Enhancing the Maintainability of the Bio2RDF Project Using Declarative Mappings

Ana Iglesias-Molina, David Chaves-Fraga, Freddy Priyatna, and Oscar Corcho

Ontology Engineering Group, Universidad Politécnica de Madrid, Spain
ana.iglesiasm@upm.es
{dchaves,fpriyatna,ocorcho}@fi.upm.es

Abstract. Bio2RDF is one of the most popular projects that integrates and publishes biomedical datasets as Linked Data. The community has actively contributed to the generation of these datasets using ad-hoc programmed scripts. In the context of the Semantic Web, Ontology-Based Data Access (OBDA) approaches have been proposed to provide data access and transformation in a more standardized way, using declarative mapping languages. In this paper, we propose the use of an OBDA approach to provide an alternative to the way in which transformations into RDF are currently done in the Bio2RDF project, with the aim of enhancing its methodology in terms of understandability, reusability and maintainability. We describe the proposed methodology together with the declarative mappings creation process aiming to improve the aforementioned features. We compare the RDF dataset generated using our proposal with the latest release of Bio2RDF for a subset of the data sources that we have dealt with. Finally, we discuss the set of challenges that we face with this approach.

Keywords: Bio2RDF · OBDA · RML

1 Introduction

In the last decades, the amount of databases that have been created to store and share biological knowledge has heavily increased [1,2]. According to [3], there are more than 1600 biological databases that are publicly accessible online, including well-known examples, such as PubMed[1], UniProt[2] or KEGG[3]. Nowadays, these resources have become essential for researchers, as they rely on them to conduct much of their work.

Each biological data source contains information specific to its domain. This means that the knowledge of a concept (e.g. enzyme, transcription factor) is distributed in multiple data sources that are created by different institutions, usually represented in different formats and terminologies. A relevant challenge in this domain is how to integrate these data sources in order to provide a

[1] https://www.ncbi.nlm.nih.gov/pubmed/
[2] https://www.uniprot.org/
[3] https://www.genome.jp/kegg/

Table 1: Comparison of the methodology of Bio2RDF in its different releases and the proposed approach with declarative mappings. The features compared are the type of tool, how many can be used, and if it allows materialisation or virtualization.

Feature	Bio2RDF Release 1	Bio2RDF Releases 2 & 3	Declarative Mappings
Tool Type	Ad-hoc solution	Ad-hoc solution	General Purpose
# Tools	1 (myBio2RDF app)	1 (PHP scripts)	Many
Materialization	Yes	Yes	Yes
Virtualization	No	No	Yes

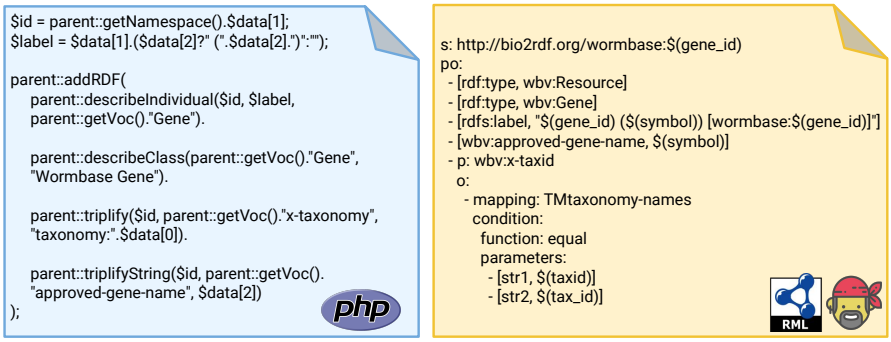
uniform and standard search interface that may allow researchers to find easily the data for their studies. One notable project that addresses this challenge with the use of a Semantic Web approach is Bio2RDF [4], an open source project, started in 2008, that integrates heterogeneous sources of biomedical data into Linked Data. For each biological database in its catalogue, Bio2RDF provides an ontology and a PHP script to transform data into RDF and publish it.

Over the years, the project has developed and refined its methodology, integrating more data sources in each release (now it gathers more than 40 datasets). Since its last release [5], many of the databases in Bio2RDF have been updated with more data, and some of them have even changed their structure. This implies that the scripts corresponding to those databases may not work as expected. Additionally, maintaining them is not an easy task for non-experts.

