Mapping Representation based on Meta-data and SPIN for Localization Workflows

Alan Meehan, Rob Brennan, Dave Lewis, Declan O'Sullivan

CNGL Centre for Global Intelligent Content, Knowledge and Data Engineering Group, School of Computer Science and Statistics, Trinity College Dublin, Ireland
{meehanal,rob.brennan,dave.lewis,declan.osullivan}@scss.tcd.ie

Abstract. The localization industry currently deploys language translation workflows based on heterogeneous tool-chains. Standardized tool interchange formats such as XLIFF (XML Localization Interchange File Format) have had some impact on enabling more agile translation workflows. However the rise of new tools based on machine translation technology and the growing demand for enterprise linked data applications has created new interoperability challenges as workflows need to encompass a broader range of tools. In this paper we present an approach of representing mappings between RDF-based representations of multilingual content and meta-data. To represent the mappings, we use a combination of SPARQL Inferencing Notation (SPIN) and meta-data. Our approach allows the mapping representation to be published as Linked Data. In contrast to other frameworks such as R2R, the mappings are executed via a standard SPARQL processor. The objective is to provide a more agile approach to translation workflows and greater interoperability between software tools by leveraging the ongoing innovation in the Multilingual Web field. Our use case is a Language Technology retraining workflow where publishing mappings leads to new opportunities for interoperability and end-to-end tool-chain analytics. We present the results from an initial experiment which compared our approach of executing and representing mappings to that of a similar approach - the R2R Framework.

Keywords: Multilingual Web, Semantic Mapping, Interoperability

1 Introduction

The localization industry is historically built on fragile cross-enterprise tool-chains with strong interoperability requirements. Ongoing research and innovation by the Multilingual Semantic Web and Linked Data communities has led to promising new technologies that can simultaneously span the language and interoperability barriers. However switching to an enterprise Linked Data model is not a straight forward task. Data needs to be transformed from its original format into a RDF-based representation and multiple domain or tool-specific vocabularies are often employed within the RDF. This increases the importance of mapping technology [1] to enable flexible end-to-end tool-chains. Where such tool-chains are in place, there is considerable com-

mercial advantage to enabling end-to-end analytics that can monitor content flows through the tools and the impact of mapping steps.

This paper focuses on the problem of making such mapping steps visible within a localization tool-chain, exposing the mappings in a way that facilities discovery, lifecycle management and the recording of mapping meta-data such as the mapping provenance. These mappings must be executable in the sense that it is desirable to have a framework that takes the mapping representation and can apply it as needed to instance data. By avoiding proprietary technologies in the execution step it is hoped that a wider range of tool vendors can be used to lower costs and simplify integration across multiple enterprises in a localization value chain.

This leads to the following two research questions that are investigated in this paper. How can mappings be expressed as Linked Data to facilitate discovery and the recording of mapping meta-data? To what extent can standard SPARQL endpoints act as an execution engine for these mappings?

We represent the executable RDF-to-RDF mappings as SPARQL[1] construct queries that can be executed on any standard SPARQL endpoint. To support mapping publication, discovery and meta-data annotation we represent the mappings as SPARQL Inference Notation (SPIN) [9] Linked Data. We aim to exploit this capability by also publishing associated mapping meta-data that will lead to new techniques for mapping lifecycle management and SPARQL-based mapping quality analytics.

Although a work in progress, the contribution of this paper is an evaluation of the relative expressivity of representing executable mappings as SPARQL construct queries compared to the mapping language of the R2R Framework [8] based on a set of test mappings previously published by the R2R team. In addition the viability of using SPIN to publish the mappings as Linked Data is evaluated by transforming the test mapping set into SPIN-based RDF representations.

The remainder of the paper is as follows: Section 2 presents a use case to illustrate where our approach of mapping representation would be useful; Section 3 presents the requirements of the mapping representation; Section 4 covers related work in the area of semantic mapping and the publication of mappings; Section 5 presents the evaluation of our approach of representing and executing mappings against the R2R Framework; we finish with conclusions and future work in Section 6.

