Towards Educational Metadata Interoperability on Semantic Web

André Behr[1], **Tiago Thompsen Primo**[2], **Rosa Vicari**[1]

[1]Informatics Institute – Federal University of Rio Grande do Sul
Postal Code 15.064 – 91.501-970 – Porto Alegre – RS – Brazil

[2]SAMSUNG Research Institute – Campinas, SP – Brazil

arbehr@inf.ufrgs.br, tiago.t@samsung.com, rosa@inf.ufrgs.br

Abstract. *Metadata is broadly used to describe learning objects and its standards have been developed to improve interoperability. With the current Web extension by the Semantic Web, learning object metadata has been migrating from XML-based metadata formats to semantic metadata formats. This work explores XSLT to provide a solution for this transition of descriptions to educational metadata. The approach consists of retrieve, convert, and store metadata in Semantic Web context. In this work, we converted OBAA metadata in XML to OWL format. The OWL format improved the description semantics and allowed reasoner inference. The converted metadata can be managed by different applications and systems. Along with that, we proposed a strategy to OWL ontology alignment.*

1. Introduction

Metadata is used to describe data on the Web to support and promote sharing and reuse [Tani et al. 2013]. Metadata standards for educational resources have been introduced as an effort to describe information in a defined scope. Dublin Core (DC) [DCMI 2012] and IEEE-LOM [LTSC 2002] are the wide-spread initiatives for the educational context, precisely to describe Learning Objects (LOs) [Palavitsinis et al. 2014].

The Semantic Web extends the traditional World Wide Web. Its information is structure and has the proper semantics to be understandable by machines. Also, the information is interlinked, building relationships among concepts. Therefore, the machines would be able to conduct automated reasoning [Berners-Lee et al. 2001].

The interlinked information is made through RDF triples. The triples could link anything on the Web. Each triple is composed of three parts: subject, predicate, and object. The subject is a Uniform Resource Identifier (URI) to describe the resource as a unique concept. The predicate correlates the subject and object. Last, the object could be a URI or a literal value. The adoption of this and other principles leads the Linked Data initiative [Heath and Bizer 2011].

Nowadays, most of the systems expose their metadata in eXtensible Markup Language (XML) format. The metadata in these systems can be retrieved by OAI-PMH[1] protocol. However, Semantic Web languages are richer because they use formal knowledge representation in the form of ontologies, for example as Resource Description

[1]http://www.openarchives.org/pmh

DOI: 10.5753/cbie.sbie.2016.1026

Framework (RDF-S) and Web Ontology Language (OWL). The ontology-based systems can harvest the metadata following links or using SPARQL[2] queries [Sicilia 2014].

"An ontology is a formal, explicit specification of a shared conceptualization" [Guarino et al. 2009]. In other words, the ontologies are a consensual description of knowledge (meaning of concepts and relations) aimed to be without ambiguity and machine processing. Ontologies can describe a hierarchy of concepts and properties in a context domain. Its axioms express relationships among these concepts. The axioms make ontology more expressive by allowing the use of inference mechanisms.

However, it is expected that more than one ontology can describe a domain. Each organization will produce their own ontology. As a result, the Semantic Web will be composed of a large variety of domain ontologies, highly contextualized, developed locally by software engineers and not by ontology specialists [Hendler 2001].

The data integration is pointed as one of the major Semantic Web drivers. To cope with that, languages have been used for encoding semantics and support data integration and interoperability. The Extensible Stylesheet Language (XSLT) is considered a bootstrap to generate Semantic Web content. Specially XML from back-end databases [Shadbolt et al. 2006]. Actually, both companies and the academy require connected databases, highly shareable, that allows interoperability and may deal with the accumulation of knowledge (i.e. new connected data) available on the Web [Isotani and Bittencourt 2015].

The challenge of this work is to transpose legacy metadata in XML format to OWL, reusing the original metadata files and providing a semantic and connected data description. The method is based on XSLT joined with a consolidated metadata ontology. In addition, OWL to OWL conversions were done with an alignment between ontologies. Learning objects described with OBAA metadata standard were explored as a study case. The converted metadata was stored in a Triple Store.

The paper follows presenting the OBAA metadata standard in Section 2. Related works are discussed in Section 3. The mapping approach is depicted in Section 4. Section 5 brings conclusion and future work.

