LexRDF Model: An RDF-based Unified Model for Heterogeneous Biomedical Ontologies*

Cui Tao, Jyotishman Pathak, Harold R. Solbrig,
Wei-Qi Wei, and Christopher G. Chute

Division of Biomedical Statistics and Informatics, Mayo Clinic, Rochester, MN 55905
{tao.cui, pathak.jyotishman, solbrig.harold, wei.weiqi, chute}@mayo.edu

Abstract. The Lexical Grid (LexGrid) project is an on-going community-driven initiative coordinated by the Mayo Clinic Division of Biomedical Statistics and Informatics (BSI). It provides a common terminology model to represent multiple vocabulary and ontology sources as well as a scalable and robust API for accessing such information. While successfully used and adopted in the biomedical and clinical community, an important requirement is to align the existing LexGrid model with emerging Semantic Web standards and specifications. This paper introduces the LexRDF model, which maps the LexGrid model elements to corresponding constructs in W3C specifications such as RDF, OWL, and SKOS. Our mapping specification successfully used W3C standards to represent most of the existing LexGrid components, and those that did not map point out issues in the existing specifications that the W3C may want to consider in future work. With LexRDF, the terminological information represented in LexGrid can be translated to RDF triples, and therefore allowing LexGrid to leverage standard tools and technologies such as SPARQL and RDF triple stores.

1 Introduction

The evolution of ontologies and vocabularies in the biomedical domain, across the spectrum of detailed nomenclatures and sophisticated classifications, has accelerated dramatically over the last decade [2, 4, 5]. This coupled with the ability to access vast amounts of patient data in electronic medical records (EMR) provides the opportunity to build semantically interoperable healthcare applications and solutions for individualized and evidence-based medicine. However, in practice, the healthcare service providers and EMR system vendors alike confront the difficulties of incorporating elaborate ontologies and vocabularies into clinical workstations and data recording system clients in an intuitive, friendly, and responsive interface while preserving the expressive power and latent semantics of the ontologies. This can be primarily attributed to incompatible ontology representation formats, multiple ontology modeling languages, and the lack of appropriate tooling and programming interfaces which hinder the wide-scale adoption and usage of biomedical ontologies in a variety of application contexts.

* Supported in part by the National Center for Biomedical Ontology, and the NCI caBIG Vocabulary Knowledge Center

To address these issues, the Mayo Clinic Division of Biomedical Statistics and Informatics has been coordinating a community-wide initiative, called LexGrid, that is aimed at developing a common terminology model and programming interfaces for uniformly storing, representing, and querying biomedical ontologies and vocabularies [20]. The premise of the LexGrid project is that a common and consistent terminology model that defines a uniform representation and semantics is the cornerstone of multiple distribution formats, heterogeneous data stores, sharing and federation. Such a model provides a foundation for building consistent and standardized APIs to access multiple vocabularies that support a rich set of features such as lexical search queries, hierarchical navigation and recursive subsumption.

While successfully used and adopted in the biomedical and clinical community (see Section 2.1 for details), the current LexGrid model has not yet been formally aligned with the most recent Semantic Web (World Wide Web Consortium; W3C) standards and specifications [27]. We consider this a limitation and believe a representation of the LexGrid model in a combination of RDF, OWL, SKOS, and alike can enable the information rendered in LexGrid to be machine-readable and interpretable, thereby paving the way for information exchange between various applications. This study was to "*RDFize*" the LexGrid model by establishing a set of mappings between the LexGrid model elements to corresponding constructs in the appropriate W3C standards. This allows LexGrid represented terminology information rendered as RDF triples that can, for example, be queried using SPARQL [26]. We successfully mapped 37 out of 45 LexGrid elements, achieving a very high degree of reusability. For the remaining LexGrid elements that had no direct mapping (e.g., LexGrid *property*), we will begin a dialog with the respective W3C working groups about possible inclusion in a subsequent version of the appropriate specification.