2 Language Technology Retraining Workflow Use Case

This section describes a use case centered on the localization industry's process of providing translated content. This was chosen as an exemplar of complex real world workflows that the authors were familiar with. We focus on a Language Technology (LT) retraining workflow[2], with the goal to provide a means for translated content to be retrieved and used to retrain multiple machine translation tools.

---

[1] http://www.w3.org/TR/sparql11-query/

[2] Currently an ongoing development at CNGL Centre for Global Intelligent Content: http://www.cngl.ie

Figure 1 illustrates the process whereby a piece of HTML source content undergoes a series of processing steps, acted on by specific tools for translation, quality assessment and post editing. An XML Localization Interchange File Format (XLIFF)[3] file is used to record the processing that the content has undergone at each step. At the end of the content flow, a custom tool, using the Extensible Stylesheet Language Transformation (XSLT)[4] language, is used to map the data from the XLIFF file into RDF using the Global Intelligent Content semantic model (GLOBIC)[5] vocabulary and stored in a triple store. This RDF data represents details such as the *source* and *target* of text content that underwent a Machine Translation (MT) process, which tool carried out the MT process, *post edits* and *quality estimates* associated with translated content. By building up data in the triple store, it becomes a rich source of MT training data. A high quality training data-set is important for MT applications in order to gain benefits during training phases.

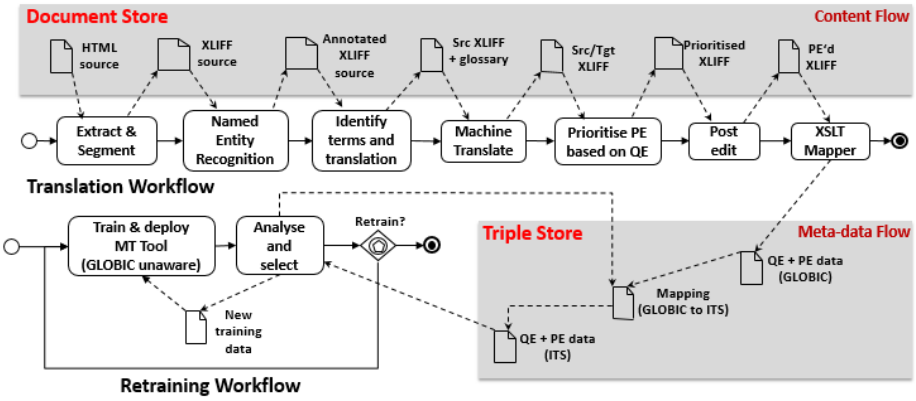

Figure 1. Language Technology Retraining Workflow

The retraining aspect of the workflow involves retrieving suitable content to be re-fed into the MT statistical machine learning tool. This is achieved by querying the triple store for translated content with a quality estimate over a certain threshold value, which is easily achieved using SPARQL queries.

Heterogeneous tools looking to utilize this training data naturally need to have the data in the triple store mapped to a schema they recognize. In Figure 1, the *MT tool* is GLOBIC unaware, it is designed to use the content represented by the Internationalization Tag Set[6] (ITS) vocabulary. Thus the *Quality Estimate (QE)* and *Post Edited (PE) data* that is represented in GLOBIC must be mapped to an ITS representation for

---

3 http://docs.oasis-open.org/xliff/xliff-core/xliff-core.html

4 http://www.w3.org/TR/xslt

5 The GLOBIC semantic model is an ongoing development at CNGL Centre for Global Intelligent Content. Its purpose in this use case is to enable greater interoperability and analytics within the workflow: http://www.scss.tcd.ie/~meehanal/gic.ttl

6 http://www.w3.org/TR/its20/

the *MT tool* to use it. As will be seen in Section 3, our approach to representing such mappings allows them to be published alongside the other data in the triple store. This allows the mappings to be discovered by users/tools through SPARQL queries and executed by the SPARQL processor itself when transformed back to SPARQL syntax from SPIN. Transforming from SPARQL syntax to SPIN or vice-versa can be can be done using the SPIN RDF Converter[7], which can be used as a free web service.