2. OBAA Metadata Standard

The Brazilian metadata standard OBAA was designed to comply with educational and technical aspects of Brazil. It is an extension of the IEEE-LOM metadata standard aiming to be interoperable with Web, Mobile, and Digital TV [Vicari et al. 2010]. This wide scope of coverage allows describing the most diverse kinds of learning objects. For example, describing learning objects with segmented videos, or people with some sort of disability. Allowing better personalization and filtering options for AI systems.

Besides, the OBAA standard has a strong synergy with the Semantic Web principles and technologies. For instance, the Technical metadata group was reused and extended from IEEE-LOM to fulfill Brazilian aspects. It was also extended to define metadata, such as services, ontology, content languages, and interaction protocols.

Nowadays there are two ontologies that describe the OBAA metadata standard.

[2] `http://www.w3.org/TR/rdf-sparql-query/`

First [Gluz and Vicari 2011], proposes an OWL-DL ontology that furnishes a basic vocabulary to describe the metadata. The second proposal [Behr et al. 2012] explores reasoning capabilities and it is OWL 2 compatible, the ontology provides new axioms to express metadata cardinalities, conditional values, and classification through application profiles.

3. Related Work

The state of the art of using XSLT to convert XML to OWL format is distinguished by two main approaches [Hacherouf et al. 2015]. The first one generates the OWL ontology from the XML Schema. The second creates the OWL ontology separately from XML instances. Furthermore, there are different approaches to lead with this conversion.

A set of patterns can be used to support the mapping of an XML Schema to an OWL ontology. The constructs mapping to patterns aims to facilitate the ontology creation process [Bedini et al. 2011]. Another alternative would be to extend OWL. XMLMaster [O'Connor and Das 2011] is an OWL-centric mapping language, extending Machester OWL Syntax, that supports XML documents with or without XML Schema.

Automatic conversions are commonly employed to convert XML to OWL. For example, a method to create the OWL ontology automatically from multiple XML data sources [Yahia et al. 2012]. This method is based on XML Schema to build the ontology, but if the schema does not exist, it can be automatically generated from the XML document. The method can be divided into four steps: (i) Trang API[3] generates the XML Schema through XML document; (ii) XML Schema Object Model (XSOM)[4] analyses the XML Schema; (iii) the output of XSOM is used as Java Universal Network/Graph framework (JUNG)[5] input; and (iv) Jena API[6] uses XSG as input to generate the OWL entities.

Other completely automated approach is EXCO [Lacoste et al. 2011]. The framework is able to handle internal references in XML Schemas, both to element and type. The mapping is able to process multiple XML instances. EXCO has two independent parts: (i) the model extractor that generates the OWL model from XML Schema Definition (XSD) and (ii) the instance generator that produces OWL from XML and the XSD. These two steps can be executed in parallel.

There are also semi-automatic conversion proposals. An example is a four-step approach that manages three healthcare datasets [Liang et al. 2015] it is divided into four steps: (i) prepares the metadata descriptions; (ii) converts semantic meanings back into the values recorded according to the corresponding description; (iii) is responsible for getting all the metadata and transpose it to classes in OWL; and (iv) transforms the source data into OWL. The authors also explore concepts alignments with existing ontologies, such as classes and individuals.

This work proposes a semi-automatic conversion for XML to OWL metadata. Our proposal differs from others by reusing a reasoning oriented OWL 2 ontology

[3]https://code.google.com/archive/p/jing-trang/
[4]https://xsom.java.net/
[5]http://jung.sourceforge.net
[6]https://jena.apache.org/