We discuss the details of the mapping process in the remainder of this paper. Section 2 gives an overview of the LexGrid model and a brief introduction to the appropriate W3C standards. Section 3 discusses how we arrived at the LexRDF mapping specification. Section 4 discusses the issues we encountered, summarizes the extensions we will propose to the W3C community, and addresses the possible future directions.

2 Background

2.1 The LexGrid Project

The LexGrid project is an on-going community-driven initiative that builds upon a set of common tools, data formats, and read/update mechanisms for storing, representing and querying biomedical ontologies and vocabularies [20]. The primary goal of LexGrid is to accommodate multiple vocabulary and ontology distribution formats and support of multiple data stores for federated vocabulary and ontology access. The LexGrid model is designed to be flexible enough to faithfully and accurately represent a wide variety of multilingual terminological resources. The model defines how ontologies are formatted and represented

programatically, and is intended to be flexible enough to accurately represent a wide variety of ontologies developed in different languages and other lexically-based knowledge resources. The model also defines several different server storage mechanisms (e.g., relational databases, Lightweight Data Access Protocol (LDAP)) and an XML format. It provides the core representation for all data managed and retrieved through the API, and can represent vocabularies provided in numerous source formats including the (*i*) Open Biomedical Ontologies (OBO) [16], e.g., Gene Ontology [8]; (*ii*) Web Ontology Language (OWL), e.g., NCI Thesaurus [15]; (*iii*) Unified Medical Language System (UMLS) Rich Release Format, e.g., NCI Metathesaurus [14], which comprises of extensive clinical terminologies such as SNOMED CT [25]; and (*iv*) Classification Markup Language (ClaML), e.g., International Classification of Diseases version 10 (ICD-10) [9]. In particular, the terminology and ontology resources in LexGrid are intended to be:

- represented using a common terminology model
- accessible through a set of common API's
- joined through shared indices
- accessible online
- downloadable
- cross-linked (i.e., have inter-ontology mappings)
- loosely coupled
- locally extensible
- globally revised, and
- available in web-space on web-time

Interested readers can review more details about the LexGrid model in [20].

Once disparate vocabulary information can be represented in a standardized model, it becomes possible to build common repositories to store vocabulary content and common programming interfaces and tools to access and manipulate that content. Toward this end, LexGrid provides a semantic foundation upon which multiple APIs can be developed that support consistent searching, navigation and cross terminology traversal. Existing API implementations include the LexEVS API [11], a reference implementation of the HL7 Common Terminology Services (CTS), and the LexWiki model [12] for representing terminology within a semantic mediawiki. These open-source tools are used in a variety of projects both internal and external to the Mayo Clinic, including the NCI Cancer Biomedical Informatics Grid (caBIG) [3], the National Center for Biomedical Ontology (NCBO) [13], the Biomedical Grid Terminology project (BioMed GT) [1], and the World Health Organization International Classification of Diseases (ICD-11) development process [10]. Our experience in developing and deploying the LexGrid technology provides an unparalleled basis for using ontologies to represent patient and clinical trial information, thereby enabling semantic information retrieval.

2.2 W3C Standard Recommendations for the Semantic Web

The World Wide Web Consortium (W3C) is the main international standards organization for the World Wide Web. Its goal is to develop interoperable technologies and tools as well as specifications and guidelines to lead the Web to its full potential. W3C recommendation has several maturity levels: Working Draft, Candidate Recommendation, Proposed Recommendation, and W3C Recommendation. The W3C recommendations we evaluated and compared with LexGrid , and included in our mapping are the followings. The Resource Description Framework (RDF) [21], RDF Schema [22], the Web Ontology Language (OWL) [18] are W3C recommendations. OWL 2 [19] and Simple Knowledge Organization System (SKOS) [23] are W3C proposed recommendations. And SKOS eXtension for Labels (SKOS-XL) [24] is a W3C candidate recommendation. In addition to these W3C recommendations, we also considered and included Dublin Core metadata element set (dc) [6] and DCMI Metadata Terms (dcterm) [7] which are widely used to describe digital materials.