3 Mapping Representation Requirements and Design

This section describes the requirements for the mapping representation and an example of how a mapping is represented under our approach.
The requirements of the mapping representation are as follows:

1. A mapping entity must be expressed as RDF, with a unique URI, allowing the mapping to be publishable on the web and discoverable via SPARQL queries.
2. The executable mapping statement must be a SPARQL query that is executable by a SPARQL processor.
3. The executable mapping statement must be expressed as RDF and must have a unique URI, allowing the statement to be queried by SPARQL and linked to a mapping entity.
4. A mapping entity is to be modeled with associated meta-data expressed as RDF, providing additional data on the mapping which can be queried via SPARQL.

To fulfil *requirement 1*, a mapping entity should be given a meaningful, unique name and modeled as an instance of the *Mapping* class from the GLOBIC vocabulary. For *requirement 2*, the executable mapping statement should be devised as a SPARQL construct query. For *requirement 3*, the SPARQL construct query should be converted to SPIN in order to be expressed as RDF and given a unique, meaningful name. The *hasRepresentation* property from the GLOBIC vocabulary should be used to link the SPIN representation of the SPARQL construct query to the mapping entity. For *requirement 4*, the associated meta-data for a mapping entity should be modeled using the following properties. The *wasCreatedBy*, the *mapDescription* and the *version* properties from the GLOBIC vocabulary and the *generatedAtTime* and *wasRevisionOf* properties from the W3C PROV[8] vocabulary are used.

Below is an example of a mapping representation that concerns the mapping of a MT quality score from the GLOBIC vocabulary to the ITS vocabulary. The SPIN representation of the SPARQL construct query is as follows:

```
01: @PREFIX gic: <http://www.scss.tcd.ie/~meehanal/gic#>.
02: @PREFIX itsrdf: <http://www.w3.org/2005/11/its/rdf#>.
03: @PREFIX sp: <http://spinrdf.org/sp#>.
04: @PREFIX ex: <http://www.example.org/example#>.
```

[7] http://spinservices.org/spinrdfconverter.html

[8] The PROV ontology contains classes and properties that can be used to model provenance data about an entity, agent or activity: http://www.w3.org/TR/prov-o/

```
05: ex:globic_to_its_mtScore_sp_2 a sp:Construct;
06:  sp:templates ([ sp:object _:b1;
07:                  sp:predicate itsrdf:mtConfidence;
08:                  sp:subject _:b2 ]);
09:  sp:where ([ sp:object _:b1;
10:              sp:predicate gic:qualityAssessment;
11:              sp:subject _:b2 ]).
12: _:b2 sp:varName "s"^^xsd:string.
13: _:b1 sp:varName "val"^^xsd:string.
```

The mapping entity plus associated meta-data is as follows:

```
14: ex:globic_to_its_mtScore_map_2 a gic:Mapping;
15:  gic:hasRepresentation ex:globic_to_its_mtScore_sp_2;
16:  gic:wasCreatedBy ex:person_1;
17:  prov:generatedAtTime "2014-01-01"^^xsd:date;
18:  gic:mapDescription "Used to map X to Y etc…";
19:  gic:version "1.1"^^xsd:float;
20:  prov:wasRevisionOf ex:globic_to_its_mtScore_map_1.
```

Examining the example above, *line 14* contains the name of the mapping entity. *Line 15* links the mapping to the SPIN representation of the SPARQL construct query on *line 05*. *Line 16* indicates what person/application is responsible for creating the mapping. *Line 17* indicates when the mapping was created. *Line 18* provides a human readable description of what the mapping does. *Line 19* indicates the current version of the mapping. *Line 20* provides a link to the previous version of the mapping.