In the context of Semantic Web technologies, there are multiple approaches that have been proposed to transform heterogeneous data sources into Linked Data, without the need of having ad-hoc scripts. One that is widely accepted is Ontology-Based Data Access (OBDA) [6], where declarative mappings [7] are used to specify the relationship between the data sources and an ontology. We summarize in Table 1 the differences between the methodology used in Bio2RDF and using declarative mappings.

There are diverse languages to write the mappings, such as R2RML [8], a W3C recommendation for mapping relational databases; RML [9], an extension of R2RML for heterogeneous data sources; and the serialization of RML, YARRRML [10]. Along with these specifications there are also several engines able to process them [11,12]. These OBDA technologies provide access as RDF views in two different manners: materialised, where the data sources are transformed into RDF; or virtualised, where SPARQL queries are translated into the query language supported by the data sources using mapping rules.

In this paper we propose a change in the methodology that Bio2RDF follows to transform data into RDF, from using scripts to an OBDA approach. Our hypothesis is that using this approach we can help to improve the maintainability, reusability and understanding of the procedure to transform the data. We present the first steps done in this direction, the creation of the mappings for a subset of the datasets of Bio2RDF, the improvements in terms of completeness achieved by comparing our approach to the previous one and the set of challenges we face.



(a) Excerpt of an script in PHP **(b)** Excerpt of a mapping in YARRRML

Fig. 1: Motivating example: comparison of scripts vs mappings for Bio2RDF. Both figures show an extract of the script/mapping responsible for the transformation of the dataset WormBase. They indicate the subject (field `gene_id`), the classes it belongs to (`Gene` and `Resource`) and two triples (fields `tax_id` and `symbol`). In Figure 1b, one of the objects of the triples is the subject of another mapping (`TMtaxonomy-names`), where there is a join condition, to match the equal values of the two different sources (`tax_id` and `taxid`).

This paper is structured as follows: Section 2 provides a motivating example to remark the need to change the Bio2RDF methodology in order to overcome its current difficulties. Section 3 describes our approach to create declarative mappings. Section 4 shows the characteristics of the mappings, its potential use, makes a comparison of the completeness of the data from our approach and the one available in the Bio2RDF SPARQL endpoint, and ends showing the challenges of the approach. Section 5 presents related work and Section 6 presents the conclusions and areas for future work.

2 Motivating example

In this section, we show an example that motivates the need to enhance the methodology currently used by Bio2RDF. We compare how the rules for establishing the relationships between the data source and the ontology are created. In Figure 1a we show an example of the PHP script responsible for the transformation of the dataset Wormbase[4], while in Figure 1b the same generation rules are represented through RML [9], using its user-friendly serialization YARRRML [10]. Both of them specify which is the subject (field `gene_id`), the classes that the instances belong to (`Gene` and `Resource`), and two predicate-object maps (fields `tax_id` and `symbol`). In the PHP script there are also references to functions and classes defined in other parts of the script and other PHP files, as there is no separation between the mapping rules and the engine.

[4] `https://www.wormbase.org/`

Table 2: Formats of the data sources. Percentage of the datasets of the Bio2RDF project according to its format.

Format	Percentage of datasets
CSV/XLSX	36.95 %
Special or Undetermined	23.91 %
XML	19.56 %
RDF	8.69 %
JSON	6.52 %
RDB	4.34 %

In terms of completeness, we compare the number of instances of the class `Gene` of the data source WormBase available since release 3 in the SPARQL endpoint of Bio2RDF[5] and the data generated by our approach. We find a discrepancy between them, in the SPARQL endpoint the result is 50894 instances, and with our approach, 51255. This shows that currently accessible data on Bio2RDF is not up to date.

3 Approach

The full transformation process defined in the Bio2RDF methodology is a long process. For each one of the current collection of 43 datasets, an ontology and a script are created to do the transformation of data into RDF. All the resources used to do this process (ontology, scripts, documentation) are publicly available[6]. Although the datasets are defined in various data structures, the majority of them are tabular data, as we show in Table 2. The second most common format is special or undetermined, which means that the data is not available, or it is available but in its in proprietary format (e.g. GenBank). There are also XML, RDF, JSON and relational databases (RDB) in smaller proportion. Most of the data sources can be found publicly and available for its download.