As illustrated above, one of our goals is to align the LexGrid model with emerging W3C semantic Web recommendations to facilitate interoperability as well as more wide-scale adoption of our tools and software infrastructure. In the remainder of this paper, we discuss our experience and share the outcomes as we embark upon this effort.

3 LexRDF Mapping Specifications

Our primary task was to determine equivalent constructs or axioms in the W3C recommendations introduced in Section 2.2 for each LexGrid element. In the case where appropriate mapping is lacking from the W3C recommendations, we proposed new constructs in the LexRDF name space. These extensions will be proposed to appropriate W3C committee for future inclusion in the W3C recommendations.

3.1 Ontology Information Mapping

LexGrid	**LexRDF**
codingScheme	owl:Ontology
source	dc:source
copyright	dc:right
codingSchemeName	rdfs:label
codingSchemeURI	xmlns
representVersion	owl:versionInfo
formalName	dc:title
defaultLanguage	dc:language
approxNumConcepts	N/A

Table 1. Ontology Information Mapping

LexGrid	**LexRDF**
entity	skos:Concept
entityType	implicit
concept	owl:Class
instance	owl:Thing
association	owl:objectProperty owl:datatypeProperty
entityCode	rdf:ID
entityCodeNamespace	xmlns
isAnonymous	implicit
isDefined	LexRDF:isDefined

Table 2. Entity Mapping

LexGrid comprises various lexical elements describing meta-data about an ontology. These include provenance (source (*dc:source*), copyright (*dc:right*), version (*owl:verionInfo*)), name (*dc:title, rdf:label*), URI, and language (*dc:language*). Table 1 shows the LexRDF mapping specification for ontology information. LexRDF successfully identified mappings for all the LexGrid ontology-information components except one: *approxNumConcepts*, which indicates the total number of ontological entities present in a given/loaded ontology. This attribute was intended as a hint to service components, especially for the large-size ontologies. Since this information can be inferred from the ontology itself, we chose to exclude it from this mapping.