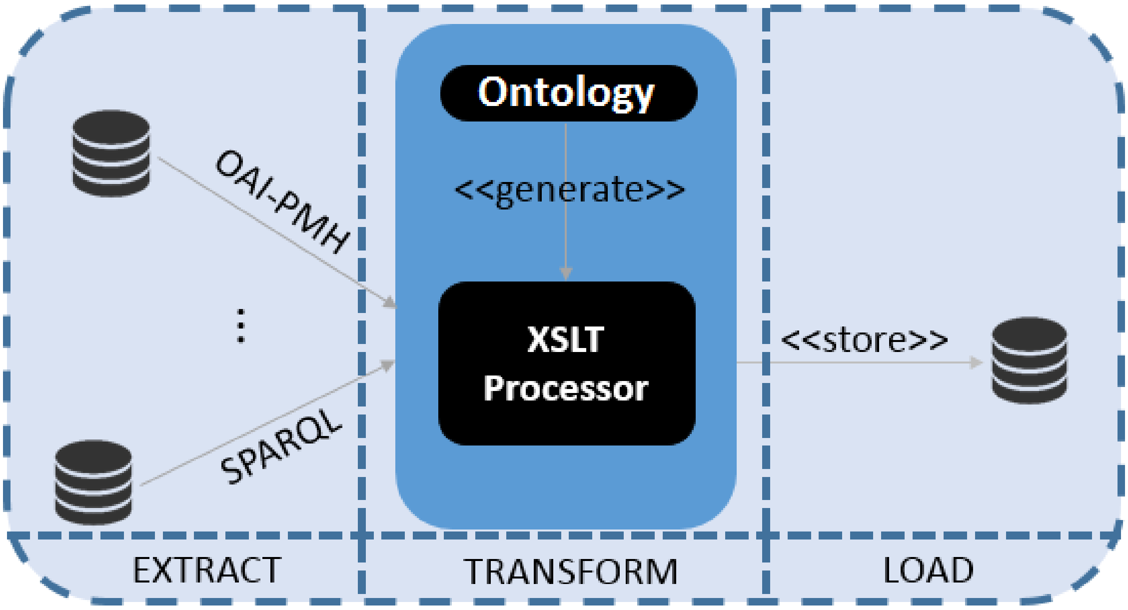


Figure 1. Extract-Transform-Load approach to provide metadata interoperability in Semantic Web

[Behr et al. 2012]. We used this approach to promote ontology reuse (one of Semantic Web main objectives) and to furnish a richer output ontology description of the data. In addition, this work deals with the alignment of OWL ontologies.

4. Methodology

In our approach, we chose to reuse an OWL ontology and the XSLT file is responsible for bridging the metadata in an ontology description. We also agree with authors that defend that having the ontology to produce the XSLT gives more semantic and expressivity to the XML metadata. Automatic creations can lead to different ontologies (i.e. with divergent URI concepts) and the Semantic Web aims reuse. Furthermore, the ontology may be aligned with other ontologies and expand its knowledge representation. Through this, conversions among OWL formats also can be done.

The methodology is divided into three phases: (i) retrieve the metadata from heterogeneous repositories, (ii) convert the metadata in an ontology description, and (iii) store the metadata in a semantic repository. This main idea is represented in Figure 1 and its phases will be explained in the following sections.

4.1. Retrieving LO Metadata

In the first phase, the educational metadata would be retrieved from a repository. The metadata can be discovered with different technologies, such as OAI-PMH, REST, SPARQL, among others. In the case of XML retrieved from databases, the metadata input could be in a single record or a list of records. For Sematic Web repositories, the input could be an ontology or a list of ontology URIs (output of the SPARQL query in Figure 2, for example).

```
SELECT *
WHERE{
?s ?p ?o .
FILTER((STR(?o) = "http://www.w3.org/2002/07/owl#Ontology")).
}
```

Figure 2. SPARQL query to retrieve a list of ontology URIs.

This work captured the metadata from two OBAA repositories[7][8]. In first, we retrieved sixteen learning object metadata in XML form and transposed them to OWL. For the second, we experimented the previous conversion of approximately a thousand of learning object metadata to OWL [Ribeiro and Primo 2013] for an OWL to OWL transformation. Besides, Dublin Core metadata from BIOE[9] was transposed to OBAA.

4.2. Processing LO Metadata

Before the XSLT Processor converts the metadata, the data is normalized. For example, inserting the correspondent XSLT in the XML or OWL metadata description and for OWL conversion transform the OWL file to RDF/XML format to easy triple access. The processing could be between different formats of the Semantic Web stack. For example XML to OWL, RDF to OWL, and OWL to OWL. It is also expected to translate mappings between metadata standards, as example Dublin Core to IEEE-LOM.

A defined OBAA ontology[10] gives the semantics to the XSLT process the metadata. The process can be summarized in the following steps: (i) import of the OBAA ontology domain; (ii) individual creation with unique URIs; and (iii) assignment of object properties and data properties. All these concepts are manipulated with XPath[11] expressions.