In our approach we select datasets in tabular format, since they represent the main group of datasets, and create declarative mappings for each one. In the first step, using the proposal of [13], we define the mapping rules in a spreadsheet to facilitate the mapping creation process, since this approach is language independent and spreadsheets are a well-known tool. The rules specified in this step are later structured in the language's specification. The set of these rules are contained in triple maps (TM). They are composed of one subject, one source (where the characteristics of the data source are defined), and a variable number of predicate-object maps (POM), the triples to be formed from a subject.

The information needed to create mapping rules is extracted from two sources: the original data and the Bio2RDF resources (mainly PHP scripts, ontology and SPARQL endpoint). The scripts are the most useful source, since they define specifically which is the source data, the subject, the rules to generate triples that have to be generated, and the transformation functions applied to the data

[5] `http://bio2rdf.org/sparql`
[6] `https://github.com/bio2rdf`

(e.g. uppercase, concatenation). They, along with the source data, give information about the metadata, such as which is the separator of the file, whether there is more than one value on the same field or the type of data (e.g. integer, date). We represent that information in a declarative manner using the W3C recommendation CSVW [14].

In the second step, the spreadsheets containing mappings are translated into the most suitable mapping language depending on the type of the dataset, following the ideas presented in [15]. For example, for a relational database we translate it into R2RML; for CSV files and related, into YARRRML or RML. For this purpose we develop Mapeathor[7], a tool that translates the spreadsheets into mappings in R2RML, RML and its serialization YARRRML. Thus, with the declarative mappings and the metadata files in CSVW, we separate rules for transforming data from the engine.

4 Results and discussion

In this section we first show the characteristics of the created mappings and their potential use; second, we test the feasibility of our approach by comparing the number of instances obtained in the Bio2RDF SPARQL endpoint and using the mappings in an OBDA engine; and third, we discuss the main challenges that we face with this approach.

4.1 Bio2RDF mappings

In this work the mappings for 14 datasets have been created, and are publicly available[8]. The mapping language used is RML for the datasets in CSV or related, and R2RML for the relational databases. All the characteristics are summarised in Table 3. We discuss several aspects affecting the complexity of processing as discussed in [16]:

- **Number of Triple Maps (TMs):** Each TM generates a new subject, and it can come from the same or different source as other subjects in other TMs. More than half of the mappings create multiple subjects per source.
- **Number of sources:** Half of the datasets are contained in more than one file. As a result, the number of mappings increases with it.
- **Number of Predicate-Object Maps (POMs):** They specify the triples that are to be created in conjunction with the subject.
- **Number of different predicates and objects:** The separate count of predicates and objects, since they can appear in different triples.
- **Number of joins:** A join links two objects from different triple maps. Except from one data source, all of them have several joins.
- **Size:** Most of the data sources are small, only 5 of them have a size bigger than a GB. Together, they have a size of 45.4 GB.

[7] https://github.com/oeg-upm/Mapeathor
[8] https://doi.org/10.5281/zenodo.3552369

Table 3: Characteristics of the declarative mappings: the number of source files, Triple Maps (TM), Predicate-Object Maps (POM), different predicates and objects, joins and size of each dataset.

Database	# Source	# TM	# POM	# Pred.	# Obj.	# Join	Size
ClinicalTrials	1	30	223	89	156	30	1 GB
CTD	8	8	46	10	33	11	3.6 GB
GenAge	2	2	25	19	24	4	337 KB
GenDR	1	2	10	7	10	3	85 KB
HGNC	1	3	48	37	47	5	28 MB
Homologene	1	1	8	7	8	2	13.8 MB
iProClass	1	1	25	18	25	2	33.1 GB
iRefIndex	1	7	40	21	38	11	2.8 GB
LSR	1	2	28	22	27	2	849 KB
NCBIgene	8	12	61	33	51	6	4.4 GB
NDC	2	3	30	20	27	2	75 MB
SIDER	3	3	26	13	20	0	44 MB
Taxonomy	4	4	26	17	23	3	323.9 MB
Wormbase	4	4	30	12	25	6	72 MB

The complexity of the mappings makes it necessary the use of many of the features of the mapping languages. For example, we need to include the extensions developed with the Function Ontology (FnO+RML) [17] to deal with heterogeneous data. This, in addition to the datasets' size and high number of TM, POMs and joins, have stressed the state-of-the-art OBDA tools. Such characteristics make these mappings suitable for testing and improving engines [16].

