PAROLE/SIMPLE 'lemon' ontology and lexicons

Editors: Sebastian Hellmann, AKSW, University of Leipzig, Germany; Steven Moran, LMU Munich, Germany; Martin Brümmer, AKSW, University of Leipzig, Germany; John McCrae, University of Bielefeld, Germany.
Open reviews: Judith Eckle-Kohler, TU Darmstadt, Germany; Philipp Cimiano, University of Bielefeld, Germany; one anonymous reviewer.

Marta Villegas and Núria Bel
Universitat Pompeu Fabra

Abstract. The PAROLE/SIMPLE 'lemon' Ontology and Lexicon are the OWL/RDF version of the PAROLE/SIMPLE lexicons (defined during the PAROLE (LE2-4017) and SIMPLE (LE4-8346) IV FP EU projects) once mapped onto lemon model and LexInfo ontology. Original PAROLE/SIMPLE lexicons contain morphological, syntactic and semantic information, organized according to a common model and to common linguistic specifications for 12 European languages. The data set we describe includes the PAROLE/SIMPLE model mapped to lemon and LexInfo ontology and the Spanish & Catalan lexicons. All data are published in the Data Hub and are distributed under CC Attribution 3.0 Unported license. The Spanish lexicon contains 199466 triples and 7572 lexical entries fully annotated with syntactic and semantic information. The Catalan lexicon contains 343714 triples and 20545 lexical entries annotated with syntactic information half of which are also annotated with semantic information. In this paper we describe the resulting data, the mapping process and the benefits obtained. We demonstrate that the Linked Open Data principles prove essential for datasets such as original PAROLE/SIMPLE lexicons where harmonization and interoperability were crucial. The resulting data is lighter and better suited for exploitation. In addition, it facilitates further extensions and linking to external resources such as WordNet, lemonUby, DBpedia etc.

Keywords: lexicon, ontology, open linked data, RDF, OWL, LE-PAROLE, SIMPLE, LexInfo, lemon

1. Introduction

The PAROLE/SIMPLE 'lemon' Ontology is the OWL/RDF version of the PAROLE & SIMPLE lexicon models (defined during the PAROLE LE2-4017 and SIMPLE LE4-8346 projects) once mapped to lemon[1] and LexInfo[2] models.

1.1. PAROLE/SIMPLE lexicons

The original PAROLE/SIMPLE[3] lexicons contain morphological, syntactic and semantic information organized according to a common model and to common linguistic specifications. PAROLE was the first project producing corpora and lexicons in so many languages[4] and built according to the same design principles, linguistic specifications and representation format. The model was based on EAGLES recommendations for morphosyntactic information and verb syntax [7] and on the extended GENELEX model [1].

The goal of SIMPLE project was to add semantic information to the set of harmonized multifunctional lexicons built for 12 European languages by the PAROLE consortium. All PAROLE/SIMPLE lexicons were defined against a common model defined in the DTD. Thus all PAROLE/SIMPLE lexicons are XML files valid against the same DTD[5]. In addition, a good number of 'descriptive' elements were defined and shared by all SIMPLE lexicons. Essentially, these in-clude: (i) Template assignment: meant to guarantee coherent encoding, across sites and languages, (ii) Domain information, (iii) Semantic class informa-tion, (iv) Semantic features: distinctive features used to better specify the semantic class of a sense, and for the definition of selectional restrictions

[1] http://lemon-model.net/
[2] http://lexinfo.net/
[3] http://www.ub.edu/gilcub/SIMPLE/simple.html
[4] Catalan, Danish, Dutch, English, Finnish, French, German, Greek, Italian, Portuguese, Spanish and Swedish

[5] Original PAROLE/SIMPLE lexicons were in SGML so we previously converted them into XML.

on the arguments (v) Semantic Roles and (vi) Semantic Rela-tions.

1.2. LMF, lemon and LexInfo

LMF [5] (Lexical Markup Framework) is an ISO standard (ISO-24613:2008) for computational lexicons. LMF combines the best designs and methods from many existing NLP lexicons[6]. LMF models are represented by UML classes, associations among the classes, and a set of ISO-12620 data categories that function as UML attribute-value pairs. LMF includes an XML DTD where XML elements in the DTD are transcoded from the UML class diagrams. The class adornment is implemented as a set of *feat* elements.