XML to OWL

For XML to OWL, the processing is done by individually parsing the metadata. The selection is done through XPath and then create an OWL individual, assigning a data property and/or an object property. Figure 3 exemplifies (a) LO XML code snippet input for the LifeCycle IEEE-LOM group and (b) its triples output. First, the elements LifeCycle and Contribute generate individuals (on the left side of the triple) and an object property (*hasLifeCycle.Contribute*) relate them. After, the data properties are assigned.

OWL to OWL

The OWL ontology alignment is done by equivalences. For example, Figure 4 depicts property alignments between ontologies to version and status metadata. Through this, the

[7] `http://cognix-repo.inf.ufrgs.br/`
[8] `http://repositorio.portalobaa.org/`
[9] `http://objetoseducacionais2.mec.gov.br/`
[10] `http://gia.inf.ufrgs.br/ontologies/OBAA.owl`
[11] `https://www.w3.org/TR/xpath/`

```xml
<obaa>
    <lifecycle>
        <version>1.0</version>
        <status>final</status>
        <contribute>
            <role>author</role>
            <entity>Silva, Vasco Sérgio
                Correia Freitas </entity>
            <date>2009-08-31T19:05:41</date>
        </contribute>
    </lifecycle>
</obaa>
```

(a)

AlkalineFuelCell version "1.0"
AlkalineFuelCell status "final"
AlkalineFuelCell hasLifeCylce.Contribute AlkalineFuelCellLifeCycleContribute
AlkalineFuelCellLifeCycleContribute lifeCycle.Contribute.Role "author"
AlkalineFuelCellLifeCycleContribute lifeCycle.Contribute.Entity "Silva, Vasco Sérgio Correia Freitas"
AlkalineFuelCellLifeCycleContribute lifeCycle.Contribute.Date "2009-08-37T19:05:41"

(b)

Figure 3. Alkaline Fuel Cell LifeCycle [Silva 2011] XML transposed to triples.

```xml
<EquivalentDataProperties>
    <DataProperty IRI="http://gia.inf.ufrgs.br/ontologies/LOM.owl#status"/>
    <DataProperty IRI="http://obaa.unisinos.br/obaa22.owl#itsStatusIs"/>
</EquivalentDataProperties>
<EquivalentDataProperties>
    <DataProperty IRI="http://gia.inf.ufrgs.br/ontologies/LOM.owl#version"/>
    <DataProperty IRI="http://obaa.unisinos.br/obaa22.owl#itsVersionIs"/>
</EquivalentDataProperties>
```

Figure 4. Data property equivalences for version and status metadata.

XSLT is generated to the conversion. The triples are selected also by XPath and each one is changed for its equivalent. In this work, considering that the OWL ontologies shared the knowledge representation methodology, this approach was feasible.

Figure 5 exemplifies part of this transformation: Ontology A triples are the input to generate triples for Ontology B output. The transformation replaces individual names, data properties, and object properties with its equivalents in other ontology.

Dublin Core to OBAA

Dublin Core has a direct mapping to IEEE-LOM, depicted by Table 1. As OBAA extends IEEE-LOM, the fifteen DC elements could be also transposed to OBAA. Figure 6 exemplifies the individuals' relationships generated for this mapping in the Brasilia Learning Object metadata[12]. Some of the ontology data properties and object properties triples are also depicted in Figure 7.

4.3. Storing LO Metadata

A semantic repository provides a graph storage for data. This allows that a concept may have *n* relationships. The semantic repositories use ontologies as semantic schemata. This makes automated reasoning possible because the relationships between the concepts are built into the ontology.

GraphDB[13] is considered the world's leading RDF Triple-Store. It is built on OWL and it uses ontologies that allow the repository to automatically reason about the

[12] http://objetoseducacionais2.mec.gov.br/handle/mec/430?show=full
[13] http://ontotext.com/products/graphdb/