4.2 Completeness

To test the completeness of the data produced with our approach, we compare the number of instances of different classes from 4 selected datasets obtained from the Bio2RDF SPARQL endpoint[9] and an OBDA engine focused on providing access over tabular open data, Morph-CSV[10]. We use the latter because it exploits the knowledge encoded in RML+FnO and CSVW to enhance data extraction from tabular files [15]. For example, it is able to modify missing values to be treated as NULLs, add the structure that some files lack (headers specially), normalize to 3NF (when there is more than one value in a single field (1NF), or there are several concepts in the same file (2NF)) and treat data formats (e.g. integers, dates). It enables RDF materialisation and query translation; the first option is applied in this use case. The generated data is available online in a Virtuoso SPARQL Endpoint[11].

The results are summarized in Table 4. Only in one class the number of instances is the same. In this case, the dataset has not been updated with more data, that is why the number of instances is equal. In the rest, we obtain more

[9] `http://bio2rdf.org/sparql`
[10] `https://github.com/oeg-upm/morph-csv`
[11] `http://bio2rdf.morph.oeg-upm.net`

Table 4: Comparison of the data completeness. It shows the number of instances belonging to classes from 4 different datasets in the data obtained from the Bio2RDF SPARQL endpoint and our approach, Morph-CSV.

Dataset	Class	Morph-CSV	Bio2RDF SPARQL endpoint
NDC	Package	**251169**	176931
WormBase	Gene	**51255**	50894
Taxonomy	Resource	**2110171**	1329119
Homologene	Homologene-Group	44233	44233

instances with our approach than what is now available on the SPARQL endpoint. This fact points out the necessity to improve the process of accessing the data, with the aim of accessing and providing the updated data more easily.

4.3 Challenges

In the development of this work we have found several issues that show some limitations faced by the proposed approach. They can be grouped into three factors: the heterogeneity of the data, the available features given by the mapping languages and their implementation in state-of-the-art tools:

- The complexity of data makes it necessary to use cleaning and transformation functions (e.g. concatenation, uppercase, regular expressions). The use of these functions enables the automation of the process and avoids the need of having manual data pre-processing. These functions are described declaratively by approaches such as the Function Ontology and CSVW.
- There are some cases where the functions are not enough to exploit data. For example, when there is more than one value in the same field, or the predicate varies in different conditions for the same object. These cases are not taken into account in the specifications of all the mapping languages yet.
- The state-of-the-art OBDA engines have some limitations too. They implement the specifications of one mapping language, which are specialized in dealing with a limited number of data formats. Thus, one engine is not enough to process all the existing data formats. It also complicates the use of the same mapping among different engines when the language that they are able to process is not the same.

Our work shows some current limitations that OBDA engines and mapping languages present when dealing with a real use case. With it we encourage their development in order to enable the access and processing of all kinds of data.

5 Related work

There are more platforms in the area of biomedical knowledge, apart from the Bio2RDF Project, that have worked on transforming their data into Linked Data,

and are accessible by SPARQL endpoints. That is the case of the European Bioinformatics Institute (EMBL-EBI) [18], UniProt [19] and DisGeNet [20]. There is also work done on creating biological-related ontologies. Another important platform is BioPortal [21], which is the major repository for biomedical ontologies and also enables the search of integrated data sources, such as ClinicalTrials[12] and ArrayExpress[13]. Moreover, there are actual examples of the application of OBDA technologies over biomedical data [22,23].