4 Related Work

There is a rich body of research in semantic mapping undertaken by the semantic web community [1]. A wide variety of approaches have been adopted to tackle the mapping challenge, from rule-based representations [2], to axiomatic representations [3], to SPARQL query representations [4-5].

Keeney et al. [6] evaluated these three mapping approaches and found that the SPARQL query approach in general excels in terms of execution time and efficient use of computational resources. Although there are some particular circumstances where there are downsides to this approach. Keeney et al. conclude that for tasks where applications wish to map and use relatively small, specific data, the SPARQL approach would be ideal.

Little research has been undertaken into publishing mappings in order for them to be discovered and re-used. Thomas et al. [7] propose that the lack of ontology and mapping meta-data impedes the task of discovering relevant mappings between ontologies. They propose a thirty-three element mapping meta-data ontology, *OM²R*, based on the mapping lifecycle. We plan to build on this work to extend the scope of

the meta-data collected on mappings, however in our approach the W3C PROV vocabulary will be used as the basis of lifecycle fields such as creation date.

A notable approach to publishing mappings however is the *R2R Framework* [8], which is a framework for executing mappings between RDF ontologies. The R2R framework has its own language called the *R2R mapping language*[9] for publishing mappings on the web. Similar to our approach of representing mappings, the R2R mappings are instances of a *Mapping* class, from the R2R mapping language, not the GLOBIC vocabulary. The mappings are modeled with meta-data and use the properties *sourcePattern* and *targetPattern* to represent the triple patterns which are executed by the R2R Framework. Our approach differs in that mapping instances are linked to a SPIN representation of a SPARQL construct query, which is ultimately executed by a SPARQL processor.

The SPARQL Inference Notation (SPIN) is a set of vocabularies that are used to represent business rules and constraints via SPARQL queries [9]. Tools that implement SPIN, such as TopBraid Composer[10] have been used in a wide range of applications [10-13]. Such tools also allow custom functions (which may not appear in the SPARQL specification) to be declared and executed, which make SPIN tools versatile at establishing data constraints and even data mapping [14].

5 Evaluation

This section describes two initial experiments of a series of planned experiments; the two here were carried out in order to evaluate our approach of representing mappings. The first experiment compared SPARQL's mapping capabilities with that of the R2R Framework. The second experiment involved testing the expressiveness of SPIN with regard to expressing SPARQL construct queries as RDF.

5.1 Experiment 1: Comparing SPARQL's mapping capabilities to the R2R Framework

Hypothesis: It is possible to represent all of the 70[11] R2R Framework test mappings as SPARQL construct queries and the execution of these SPARQL construct queries will produce identical results as the R2R Framework mapping results.
Method: First, a data-set[12] was collected. The creators of the R2R Framework devised 72 test mappings[13] between DBpedia and 11 other data-sources to test their framework. We collected instance data, related by the test mappings, via SPARQL endpoints and data dump files. Then the test mappings were executed against the data-set using the R2R Framework. This resulted in 70 output files consisting of new-

9 http://wifo5-03.informatik.uni-mannheim.de/bizer/r2r/spec/

10 http://www.topquadrant.com/tools/IDE-topbraid-composer-maestro-edition/

11 Note that only 70 of the 72 test mappings were carried out as data from BookMashup could not be obtained

12 Data from this experiment can be found at: http://www.scss.tcd.ie/~meehanal/Experiment1/

13 http://wifo5-03.informatik.uni-mannheim.de/bizer/r2r/examples/DBpediaToX.ttl

ly inferred triples. Next the data-set was loaded into an Apache Jena triple-store with Fuseki SPARQL server[14]. The 70 test mappings were represented as SPARQL construct queries and executed against the data in the triple-store. This resulted in 70 output files consisting of newly inferred triples. Lastly, the output files from the R2R Framework were compared with the respective output files derived from the SPARQL construct queries.