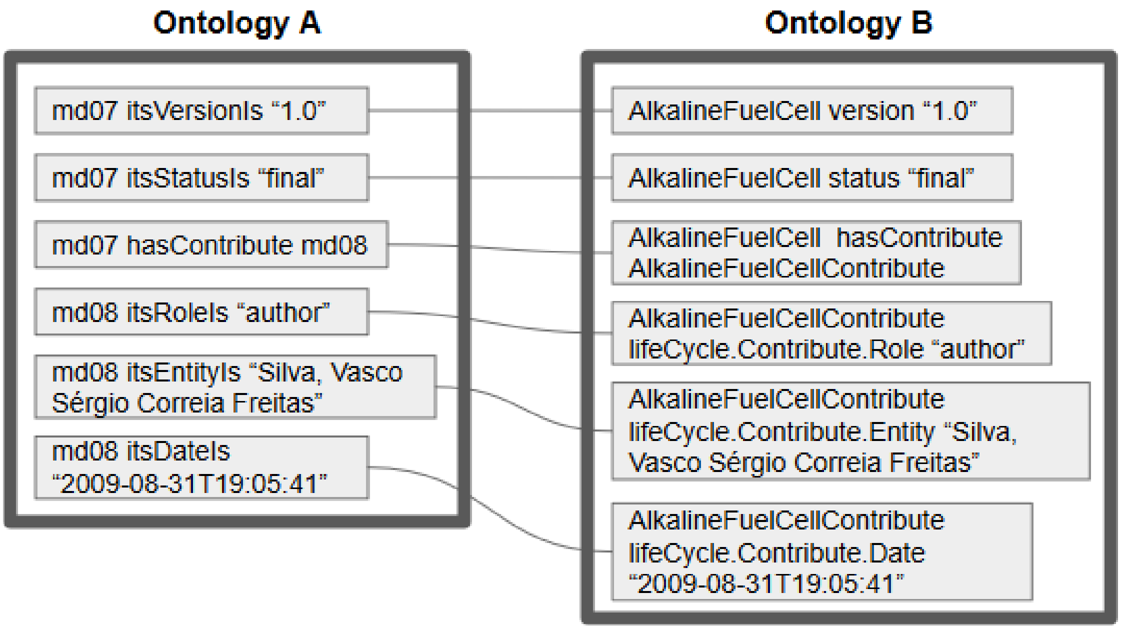


Figure 5. Example of equivalent triples between OBAA ontologies.

Table 1. IEEE-LOM mapping to unqualified Dublin Core Metadata Element Set [LTSC 2002].

DC.Identifier	1.1.2:General.Identifier.Entry
DC.Title	1.2:General.Title
DC.Language	1.3:General.Language
DC.Description	1.4:General.Description
DC.Subject	1.5:General.Keyword or 9:Classification with 9.1:Classification.Purpose equals "Discipline" or "Idea".
DC.Coverage	1.6:General.Coverage
DC.Type	5.2:Educational.LearningResourceType
DC.Date	2.3.3:LifeCycle.Contribute.Date when 2.3.1:LifeCycle.Contribute.Role has a value of "Publisher".
DC.Creator	2.3.2:LifeCycle.Contribute.Entity when 2.3.1:LifeCycle.Contribute.Role has a value of "Author".
DC.OtherContributor	2.3.2:LifeCycle.Contribute.Entity with the type of contribution specified in 2.3.1:LifeCycle.Contribute.Role.
DC.Publisher	2.3.2:LifeCycle.Contribute.Entity when 2.3.1:LifeCycle.Contribute.Role has a value of "Publisher".
DC.Format	4.1:Technical.Format
DC.Rights	6.3:Rights.Description
DC.Relation	7.2.2:Relation.Resource.Description
DC.Source	7.2:Relation.Resource when the value of 7.1:Relation.Kind is "IsBasedOn"