In the context of OBDA, several mapping languages and their corresponding processors have been proposed in the state of the art. The W3C Recommendation R2RML [8] is a declarative mapping language for specifying transformation rules from relational databases to RDF. There are several R2RML processors available such as Ontop [12] and Morph-RDB [11]. Several mapping languages have been proposed to extend R2RML for the purpose of generating RDF from non-relational database sources. Dimou *et al.* [9] proposed RML to generate RDF datasets from (semi)-structured data. There are several RML processors such as: RMLMapper[14], RDFizer[15] and RocketRML [24]. Furthermore, this language has been aligned with the Function Ontology (FnO+RML) to expand its capabilities when dealing with data [17]. There are also non declarative mapping languages, such as Tarql[16] and SPARQL-Generate [25].

A relevant work that is inline with our proposal is the use of declarative mappings to substitute the ad-hoc DBpedia mappings, as reported in [26]. Because of the limitations encountered in the previous methodology, the authors propose the use of declarative mappings (RML+FnO). The implementation shows a better transformation of the data with improved quality, and the enhancement in the maintainability to define transformation rules.

6 Conclusions and future work

This paper is intended to be an alternative approach to the methodology that Bio2RDF follows for RDF generation in order to make it more maintainable. We propose the use of OBDA technologies, as they enable defining declaratively the transformation rules from the source data to the ontology. In comparison to the existing approach, there is now a clear separation between the transformation rules and the engine that executes the rules.

The created mappings can also be useful for another purpose: generating an OBDA testbed. The real data used in this work together with their mappings characteristics may be used as a testbed for OBDA tools, helping them to cope with new unexpected situations that may not be covered by the standard R2RML/RML test cases.

This paper addresses only data sources that are stored in CSVs or relational databases. There are other formats in Bio2RDF that need to be considered as

[12] `https://clinicaltrials.gov/`
[13] `https://www.ebi.ac.uk/arrayexpress/`
[14] `https://github.com/RMLio/rmlmapper-java`
[15] `https://github.com/SDM-TIB/SDM-RDFizer`
[16] `https://github.com/tarql/tarql`

well. Here relies an essential part of the future work: to improve our system so that it can access and deal with additional formats while keeping the essence of OBDA declarative mappings.

Acknowledgements. We would like to acknowledge Luis Pozo for helping in the development of Mapeathor. The work presented in this paper is supported by the Spanish Ministerio de Economía, Industria y Competitividad and EU FEDER funds under the DATOS 4.0: RETOS Y SOLUCIONES - UPM Spanish national project (TIN2016-78011-C4-4-R) and by an FPI grant (BES-2017-082511).

References

1. Zou, D., Ma, L., Yu, J., Zhang, Z.: Biological databases for human research. Genomics, proteomics & bioinformatics **13**(1) (2015) 55–63
2. Toomula, N., Kumar, A., Kumar, D., Bheemidi, V.S.: Biological databases-integration of life science data. J. Comput. Sci. Syst. Biol **4** (2012) 87–92
3. Rigden, D.J., Fernández, X.: The 26th annual Nucleic Acids Research database issue and Molecular Biology Database Collection. Nucleic Acids Research **47**(D1) (2018) D1–D7
4. Belleau, F., Nolin, M.A., Tourigny, N., Rigault, P., Morissette, J.: Bio2RDF: towards a mashup to build bioinformatics knowledge systems. Journal of biomedical informatics **41**(5) (2008) 706–716
5. Dumontier, M., Callahan, A., Cruz-Toledo, J., Ansell, P., Emonet, V., Belleau, F., Droit, A.: Bio2RDF release 3: a larger connected network of linked data for the life sciences. In: Proceedings of the 2014 International Conference on Posters & Demonstrations Track. Volume 1272. (2014) 401–404
6. Poggi, A., Lembo, D., Calvanese, D., De Giacomo, G., Lenzerini, M., Rosati, R.: Linking data to ontologies. In: Journal on data semantics X. Springer (2008) 133–173
7. De Meester, B., Heyvaert, P., Verborgh, R., Dimou, A.: Mapping Languages Analysis of Comparative Characteristics . In: Proceeding of the First International Workshop on Knowledge Graph Building. (2019)
8. Das, S., Sundara, S., Cyganiak, R.: R2RML: RDB to RDF Mapping Language. W3C Recommendation (2012)
9. 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: LDOW. (2014)
10. Heyvaert, P., De Meester, B., Dimou, A., Verborgh, R.: Declarative Rules for Linked Data Generation at Your Fingertips! In: European Semantic Web Conference, Springer (2018) 213–217
11. Priyatna, F., Corcho, O., Sequeda, J.: Formalisation and experiences of R2RML-based SPARQL to SQL query translation using morph. In: Proceedings of the 23rd international conference on World wide web, ACM (2014) 479–490
12. Calvanese, D., Cogrel, B., Komla-Ebri, S., Kontchakov, R., Lanti, D., Rezk, M., Rodriguez-Muro, M., Xiao, G.: Ontop: Answering SPARQL queries over relational databases. Semantic Web **8**(3) (2017) 471–487
13. Iglesias-Molina, A., Chaves-Fraga, D., Priyatna, F., Corcho, O.: Towards the definition of a language-independent mapping template for knowledge graph creation. In: Proceedings of the Third International Workshop on Capturing Scientific Knowledge. (2019)