Results: It was found that the SPARQL construct queries created identical outputs as the R2R Framework for all of the 70 test mappings, which indicates that the SPARQL construct queries were accurate representations of the R2R Framework test mappings.

Discussion: The results are promising in showing that SPARQL is as capable as the R2R Framework for executing mappings on RDF data sets. The SPARQL 1.1 specification standardized a number of functions, such as string manipulation functions, which allow for more complex mappings to be carried out. Prior to SPARQL 1.1, the R2R Framework test mapping that involved string manipulation would not be possible using the SPARQL 1.0 specification, allowing the R2R Framework to represent more complex mappings than SPARQL.

5.2 Experiment 2: Testing SPIN's Expressivity

Hypothesis: It is possible to express all 70 of the SPARQL construct queries (from Experiment 1) as RDF via SPIN.

Method: This test used the SPIN RDF Converter and TopBraid Composer 4.4.0 (free edition) to transform the 70 SPARQL construct queries to SPIN syntax. An error is produced by the converter and composer if it cannot represent a SPARQL query.

Results: It was found that SPIN could represent all 70 of the SPARQL construct queries. The construct queries were categorized according to Scharffe's correspondences patterns [15]. All 70 fell into 3 patterns: *Equivalent Class*, *Equivalent Relation* and *Property Value Transformation* as illustrated in Figure 2 (some queries span across two patterns).

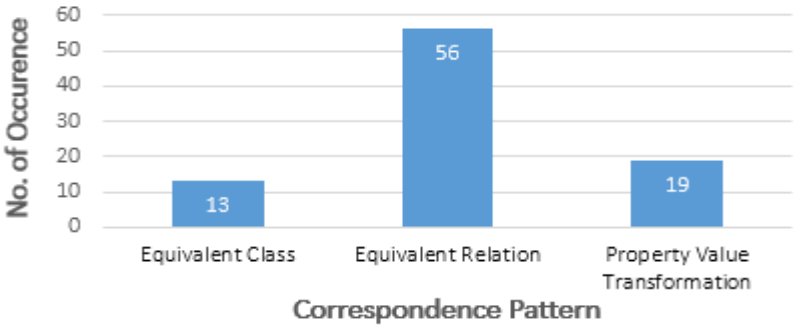

Figure 2. R2R Test Mappings broken down by Correspondence Pattern Type

[14] http://jena.apache.org/documentation/serving_data/

Discussion: Initially it was found that the SPIN RDF Converter could only represent 64 of the 70 SPARQL construct queries. Specifically the SPARQL 1.1 standardized functions *REPLACE*, *STRBEFORE* and *STRAFTER* could not be represented. The creators of SPIN were contacted and the SPIN RDF Converter is using an outdated version of SPIN and will be updated in the near future. However, TopBraid Composer 4.4.0 uses the latest version of SPIN and this was used to represent the 6 SPARQL construct queries that the SPIN RDF Converter could not.

6 Conclusions and Future Work

In this paper we have proposed a semantic mapping representation, between RDF data sources, that allows the mapping to be published, discovered and executed. The goal of the mapping representation is to provide an approach towards greater interoperability between heterogeneous tools operating within a localization tool-chain.

We represent the executable mapping statement as a SPARQL construct query which is expressed as RDF via SPIN. Mappings are modelled with meta-data, also expressed as RDF using the GLOBIC and W3C PROV vocabularies. All aspects of a mapping are published in a triple store alongside other data, where they can be discovered, queried and ultimately executed by a SPARQL processor.

We have shown that SPARQL construct queries are just as expressive as the R2R Mapping Language for representing a wide variety of mappings and that these queries can be represent in RDF via the SPIN syntax.