Lemon[3][6] ('lexicon model for ontologies' developed by the Monnet project http://www.monnet-project.eu/) is a model for modeling lexica in RDF. The lemon model consists of a core path defined as: OntologyEntity ↔ LexicalSense ↔ LexicalEntry → LexicalForm → WrittenRepresentation. Lemon is highly compliant with LMF.

LexInfo [2][4] is a model for the linguistic grounding of ontologies and as such allows for the association of linguistic information (such as part-of-speech, subcategorization frames etc.) with ontology elements (such as concepts, relations, individuals, etc.). LexInfo it is also highly compliant with LMF and the lemon model.

1.3. The mapping

Mapping PAROLE/SIMPLE lexicons onto lemon/LexInfo involves three tasks. Firstly, the original PAROLE/SIMPLE model expressed in the DTD needs to be mapped onto the lemon model. This can be seen as the lexicon format mapping. Secondly, all descriptive elements defined by PAROLE/SIMPLE lexicons are mapped onto the LexInfo ontology. This includes language dependent descriptive elements and common elements[7]. This broadly corresponds to the ontology mapping part. Finally, lexical entries are mapped.

The resulting dataset is organized into three files. One contains the PAROLE/SIMPLE Ontology which essentially imports lemon and LexInfo ontologies and adds 'PAROLE elements' (classes and/or properties) whenever these could not be mapped. The other two files collect the Spanish and Catalan lexical entries.

In the following sections we describe the clues of the mapping process and highlight some of the benefits obtained.

2. From PAROLE/SIMPLE model to lemon

The strategy followed when mapping PAROLE/SIMPLE model onto lemon can be summarized as follows:

Elements from the DTD were mapped onto classes. Whenever possible, lemon (and LexInfo) classes were used. Otherwise, new classes were created. For example: PAROLE *Description* elements become *lemon:Frames*. In contrast, the *parole:Connotation* class was created as a subclass of *parole:Element* and *lemon:PropertyValue* as shown in Figure 1. Note that many PAROLE/SIMPLE elements are not mapped and simply disappear in the target model. This is partially due to the fact that RDF allows a better modeling and they are no longer needed.

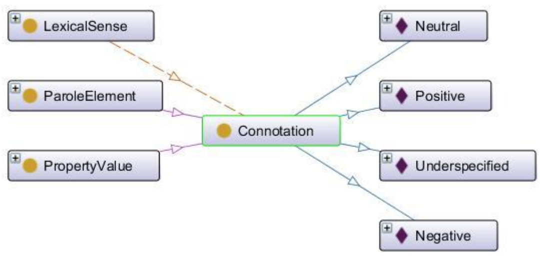

Figure 1 'Adding classes'

Attributes from the DTD were mapped onto Properties. Again, whenever possible, lemon or LexInfo properties were used. For example: PAROLE MuS/@gramcat[8] becomes *lexinfo:partOfSpeech*.

Values. When the PAROLE/SIMPLE DTD establishes the set of values for a given attribute, these values are mapped onto the corresponding LexInfo values. For example: the PAROLE pair: "NOUN" + "COMMON" simply translates as *lexinfo:commonNoun* as shown in Figure 2.

Parent/child relations between elements in the DTD were mapped onto relevant Properties. For example: the parent/child relation between a PAROLE

[6] Especially GENELEX, PAROLE and SIMPLE.

[7] Note that, whereas PAROLE lexicons are structurally compatible, in certain aspects they are semantically idiosyncratic as each lexicon defines its own 'descriptive' elements. Thus for example, subcategorization frames are defined in each lexicon without any reference or relation to the others. In contrast, SIMPLE lexicons go one step further and define a set of shared descriptive elements.

[8] We use XPath expressions when referring to source data.

verbal *Construction* and its subject *InstantiatedPositionC* element becomes *lexinfo:subject* property.

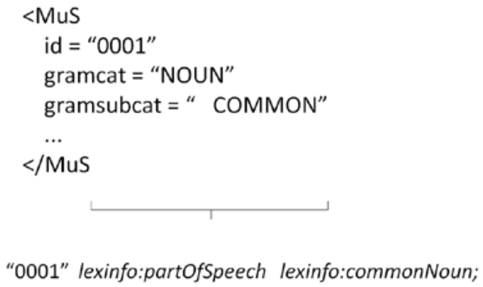

Figure 2 ‘Attribute mapping’

IDREFs pointing mechanisms between elements in the DTD became properties. For example: the relation between PAROLE morphological and syntactic units (*MuS* & *SynUs*) is expressed by means of the *lemon:synBehaviour* property as shown in Figure 3.