14. Tennison, J., Kellogg, G., Herman, I.: Model for tabular data and metadata on the web. W3C recommendation. World Wide Web Consortium (W3C) (2015)
15. Corcho, O., Priyatna, F., Chaves-Fraga, D.: Towards a New Generation of Ontology Based Data Access. Semantic Web Journal (2019)
16. Chaves-Fraga, D., Endris, K.M., Iglesias, E., Corcho, O., Vida, M.E.: What are the Parameters that Affect the Construction of a Knowledge Graph? In: OTM Confederated International Conferences "On the Move to Meaningful Internet Systems", Springer (2019)
17. De Meester, B., Maroy, W., Dimou, A., Verborgh, R., Mannens, E.: Declarative data transformations for Linked Data generation: the case of DBpedia. In: European Semantic Web Conference, Springer (2017) 33–48
18. Jupp, S., Malone, J., Bolleman, J., Brandizi, M., Davies, M., Garcia, L., Gaulton, A., Gehant, S., Laibe, C., Redaschi, N., et al.: The EBI RDF platform: linked open data for the life sciences. Bioinformatics **30**(9) (2014) 1338–1339
19. Jain, E., Bairoch, A., Duvaud, S., Phan, I., Redaschi, N., Suzek, B.E., Martin, M.J., McGarvey, P., Gasteiger, E.: Infrastructure for the life sciences: design and implementation of the UniProt website. BMC bioinformatics **10**(1) (2009) 136
20. Queralt-Rosinach, N., Pinero, J., Bravo, À., Sanz, F., Furlong, L.I.: DisGeNET-RDF: harnessing the innovative power of the Semantic Web to explore the genetic basis of diseases. Bioinformatics **32**(14) (2016) 2236–2238
21. Noy, N.F., Shah, N.H., Whetzel, P.L., Dai, B., Dorf, M., Griffith, N., Jonquet, C., Rubin, D.L., Storey, M.A., Chute, C.G., et al.: BioPortal: ontologies and integrated data resources at the click of a mouse. Nucleic acids research **37**(suppl_2) (2009) W170–W173
22. Vidal, M.E., Jozashoori, S.: Semantic Data Integration Techniques for Transforming Big Biomedical Data into Actionable Knowledge. In: 2019 IEEE 32nd International Symposium on Computer-Based Medical Systems (CBMS), IEEE (2019) 563–566
23. Priyatna, F., Alonso-Calvo, R., Paraiso-Medina, S., Corcho, O.: Querying clinical data in hl7 rim based relational model with morph-rdb. Journal of biomedical semantics **8**(1) (2017) 49
24. Şimşek, U., Kärle, E., Fensel, D.: RocketRML-A NodeJS implementation of a use-case specific RML mapper. In: Proceeding of the First International Workshop on Knowledge Graph Building. (2019)
25. Lefrançois, M., Zimmermann, A., Bakerally, N.: A SPARQL extension for generating RDF from heterogeneous formats. In: European Semantic Web Conference, Springer (2017) 35–50
26. Maroy, W., Dimou, A., Kontokostas, D., De Meester, B., Verborgh, R., Lehmann, J., Mannens, E., Hellmann, S.: Sustainable linked data generation: The case of dbpedia. In: International Semantic Web Conference, Springer (2017) 297–313