3.2 Entity Mapping

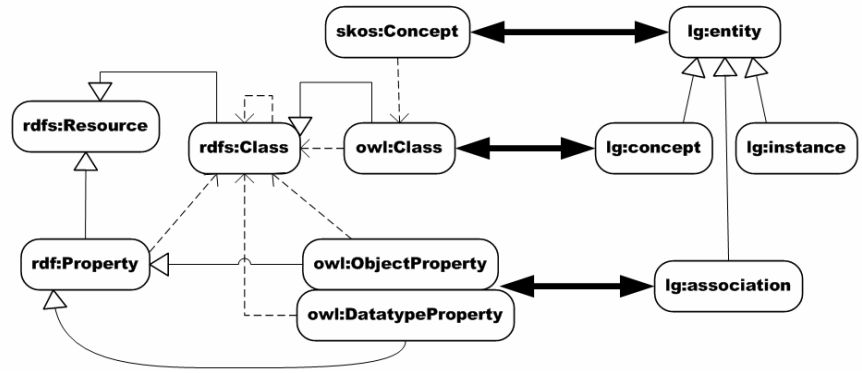

Fig. 1. LexRDF Entity Definition Overview

A LexGrid *entity* represents any resource in a terminology or ontology. Figure 1 shows the syntax graph of the LexGrid entity components. A dashed arrow from element A to element B indicates that A is an instance of B. An arrow with a clear arrowhead from A to B indicates that A is a subclass of B. We use *lg* to represent the LexGrid name space. LexGrid has defined *lg:concept*, *lg:association*, and *lg:instance* as subclasses of *lg:entity*. LexRDF maps *lg:concept* to *owl:Class*, meaning that *lg:concept* inherits the definition of *owl:Class*—both an instance and a subclass of *rdfs:Class*. The *lg:association* element is equivalent to the union of *owl:ObjectProperty* and *owl:DatatypeProperty*, which are both instances of *rdfs:Class* and subclasses of *rdf:Property*. The *lg:instance* element is a general holder of OWL individuals which are instances of OWL classes. LexRDF uses *owl:Thing* to declare a LexGrid instance in RDF triple representation when no specific type is defined for an instance. LexRDF also maps *lg:entity* to *skos:Concept*, which is defined as an instance of *owl:Class*. This mapping specification preserves the original LexGrid definition without introducing any contradictions of definition in the recommendation name spaces.

Table 2 specifies LexRDF mappings for the LexGrid components related to entities. In addition to the mapping specification discussed above, each entity has an *entityCode* which is used as the URI for the corresponding entity in LexRDF. The *entityCodeNamespace* is the xmlns in LexRDF. LexGrid represents the anonymous classes in OWL using anonymous concepts. In this case, the *isAnonymous* flag is set to be true in the loaded code system. In all other cases, the *isAnonymous* flag is false. We believe this information is implicitly expressed in OWL, therefore we did not specify a mapping for *isAnonymous*. LexGrid also defined a *isDefined* flag (true means that the entity is considered to be completely defined (i.e. necessary and sufficient) within the context of the containing code system; and false means that only the necessary components are present). We use *LexRDF:isDefined* to represent this flag. The domain of *LexRDF:isDefined* is *skos:Concept* and the range is boolean values.

3.3 Property Mapping

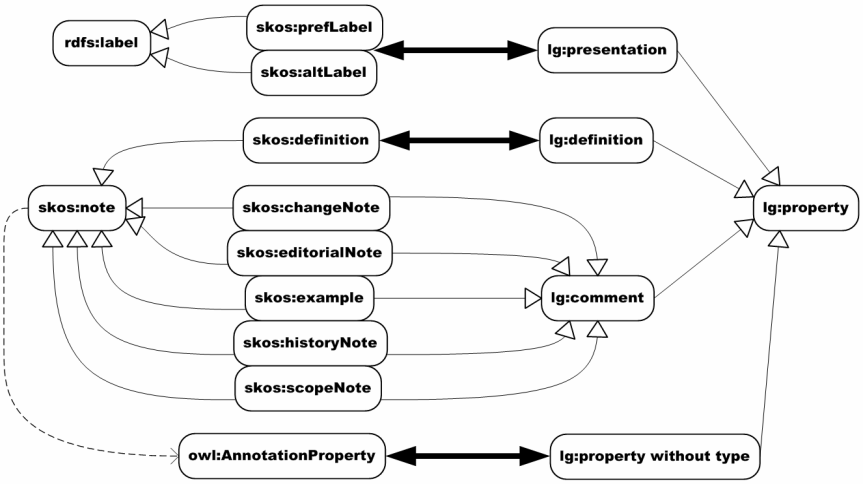

Fig. 2. LexRDF Property Definition Overview

Every instance of a LexGrid entity is associated with a set of properties, which are analogous to annotation properties in OWL. Table 3 shows the LexRDF mapping specification for property information and Figure 2 shows the property definition overview. Each *lg:property* could have an optional type (comment, presentation, or definition). Each *lg:presenation* and *lg:definition* has a *isPreferred* flag which indicates whether it was "preferred" in the given language and context. When no type is specified, a *lg:property* is mapped to an *owl:AnnotationProperty*. The *lg:comment* is a super property of *skos:changeNote*, *skos:editorialNote*, *skos:example*, *skos:historyNote*, and *skos:scopeNote*. The *lg:presentation* is

LexGrid	OWL
property	owl:AnnotationProperty when no type specified
language	dc:language
source	dc:source
propertyType	implicit
comment	skos:note except skos:definition
presentation	skos:altLabel, skos:prefLabel
definition	skos:definition
isPreferred	LexRDF:isPreferred
degreeOfFidelity	LexRDF:degreeOfFidelity
matchIfNoContext	LexRDF:matchIfNoContext
representationalForm	LexRDF:representationalForm
propertyLink	LexRDF:propertyLink

Table 3. Property Mapping

mapped to *skos:prefLabel* when the *isPreferred* flag is set to *true* and to *skos:altLabel* otherwise. The LexGrid *definition* element is mapped to *skos:definition*. LexRDF uses a *LexRDF:isPreferred* construct to reify whether a definition is preferred or not.