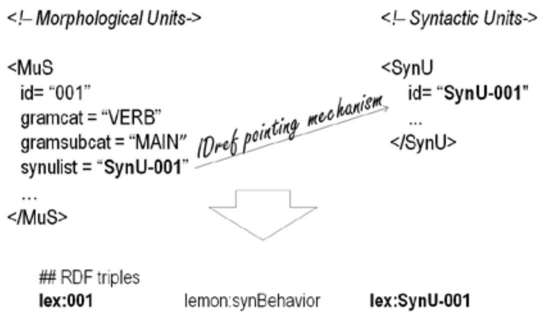

Figure 3 ‘Mapping the IDREF pointing mechanism’

Though the mapping process implied a considerable effort we think the task was worth it. The source model (DTD) and common descriptive elements are already mapped and can be reused by other languages. This process involved two main aspects: a change of model (both conceptually[9] and formally) and a change of vocabulary which was a rather painful task[10]. Lexical entries and language dependent data in source lexicons will require additional mapping processes. Once the mapping between the two models is defined, rewriting the original entries is not difficult and conversion rules can be easily applied. The biggest problem here is the difficulty to get relevant PAROLE data. Whether the data are in XML files or in a data base, the fact is that gathering data for a particular lexical entry is quite a complex task. In PAROLE, a lexical entry is split into different elements across different layers which makes this task rather complex. Our strategy was to follow PAROLE layered structure; thus, for each layer we generated the relevant triples. Using source IDs (and IDREFs) to create the target URIs allows for such a strategy and guarantees consistency of data. The migration of subcategorization frames deserves special attention: in the PAROLE model, syntactic frames are defined locally in each language lexicon. This means that the conversion to the lemon model needs to be addressed for each particular lexicon and its difficulty depends on the nature of the source data[11]. In any case, for each input frame we need to find the corresponding LexInfo frame. Whenever this matching is not possible we need to place the input frame in the LexInfo ontology[12]. In our case, moving to LexInfo frames allowed us to improve certain aspects such as bounded prepositions and, more important, to get a better organization of the frame system which is eventually organized as an ontology[13].

In any case, conversion tasks can benefit from already defined conversion templates which can be reused when mapping lexical entries from different languages and sources. Figure 4 shows part of the XSL template used to map PAROLE features to LexInfo ontology.

```
<xsl:when   test="./@value = 'VERB'">
            <xsl:text>lexinfo:partOfSpeech lexinfo:VerbPOS </xsl:text>
</xsl:when>
 <xsl:when   test="./@value = 'PASSIVE'">
             <xsl:text> lexinfo:voice lexinfo:passiveVoice</xsl:text>
</xsl:when>
 <xsl:when   test="./@value = '1'">
             <xsl:text> lexinfo:person lexinfo:firstPerson</xsl:text>
</xsl:when>
 <xsl:when   test="./@value = 'GCOMMON'">
            <xsl:text> lexinfo:gender lexinfo:commonGender<xsl:text>
</xsl:when>
<xsl:when   test="./@value = 'SINGULAR'">
            <xsl:text> lexinfo:number lexinfo:singular</xsl:text>
</xsl:when>
```

[9] Notice that we moved from a layered model onto an 'integrated' one and from an ER model formalized as a DTD into an ontology.

[10] Finding the right correspondences between PAROLE/LexInfo 'vocabularies' was rather tough and not always easy due to the lack of documentation in the original lexicons. Note for instance that the LexInfo model includes up to 319 morphosyntactic features.

[11] Notice that PAROLE model allowed for both coarse-grained and fine-grained descriptions.

[12] Our lexicons include lots of pronominal and sentential frames (used to distinguish between indicative and subjunctive complements) that were not listed in the LexInfo ontology.

[13] Note for instance that getting all transitive verbs in the original PAROLE lexicons meant searching for each and every transitive frame.