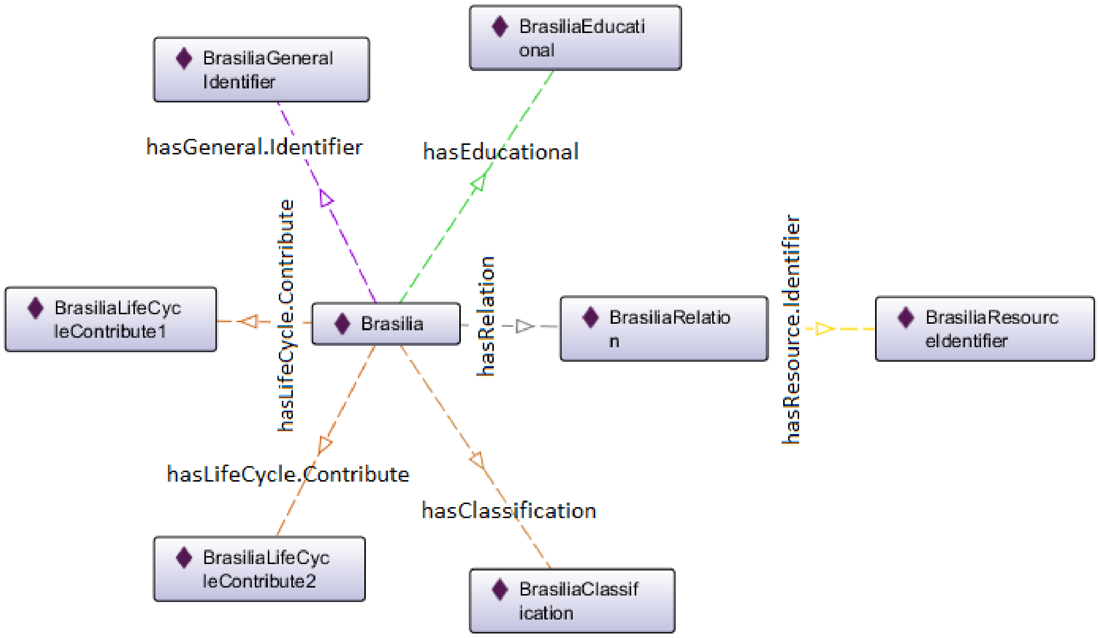
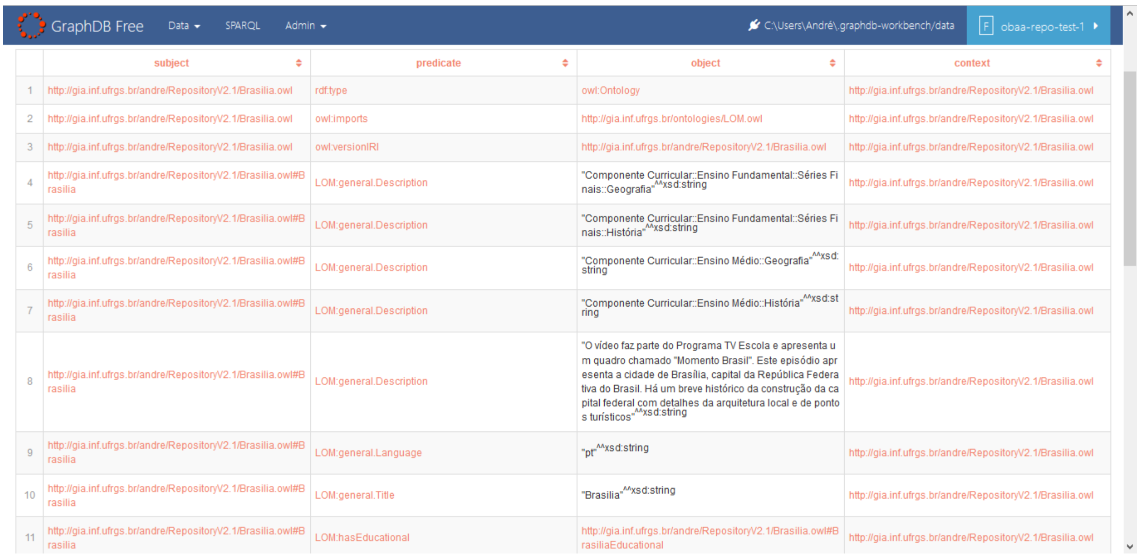


Figure 6. Relationships of Brasilia's individuals.

data [Pokornỳ 2015]. We explored GraphDB as an ontology repository alternative to keep the generated OWL. All ontologies were imported to the semantic repository. This storage can be retrieved through SPARQL or REST API.