As an example, Figure 3 illustrates how LexRDF presents entity property and property reification. Figure 3(a) shows the original representation of a sample term in the OBO [16] format. Figure 3(b) shows how LexGrid represents it and Figure 3(c) shows the LexRDF representation. LexGrid presents the OBO term as an *entity* with the entity type as *concept*. The two *presentations* in Figure 3(b) represent lines 3 and 5 in Figure 3(a); and the *definition* in Figure 3(b) represents line 4 in Figure 3(a). LexRDF specifies the term *FAO:0000025* as an *owl:Class* and has a *skos:prefLabel* "mid reproductive" which is represent as a preferred presentation in LexGrid. LexRDF also uses *skos:altLabel* to represent the property with the *lg:isPreferred* flag set to false. The definition of this term has a source information "TAIR:lr". LexRDF uses RDF reification to reify the source of the definition. It creates an anonymous node `A1` which is a *rdf:statement* and then defines the subject, object, and predicate of `A1` as rows 4-7 in Figure 3(c) show. The representation is equivalent to the triple `FAO:0000025 skos:definition ''middle stages of reproductive phase.''`. LexRDF then reifies that `A1` has a source "TAIR:lr" using the predicate *dc:source*. LexGrid also set this definition as a preferred one by default. Therefore LexRDF reified `A1` as a preferred definition using predicate *LexRDF:isPreferred* as row 9 in Figure 3(c) shows.

LexGrid uses *propertyLink* to define relationships between two properties. LexRDF defined a new annotation property, *LexRDF:propertyLink*. Each property link is defined as an instance of *owl:ObjectProperty* and a sub-property of *LexRDF:propertyLink*. LexRDF uses RDF reification to define a link between two properties. Figure 4 shows an example. A concept `A` has a preferred presentation "FAO", and another presentation "Food and Agriculture Organization". The relation between the two presentations is that the former is an acronym of the latter. The LexRDF representation is as fellows. `A1` and `A2` are the two properties of concept `A`. The relationship between `A1` and `A2` is *sns:acronymOf* where *sns* represents the source name space. And *sns:acronymOf* is also defined as a sub-property of *LexRDF:propertyLink*.

(a)

```
1.  [Term]
2.  id: FAO:0000025
3.  name: mid reproductive
4.  def: ‘‘middle stages of reproductive phase.’’ [TAIR:lr]
5.  synonym: ‘‘principal growth stages 6.1-6.3’’
```

(b)

```
Entity Code:FAO:0000025
Entity Type:Concept
Presentation:mid reproductive
      Property Name:textualPresentation
      is Preferred:true
Presentation:‘‘principal growth stages 6.1-6.3’’
      Property Name:synonym
      is Preferred:false
Definition:‘‘middle stages of reproductive phase.’’
      Property Name:definition
      is Preferred:true
      Source:TAIR:lr
```

(c)

	Subject	**Predicate**	**Object**
1	FAO:0000025	rdf:type	owl:Class
2	FAO:0000025	skos:prefLabel	mid reproductive
3	FAO:0000025	skos:altLabel	“principal growth stages 6.1-6.3”
4	A1	rdf:type	rdf:Statement
5	A1	rdf:subject	FAO:0000025
6	A1	rdf:predicate	skos:definition
7	A1	rdf:object	“middle stages of reproductive phase.”
8	A1	dc:source	TAIR:lr
9	A1	lexRDF:isPrefered	true