Figure 4 ‘Mapping features’

3. Some benefits: syntax/semantic linking

The Lemon model simplifies the original PAROLE/SIMPLE model in a good number of aspects. This is partly due to the use of RDF which allows for a more compact and efficient representation. The case of syntax/semantic mappings is particularly interesting. The original PAROLE/SIMPLE data include a complex machinery to define syntactic subcategorization frames and semantic argument structures. In the former case, we have to deal with a large set of related elements: *SynU*, *Description*, *Construction*, *Self*, *InstantiatedPositionC*, *PositionC*, *SyntagmaNT*, etc. The relation among these elements is established by means of the parent/child relation mechanism or ID/IDREF pointing mechanism as exemplified in Figure 5.

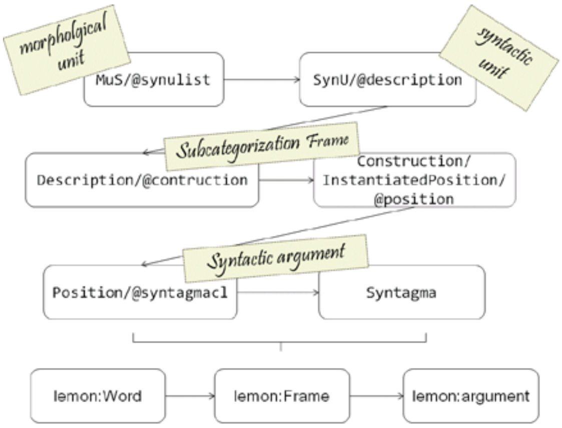

Figure 5 ‘Subcategorization information’

Similarly, argument structure representation is also complex and, again, we find a good number of elements involved: *PredicativeRepresentation*, *Predicate*, *Argument*, *InfArg*, *SemanticRole*, etc.

Syntax semantic linking in the PAROLE/SIMPLE model is even more complex and, in most cases, useless. Syntactic frame descriptions and semantic predicate descriptions are completely separated. The former involve syntactic arguments whereas the latter involve semantic arguments with no relation at all between them. Syntax/semantic relations are expressed by means of additional elements: the *Correspondence* element and its ‘descendants’. *Correspondence* are global elements that point to *SimpleCorrespArgPos* elements which are the eventual holders of the syn/sem argument linking. Since *SimpleCorrespArgPos* elements are global, the linking is defined not in terms of arguments IDs but in terms of the position they occupy in the syntactic frame and the semantic predicate. Note in addition (see Figure 6) that neither the syntactic frame nor the predicate involved are at hand.

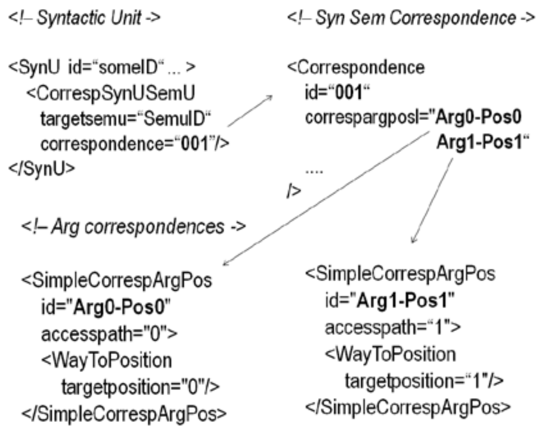

Figure 6 ‘Syn / sem linking in PAROLE/SIMPLE’

The lemon model allows defining all these things in a much easier way, essentially:

Description, Construction & Self elements are mapped to *lemon:Frame* class and related onto the relevant entry by means of the *lemon:synBehaviour* property.

InstantiatedPositionC, Position & Syntagmas are mapped onto *lemon:Argument* class and related to the relevant *lemon:Frame* via some *lemon:synArg* relation.

PredicativeRepresentation & Predicate are also mapped onto *lemon:Frame*

Argument, SemanticRole & InfArg come *lemon:Argument* class and link to vant *lemon:Frame* via some *lemon:semArg* relation.