Figure 7 shows the triples indexation of the Brasilia LO metadata ontology. All the ontologies could be retrieved with an SPARQL query. Besides, it is possible to query these ontologies to find LOs for students with special needs or classify LOs according to determined application profile.



	subject	predicate	object	context
1	http://gia.inf.ufrgs.br/andre/RepositoryV2.1/Brasilia.owl	rdf:type	owl:Ontology	http://gia.inf.ufrgs.br/andre/RepositoryV2.1/Brasilia.owl
2	http://gia.inf.ufrgs.br/andre/RepositoryV2.1/Brasilia.owl	owl:imports	http://gia.inf.ufrgs.br/ontologies/LOM.owl	http://gia.inf.ufrgs.br/andre/RepositoryV2.1/Brasilia.owl
3	http://gia.inf.ufrgs.br/andre/RepositoryV2.1/Brasilia.owl	owl:versionIRI	http://gia.inf.ufrgs.br/andre/RepositoryV2.1/Brasilia.owl	http://gia.inf.ufrgs.br/andre/RepositoryV2.1/Brasilia.owl
4	http://gia.inf.ufrgs.br/andre/RepositoryV2.1/Brasilia.owl#Brasilia	LOM:general.Description	"Componente Curricular::Ensino Fundamental::Séries Finais::Geografia"^^xsd:string	http://gia.inf.ufrgs.br/andre/RepositoryV2.1/Brasilia.owl
5	http://gia.inf.ufrgs.br/andre/RepositoryV2.1/Brasilia.owl#Brasilia	LOM:general.Description	"Componente Curricular::Ensino Fundamental::Séries Finais::História"^^xsd:string	http://gia.inf.ufrgs.br/andre/RepositoryV2.1/Brasilia.owl
6	http://gia.inf.ufrgs.br/andre/RepositoryV2.1/Brasilia.owl#Brasilia	LOM:general.Description	"Componente Curricular::Ensino Médio::Geografia"^^xsd:string	http://gia.inf.ufrgs.br/andre/RepositoryV2.1/Brasilia.owl
7	http://gia.inf.ufrgs.br/andre/RepositoryV2.1/Brasilia.owl#Brasilia	LOM:general.Description	"Componente Curricular::Ensino Médio::História"^^xsd:string	http://gia.inf.ufrgs.br/andre/RepositoryV2.1/Brasilia.owl
8	http://gia.inf.ufrgs.br/andre/RepositoryV2.1/Brasilia.owl#Brasilia	LOM:general.Description	"O vídeo faz parte do Programa TV Escola e apresenta um quadro chamado "Momento Brasil". Este episódio apresenta a cidade de Brasília, capital da República Federativa do Brasil. Há um breve histórico da construção da capital federal com detalhes da arquitetura local e de pontos turísticos"^^xsd:string	http://gia.inf.ufrgs.br/andre/RepositoryV2.1/Brasilia.owl
9	http://gia.inf.ufrgs.br/andre/RepositoryV2.1/Brasilia.owl#Brasilia	LOM:general.Language	"pt"^^xsd:string	http://gia.inf.ufrgs.br/andre/RepositoryV2.1/Brasilia.owl
10	http://gia.inf.ufrgs.br/andre/RepositoryV2.1/Brasilia.owl#Brasilia	LOM:general.Title	"Brasilia"^^xsd:string	http://gia.inf.ufrgs.br/andre/RepositoryV2.1/Brasilia.owl
11	http://gia.inf.ufrgs.br/andre/RepositoryV2.1/Brasilia.owl#Brasilia	LOM:hasEducational	http://gia.inf.ufrgs.br/andre/RepositoryV2.1/Brasilia.owl#BrasiliaEducational	http://gia.inf.ufrgs.br/andre/RepositoryV2.1/Brasilia.owl

Figure 7. Triples indexation of Brasilia LO metadata.

5. Conclusion and Future Work

This work brings a proposal to cope with the Semantic Web interoperability. The approach may collect data from different sources, process the information to exchange the results with other applications. The data is machine-readable and could be accessed by diverse intelligent software agents.

The resultant OWL description decreases problems with interpretation of data representation. Besides, the ontologies allow automatic checking of their consistency. Even losing XML order, as in a list of authors (sometimes in the order of relevance), the OWL format provides a richer representation. The representation in triples allows better relationships among concepts. Furthermore, it is possible to explore its axioms to relate with different ontologies. Equivalences can be made to align concepts in domain ontologies, establishing a richer world of description.

The XSLT-based approach provides a simple (independent of compiled languages), but powerful alternative to manage both XML and OWL. The XPath node selection can access XML nodes and OWL triples. However, this approach still has to lead to inconsistency, incomplete, duplicates, and different patterns.