Fig. 3. An Example of Property and Property Reification (fungal_anatomy.obo)

LexRDF also defined three new annotation properties: *LexRDF:degreeOfFidelity*, *LexRDF:matchIfNoContext*, and *LexRDF:representationalForm*. The degree of fidelity states how closely a term approximates the intended meaning of an entry code. The *MatchIfNoContext* flag should be set to true when the entity presentation is valid in a contextual setting. The representational form states how the term represents the concept (abbreviation, acronym, etc.).

3.4 Association Mapping

LexGrid uses associations to represent relationships between entities. The association definition may also further define the nature of the relationship such as forward and inverse names, transitivity, symmetry, reflexivity, and etc. Table 4 shows the LexRDF mapping specification for LexGrid association elements. LexRDF used OWL properties and assertions to represent all of them except *reverseName*, *isAntiTransitive*, and *isNavigable*. LexRDF uses a new construct *LexRDF:reverseName* to represent the name of the association on the reverse direction when a target to source side of the association is meaningful. *LexRDF:isAntiTransitive* is used to represent a property that is not transitive.

```
<A1> rdf:type rdf:Statement; rdf:subject <A>;
     rdf:predicate skos:prefLabel; rdf:object "FAO";
<A2> rdf:type rdf:Statement;  rdf:subject <A>;
     rdf:predicate skos:altLabel;
     rdf:object "Food and Agriculture Organization";
<A1> sns:acronymOf <A2>;
 sns:acronymOf rdf:subProperty LexRDF:propertyLink;
```

Fig. 4. An Example of Property Link

LexGrid	**OWL**
associationName	rdf:ID
forwardName	rdf:ID
reverseName	LexRDF:reverseName
inverse	owl:inverseOf
isTransitive	owl:TransitiveProperty
isSymmetric	owl:SymmetricProperty
isAntiTransitive	LexRDF:AntiTransitiveProperty
isReflexive	owl:ReflexiveProperty
isFunctional	owl:FunctionalProperty
isReverseFunctional	owl:InverseFunctionalProperty
isNavigable	LexRDF:isNavigable
associationQualification	LexRDF:assocaitionQualification

Table 4. Association Mapping

	Subject	**Predicate**	**Object**
1	Poland_anomaly	rdf:type	owl:class
2	Dextrocardia	rdf:type	owl:class
3	Poland_anomaly	rdfs:subClassOf	A1
4	A1	rdf:type	owl:Restriction
5	A1	owl:onProperty	HAS_CLINICAL_SIGN
6	A1	owl:someValuesFrom	Dextrocardia
7	A1	sns:Frequency	“Very frequent”
8	sns:Frequency	rdf:subProperty	lexRDF:associationQualification
9	HAS_CLINICAL_SIGN	rdf:type	owl:objectProperty

Table 5. RDF Triples for an Example of AssociationQualifier

In addition, an association could be modified by using LexGrid *associationQualification*. LexRDF uses *LexRDF:isNavigable* to describe that the reverse direction of the association is "navigable", meaning that it makes sense to represent the target to source side of the association. For example, one can define *Poland anomaly* $\frac{HAS\_CLINICAL\_SIGN}{Frequency=Very\ frequent}$ *Dextrocardia*, where `HAS_CLINICAL_SIGN` is the association name, *Poland anomaly* is the association source and Dextrocardia is the association target. This association instance also has an association qualification indicates how frequently the disease has the symptom. The association qualification has a name `Frequency` and a value `Very frequent`. Table 5 shows how LexRDF represents this example. By default, LexRDF uses OWL *someValuesFrom* restriction to represent an association instance. LexRDF first declares an anonymous note `A1` for the association instance (rows 3-6 in Table 5). For *associationQualification*, LexRDF defined a new OWL annotation property, *LexRDF:associationQualification*. Every actual association qualifier is defined as a sub-property of *LexRDF:associationQualification*, and therefore is also an instance of OWL annotation property. Rows 7-8 show how LexRDF defines and reifies association qualifiers.