A simplified entry for the English verb 'write' can be found in Figure 7. Figure 8 gives a partial graphical representation. There we can see that both the syntactic frame and the lexical sense point to ARG0 and ARG1 instances. In the former case, the frame links to its arguments by means of *subject* and *object* properties. In the latter case, the lexical sense links to its arguments by means of *agent* and *patient* properties. Finally, arguments are also specified for a semantic template (Human & SemioticArtifact respectively) and syntactic realization (NP in both cases).

```
## Lexical Entry
lex:write  a  lemon:LexicalEntry ;
        lexinfo:partOfSpeech   lexinfo:mainVerb ;
        lemon:synBehavior   lex:write_transitive;
        lemon:sense   lex:write_SymbolicCreation .

## Lexical Senses
lex:write_SymbolicCreation  a lemon:LexicalSense ;
        parole:template   parole:SymbolicCreation ;
        parole:roleAgent   lex:write_ARGo ;
        parole:rolePatient   lex:write_ARG1 .

## Frames
lex:write_transitive  a  lex:Transitive ;
        lexinfo:subject  lex:write_ARGo;
        lexinfo:directObject   lex:write_ARG1 .
## Argument info
lex:write_ARGo  a   lexinfo:Subject ;
        lemon:constituent lex:NP
        parole:template   parole:Human .
lex:write_ARG1  a   lexinfo:DirectObject ;
        lemon:constituent lex:NP
        parole:template   parole:SemioticArctifact .
```

Figure 7 'A simplified entry for the English verb write'

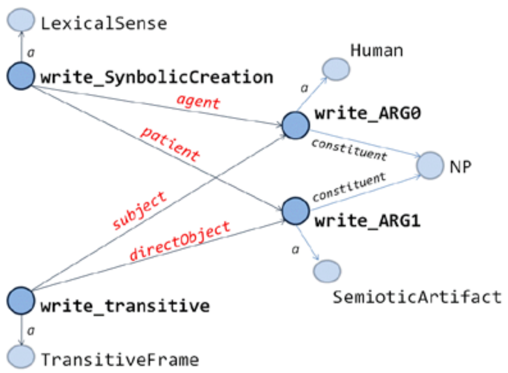

Figure 8 'Simplified Syn/Sem linking'

4. Some benefits: subcategorization frames

Each original PAROLE lexicon defines the set of subcategorization frames for a particular language. Contrary to semantic descriptions, syntactic descriptions are essentially language dependent. Thus whereas all lexicons share the same set of semantic de-scriptive elements (domain, semantic class, semantic relations, etc) such homogeneity was not defined in the syntactic layer. This means that subcategorization information cannot be easily shared among the lex-icons. Basically, this is due to the fact that PAROLE aimed at being a flexible model to accommodate dif-ferent approaches. This might be welcome but proves problematic when addressing interoperability among resources and prevented us from providing a general frame ontology[14] that fulfills the requirements of the different PAROLE lexicons[15]. Instead, we included frame descriptions as part of each language lexicon. Note, however, that the fact that these 'language dependant' frames are eventually integrated into the LexInfo ontology guarantees some interoperability as they all share a core model.

LexInfo defines the subcategorization ontology as the instantiation the lemon classes. As we saw, lemon includes the notion of *Frame*. Frames are indicated with the *synBehaviour* property and their arguments with the property *synArg*. LexInfo defines subproperties of *synArg* to represent the syntactic functions of arguments and organizes frames into subclasses. Our mapping to LexInfo implied mapping PAROLE subcategorization frames onto this model (*Description* elements and their 'descendants'). The mapping process was done in two steps. First, we defined a style sheet converter that reads our PAROLE XML lexicon and for each *Description* element generates a new *Frame*. Consequently, all newly created frames were treated as subclasses of the eral *lemon:Frame*. Second, we collapsed some frames into one single class[16], thus simplifying the model, and organized them in the LexInfo ontology. As a result, the PAROLE lexicons become lighter than the original ones; are better organized; share a wide core frame ontology and allow queries that were otherwise impossible in the original PAROLE lexicons; for instance we can easily get all 'control' verbs; verbs with a sentential complement; verbs with an indirect object, etc[17].

---

[14] as an extension of the LexInfo one

[15] Differences between lexicons are important. For example, in the Spanish lexicon optionality of arguments is dealt at the argument level (optional complements are marked as optional) whereas in the Catalan lexicon optionality of an argument generates two distinct frames; Spanish frames include passive constructions whereas Catalan frames do not; etc.

