Analyze.** Statistics are gathered on all the classes and properties of ontologies that have been used in the mappings (e.g. MappingPedia, R2RML, Data Catalog, etc.); also, statistics may be requested over a period of time.

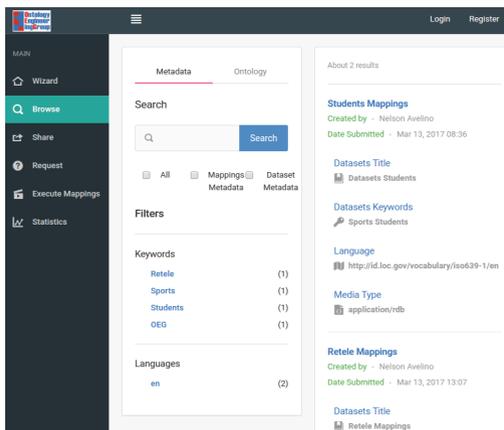


Fig. 2. Screenshot browse UI

3.1 Request and Analyze Functionalities

In the demo we will show the functionalities of MappingPedia with an emphasis on two cases: mapping request and mapping analysis.

Mapping request. There are two actors in a mapping request: the data owner and the mapping creator. The data owner wants to publish his dataset as Linked Data. He creates a request for mappings and provides information on his dataset. A mapping creator may assign himself a request for mappings, the dataset owner will be notified. Once the mappings have been uploaded the owner will again be notified. He may execute the mappings and according the outcome of the execution he may decide to close the request or reopen it.

Mapping analysis. The objective of mapping analysis is to provide a generic overview and analytics about the mappings to a user who is browsing. MappingPedia uses Loupe profiling services to generate information such as the number of mappings available, top classes being mapped, top properties being mapped, and the number of columns mapped for each mapping. Currently MappingPedia

is populated with the 685 English DBpedia mappings that are available in the RML format³. The analysis of those mappings is presented in the “Statistics” option. A screenshot of this functionality is illustrated in Fig. 3.

The screenshot shows a web interface for analyzing R2RML mappings. It features a sidebar with navigation icons and a main content area with two sections: 'Classes' and 'Properties'. Both sections include a search bar and a table of results.

Classes and properties for: <http://www.w3.org/ns/r2rml#>

Classes

Classes	Instance Ratio	Total Instances	Actions
http://www.w3.org/ns/r2rml#TriplesMap	1.7912	685	⋮
http://www.w3.org/ns/r2rml#SubjectMap	1.7912	685	⋮
http://www.w3.org/ns/r2rml#PredicateObjectMap	47.3132	18094	⋮
http://www.w3.org/ns/r2rml#ObjectMap	47.3132	18094	⋮

Showing 1 to 4 of 4 entries

Properties

Properties	Distinct Objects	Total Objects	Actions
http://www.w3.org/ns/r2rml#class	336	685	⋮
http://www.w3.org/ns/r2rml#objectMap	18094	18094	⋮

Fig. 3. Screenshot analyze UI

4 Conclusions and Future Work

We have presented MappingPedia, a collaborative environment that integrates various R2RML-based tools for the purpose of discovery, sharing, requesting and executing R2RML mappings. MappingPedia is in its first version and we envision several features that we will implement.

In the future we will integrate an R2RML mapping editor so that a mapping creator is able to work directly on MappingPedia once he has been assigned a mapping request. We will also integrate MappingPedia with data characterization tools in order to propose mappings based on the content of a dataset. Finally, we also plan to integrate MappingPedia with RML and xR2RML engines.

References

1. de Medeiros, L.F., Priyatna, F., Corcho, Ó.: MIRROR: automatic R2RML mapping generation from relational databases. In: Engineering the Web in the Big Data Era. ICWE 2015 (2015)
2. Dimou, A., Vander Sande, M., Colpaert, P., Verborgh, R., Mannens, E., Van de Walle, R.: RML: a generic language for integrated RDF mappings of heterogeneous data. In: Linked Data on the Web LDOW 2014 Co-located with WWW 2014

³ <http://mappings.dbpedia.org/server/mappings/en/pages/rdf/all>.

3. Heyvaert, P., Dimou, A., Verborgh, R., Mannens, E., de Walle, R.V.: Towards approaches for generating RDF mapping definitions. In: Posters & Demonstrations Track Co-located with International Semantic Web Conference ISWC 2015
4. Michel, F., Djimenou, L., Faron-Zucker, C., Montagnat, J.: Translation of relational and non-relational databases into RDF with xR2RML. In: International Conference on Web Information Systems and Technologies WEBIST 2015
5. Mihindukulasooriya, N., Poveda-Villalón, M., García-Castro, R., Gómez-Pérez, A.: Loupe - an online tool for inspecting datasets in the linked data cloud. In: Posters & Demonstrations Track Co-located with the International Semantic Web Conference ISWC 2015
6. Pinkel, C., Binnig, C., Haase, P., Martin, C., Sengupta, K., Trame, J.: How to best find a partner? an evaluation of editing approaches to construct R2RML mappings. In: Extended Semantic Web Conference ESWC 2014
7. Priyatna, F., Corcho, Ó., Sequeda, J.: Formalisation and experiences of R2RML-based SPARQL to SQL query translation using morph. In: WWW 2014