Future work will investigate a model of SPARQL-based mapping quality analytics and lifecycle management where all aspects of a mapping (meta-data and SPIN representation) can be queried and even updated/deleted using SPARQL queries.

Acknowledgements. This research is supported by the Science Foundation Ireland (Grant 12/CE/I2267) as part of CNGL Centre for Global Intelligent Content (www.cngl.ie) at Trinity College Dublin.

References

1. Shvaiko, P. and Euzenat, J. "Ontology matching: state of the art and future challenges." *Knowledge and Data Engineering, IEEE Transactions on* 25, no. 1, 158-176, 2013.
2. Arch-int, N. and Arch-int, S. "Semantic Ontology Mapping for Interoperability of Learning Resource Systems using a rule-based reasoning approach." *Expert Systems with Applications* 40, no. 18, pp. 7428-7443, 2013.
3. Kumar, S. and Harding, J. A. "Ontology mapping using description logic and bridging axioms." *Computers in Industry* 64, no. 1, pp. 19-28, 2013.
4. Euzenat, J., Polleres, A. and Scharffe, F. "Processing ontology alignments with SPARQL." In *Complex, Intelligent and Software Intensive Systems, 2008. CISIS 2008. International Conference on*, pp. 913-917. IEEE, 2008.

5. Rivero, C. R., Hernández, I., Ruiz, D., and Corchuelo, R. "Generating SPARQL executable mappings to integrate ontologies." In *Conceptual Modeling–ER 2011*, pp. 118-131. Springer Berlin Heidelberg, 2011.
6. Keeney, J., Boran, A., Bedini, I., Matheus, C. J. and Patel-Schneider, P. F. "Approaches to Relating and Integrating Semantic Data from Heterogeneous Sources." In *Proceedings of the 2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology-Volume 01*, pp. 170-177. IEEE Computer Society, 2011.
7. Thomas, H., Brennan, R., and O'Sullivan, D. "Using the OM 2 R Meta-Data Model for Ontology Mapping Reuse for the Ontology Alignment Challenge–a Case Study." In *Proceedings of the 7th Intl. Workshop on Ontology Matching*, vol. 946. 2012.
8. Bizer, C. and Schultz, A. "The R2R Framework: Publishing and Discovering Mappings on the Web." In *1st International Workshop on Consuming Linked Data (COLD2010)*, Shanghai, China, November, 2010.
9. Knublauch, H. SPIN SPARQL Inferencing Notation. *http://spinrdf.org/* (accessed 06 03, 2014).
10. Fürber, C. and Hepp, M. "Using SPARQL and SPIN for data quality management on the Semantic Web." In *Business Information Systems*, pp. 35-46. Springer Berlin Heidelberg, 2010.
11. Spohr, D., Cimiano, P., McCrae, J. and O'Riain, S. "Using spin to formalise accounting regulations on the semantic web." In *International Workshop on Finance and Economics on the Semantic Web (FEOSW 2012)*, pp. 1-15. 2012.
12. Lefrançois, M. and Gandon, F. "ULiS: An Expert System on Linguistics to Support Multilingual Management of Interlingual Semantic Web Knowledge bases." In *MSW-Proc. 2nd Workshop on the Multilingual Semantic Web, collocated with ISWC-2011*, vol. 775, pp. 50-61. 2011.
13. Andreasik, J., Ciebiera, A. and Umpirowicz, A. "ControlSem–distributed decision support system based on semantic web technologies for the analysis of the medical procedures." In *Human System Interactions (HSI), 2010 3rd Conference on*. IEEE, 2010.
14. Kovalenko, O., Debruyne, C., Serral, E. and Biffl, S. "Evaluation of Technologies for Mapping Representation in Ontologies." In *On the Move to Meaningful Internet Systems: OTM 2013 Conferences*, pp. 564-571. Springer Berlin Heidelberg, 2013.
15. Scharffe, F. "Correspondence Patterns Representation." PhD thesis, University of Innsbruck, 2009.