4 Discussion, Conclusion, and Future Work

We discussed the LexRDF mapping specification with respective to ontology information, entity, property, and association. LexRDF has successfully mapped 37 out of 45 LexGrid elements, achieving a very high degree of reusability. We have also discovered some interesting issues where the W3C recommendation languages cannot fully represent our needs in LexGrid.

Generic holder for properties and comments As Figure 2 shows, LexGrid has a common superclass *lg:property* for comments, presentations, and definitions. In LexRDF, we use *skos:prefLabel* and *skos:altLabel*, both of which are sub-properties of *rdfs:label*, to represent *lg:comment*; we use *skos:definition*, which is an instance of *owl:AnnotationProperty*, to represent *lg:definition*. The properties in the subset of *skos:note* which we use to represent *lg:comment* are also defined as instances of *owl:AnnotationProperty*. SKOS provides *skos:notes* as a general superset for definition, example, and a set of different notes. But it does not define a common ancestor for labels, and notes. We cannot find an appropriate component to represent generic properties. We have a similar problem with *lg:comment*. Currently it is mapped to a set of sub-properties of *skos:note*, but a generic comment class is also preferred.

Preferred properties SKOS has defined *prefLabel* and *altLabel*, but no such constructs are provided for "definitions". Currently, we are using *LexRDF:isPreferred* as a tag to specify whether a definition is preferred or not. Akin to *prefLabel* and *altLabel*, our objective is to propose *prefDefinition* and *altDefinition* to the SKOS committee to be introduced in the future specification.

Association Qualification LexGrid provides an option for modifying an association instance by adding association qualifiers. We have found this to be

needed in the clinical domain and believe that it is an important requirement to be considered by the appropriate W3C standards group.

Relation among properties We have a requirement to describe relations among properties. SKOS provides *skosxl:labelRelation* that can represent relations between two labels. The property *skosxl:labelRelation*, however, is defined as a symmetric property with domain and range as *skosxl:Label*. These limitations restrict us from using it for our LexGrid *propertyLink*. We proposed a more general property *LexRDF:propertyLink* which is a super-property of *skosxl:labelRelation*. By using *LexRDF:propertyLink*, we can define relations between any two LexGrid properties. For example, we can assert that a particular label is an acronym of another, or that a given definition is a literal translation of the same definition in another language.

Property groups LexGrid is represent in UML where each concept could have multiple attributes defined. For example, The LexGrid *property* element has attributes *name* and *value*. Same as *associationQualification*. How to represent this situation was a challenge for us. Currently, LexRDF defines each generic property or association qualification using a new OWL annotation property with its *name* value as the URI (i.e., *sns:Frequency*). These new properties are also defined as sub-properties of either *LexRDF:entityProperty* or *LexRDF:associationQualification*. This approach brings new interoperability problems since many new annotation properties were being defined. We need to design a mechanism which can be used to represent a group of properties (i.e, *name* and *value*), then use this group to reify other elements.

In addition, we encountered the similar issue with association qualifications. Sometimes one association might have multiple groups of qualifiers. For example, in UMLS we can have an association `C001 PAR C002`, where `PAR` is the association, `C001` is the source, and `C002` is the target. This association has two groups of qualifiers: {`Rela=sub Type, Sab=LNC`} and {`Rela=is a, Sab=SNOMED`}. We should consider defining a *propertyGroup* similar to *owl:propertyChain* where a group of properties can be defined together.

Missing lexical constructs For some lexical information in LexGrid (e.g., degreeOfFidelity, representationalForm, isDefined), we cannot specify mappings. Coding and tags for these properties are being developed in the ISO TC37 community (http://www.tc37sc4.org/index.php) which we believe should be merged into the W3C specifications. We have initiated communication with the respective W3C working groups for their inclusion in appropriate specifications.