Our ontology-based approach tries to lead with the aforementioned issues. The ontology provides a description that can be reused in metadata instances. Even been a propose-built ontology, it furnishes more expressivity of semantic description than metadata schemas.

As future work, it is aimed to provide and extended validation of the proposal by exploring performance issues, and a guide to building such type of ontology, as also to provide this methodology as a service. To accomplish this, we will analyze the use of a tool for learning object description that will provide OWL ontology alignment and conversion. The service also would be able to support conversions between different metadata standards.

References

Bedini, I., Matheus, C., Patel-Schneider, P. F., Boran, A., and Nguyen, B. (2011). Transforming xml schema to owl using patterns. In *Semantic Computing (ICSC), 2011 Fifth IEEE International Conference on*, pages 102–109. IEEE.

Behr, A., Primo, T., and Viccari, R. (2012). An ontology for the obaa metadata standard. In *Anais do Simpósio Brasileiro de Informática na Educação*, volume 23.

Berners-Lee, T., Hendler, J., Lassila, O., et al. (2001). The semantic web. *Scientific american*, 284(5):28–37.

DCMI (2012). Dublin core metadata element set, version 1.1.

Gluz, J. C. and Vicari, R. M. (2011). Uma ontologia owl para metadados ieee-lom, dublin-core e obaa. In *Anais do Simpósio Brasileiro de Informática na Educação*, volume 1.

Guarino, N., Oberle, D., and Staab, S. (2009). What is an ontology? In *Handbook on ontologies*, pages 1–17. Springer.

Hacherouf, M., Bahloul, S. N., and Cruz, C. (2015). Transforming xml documents to owl ontologies: A survey. *Journal of Information Science*, 41(2):242–259.

Heath, T. and Bizer, C. (2011). Linked data: Evolving the web into a global data space. *Synthesis lectures on the semantic web: theory and technology*, 1(1):1–136.

Hendler, J. (2001). Agents and the semantic web. *IEEE Intelligent systems*, (2):30–37.

Isotani, S. and Bittencourt, I. I. (2015). *Dados Abertos Conectados*. Novatec Editora.

Lacoste, D., Sawant, K. P., and Roy, S. (2011). An efficient xml to owl converter. In *Proceedings of the 4th India software engineering conference*, pages 145–154. ACM.

Liang, S., Taweel, A., Miles, S., Kovalchuk, Y., Spiridou, A., Barratt, B., Hoang, U., Crichton, S., Delaney, B., Wolfe, C., et al. (2015). Semi automated transformation to owl formatted files as an approach to data integration. *Methods of information in medicine*, 54(1):32–40.

LTSC (2002). Ieee standard for learning object metadata (draft). ieee standard 1484.12.1.

O'Connor, M. J. and Das, A. (2011). Acquiring owl ontologies from xml documents. In *Proceedings of the sixth international conference on Knowledge capture*, pages 17–24. ACM.

Palavitsinis, N., Manouselis, N., and Sanchez-Alonso, S. (2014). Metadata quality in learning object repositories: a case study. *The Electronic Library*, 32(1):62–82.

Pokornỳ, J. (2015). Graph databases: Their power and limitations. In *Computer Information Systems and Industrial Management*, pages 58–69. Springer.

Ribeiro, R. and Primo, T. T. (2013). Proposta para um algoritmo de conversão de metadados em ontologias. In *WebMedia 2013 WSEDU*.

Shadbolt, N., Hall, W., and Berners-Lee, T. (2006). The semantic web revisited. *Intelligent Systems, IEEE*, 21(3):96–101.

Sicilia, M.-A. (2014). *Handbook of metadata, semantics and ontologies*. World Scientific.

Silva, J. M. C. d. (2011). Análise técnica e pedagógica de metadados para objetos de aprendizagem.

Tani, A., Candela, L., and Castelli, D. (2013). Dealing with metadata quality: The legacy of digital library efforts. *Information Processing & Management*, 49(6):1194–1205.

Vicari, R. M., Ribeiro, A., da Silva, J. M. C., Santos, E. R., Primo, T., and Bez, M. (2010). Brazilian proposal for agent-based learning objects metadata standard-obaa. In *Metadata and Semantic Research*, pages 300–311. Springer.

Yahia, N., Mokhtar, S. A., and Ahmed, A. (2012). Automatic generation of owl ontology from xml data source. *arXiv preprint arXiv:1206.0570*.