In summary, this paper introduced our on-going work to map the elements from the HL7 and ISO compliant LexGrid model to various Semantic Web recommendations. Although mostly successful, we have identified several limitations of the existing W3C specifications that warrant broader community engagement.

Several directions remain to be pursued. We are working on implementing a "bridge" that can load the LexGrid content and transferred it to an RDF triple store according to the LexRDF mapping specification. We would also like to formalize the LexRDF mapping specification by using standards such as the OMG Ontology Definition Metamodel (ODM) [17].

References

1. Biomedical grid terminology. http://www.biomedgt.org.
2. Olivier Bodenreider. Biomedical Ontologies in Action: Role in Knowledge Management, Data Integration and Decision Support. In A. Geissbuhler and C. Kulikowski, editors, *IMIA Yearbook of Medical Informatics*, volume 47, pages 67–79. International Medical Informatics Association, 2008.
3. The national cancer institute cancer biomedical informatics grid (caBIG). http://cabig.nci.nih.gov.
4. Christopher G. Chute. The Copernican Era of Healthcare Terminology: A Re-Centering of Health Information Systems. In *AMIA Annual Symposium*, pages 68–73, 1998.
5. Christopher G. Chute. Clinical Classification and Terminology: Some History and Current Observations. *Journal of American Medical Informatics Association*, 7(3):298–303, 2000.
6. DCMI namespace for the Dublin Core metadata element set. http://purl.org/dc/elements/1.1/.
7. DCMI metadata terms. http://dublincore.org/documents/dcmi-terms/.
8. The gene ontology. http://www.geneontology.org.
9. International classification of diseases version 10. http://www.who.int/classifications/icd/en/.
10. International classification of diseases version 11. http://www.who.int/classifications/icd/ICDRevision.
11. LexEVS API. http://gforge.nci.nih.gov/projects/lexevs.
12. LexWiki API. https://cabig-kc.nci.nih.gov/Vocab/KC/index.php/LexWiki.
13. The national center for biomedical ontology. http://www.bioontology.org.
14. NCI metathesaurus. http://ncimeta.nci.nih.gov.
15. NCI thesaurus. http://nciterms.nci.nih.gov.
16. The open biomedical ontologies. http://www.obofoundry.org/.
17. The ontology definition metamodel (ODM). http://www.omg.org/spec/ODM/1.0/, 2009.
18. RDF schema for OWL Full. www.w3.org/2002/07/owl.
19. OWL 2 web ontology language structural specification and functional-style syntax. http://www.w3.org/TR/owl2-syntax/.
20. J. Pathak, H.R. Solbrig, J.D. Buntrock, T.M. Johnson, and C.G. Chute. LexGrid: A Framework for Representing, Storing, and Querying Biomedical Terminologies from Simple to Sublime. *Journal of the American Medical Informatics Association*, 16(3):305–315, 2009.
21. The RDF vocabulary. http://www.w3.org/1999/02/22-rdf-syntax-ns.
22. The RDF schema vocabulary (RDFS). http://www.w3.org/2000/01/rdf-schema.
23. SKOS vocabulary. http://www.w3.org/2006/07/SWD/SKOS/reference/20090315/skos.rdf.
24. SKOS XL vocabulary. http://www.w3.org/2006/07/SWD/SKOS/reference/20090315/skos-xl.rdf.
25. Systematized nomenclature of medicine–clinical terms (SNOMED CT). http://www.snomed.org.
26. SPARQL Query Language for RDF. www.w3.org/TR/rdf-sparql-query/.
27. C. Tao, J. Pathak, H.R. Solbrig, and C.G. Chute. LexOWL: A bridge from LexGrid to OWL. In *Proceedings of the First International Conference of Biomedical Ontology (ICBO 09)*, pages 131–134, Buffalo, New York, July 2009.