[16] For example, the original Spanish lexicon includes 12 intransitive prepositional *Descriptions*, one for each bounded preposition. All these frames are mapped to IntransitivePP Frame as the information about the preposition is encoded by means of a property attached to the PP argument.

[17] SPARQL query to retrieve verbs with a sentential argument (with inference): SELECT DISTINCT ?label
WHERE { ?entry lemon:synBehavior ?frame ; rdfs:label ?label.
?frame lemon:synArg ?arg .
?arg lemon:constituent parole:Clause.}

5. Some benefits: exploitation

The most difficult problem of the original PAROLE/SIMPLE lexicons is exploitation and management. When moving from the original PAROLE/SIMPLE model to a relational database, we end up with a complex database with a huge number of related tables[18]. Having PAROLE/SIMPLE lexicons in a database means managing lots of tables and very often we need to split complex queries into several sub queries [7]. Note, for example, that getting the senses of a given lemma is not easy and we need a complex SQL query involving up to six different tables. Similarly, a query such as “find the lemma and template of all senses with a negative connotation” is a real challenge in the original PAROLE/SIMPLE lexica. Such a query is quite simple in RDF as shown in Figure 9. The results are given in Figure 10.

```
SELECT ?label ?template  WHERE {
   ?entry rdfs:label ?label .
   ?entry lemon:sense ?sense.
    ?sense  parole:connotation parole:Negative .
    parole:template ?template.
    }
```

Figure 9 ’SPARQL sample query’

[label]	template
aburrimiento	parole:TemplExperienceEvent
aburrir	parole:TemplExperienceEvent
accidente	parole:TemplEvent
adversidad	parole:TemplQuality
alarma	parole:TemplAgentive
alarma	parole:TemplExperienceEvent
alarmar	parole:TemplExperienceEvent
amargar	parole:TemplExperienceEvent
amargo	parole:TemplADJExtensional
amargo	parole:TemplADJPhysicalProperty
amargura	parole:TemplExperienceEvent
amargura	parole:TemplAgentive
amargura	parole:TemplQuality
ambigüedad	parole:TemplQuality
animal	parole:TemplHuman
antiestético	parole:TemplADJPhysicalProperty
antihigiénico	parole:TemplADJPhysicalProperty
antipático	parole:TemplADJPsychologicalProperty
apatía	parole:TemplPsychproperty
apestoso	parole:TemplADJPhysicalProperty
apurar	parole:TemplExperienceEvent
apático	parole:TemplADJPsychologicalProperty
arbitrariedad	parole:TemplQuality
asno	parole:TemplHuman
asquerosidad	parole:TemplQuality
asqueroso	parole:TemplADJPhysicalProperty
asustar	parole:TemplExperienceEvent

Figure 10 ‘Query results’

[18] Our PAROLE/SIMPLE database included 223 tables.

6. The sources

The Ontology and both the Spanish and Catalan lexicons are distributed under CC Attribution 3.0 Unported license. These datasets are published in the Data Hub (http://datahub.io/dataset/parole-simple-ont) and can be downloaded in both XML RDF format and RDF Turtle format.

The Spanish lexicon contains 199,466 triples with 7,572 lexical entries fully annotated with syntactic and semantic information distributed as follows: 5,659 common nouns, 729 proper nouns, 859 adjectives and 325 verbs. The lexicon contains 11,430 LexicalSenses.

The Catalan lexicon contains 343,714 triples and 20,545 lexical entries annotated with syntactic information half of which are also annotated with semantic information. Lexical entries include 3,064 verbs, 13,206 common nous, 247 proper nous, 3,101 adjectives and 511 adverbs. The rest belong to closed categories. The lexicon contains 11,813 LexicalSenses.

Table 1 lists the properties assigned to LexicalSenses[19] in both lexicons.

Property	Spanish	Catalan
id	11430	11813
template	9924	10782
example	9727	10443
semanticClass	8987	10027
semanticRelation	15808	23835
countability	6827	5573
semanticFeature	3222	4328
semanticRole	2294	4381
copulaType	971	
connotation	1314	1364
adjType	979	715
comment	1506	8388
domain	107	56
gradable	246	
definition		10658
TOTAL	73342	102363

Table 1 ‘Triples assigned to LexicalSense’

7. Summary and Conclusions

The dataset described here is the result of mapping PAROLE/SIMPLE Spanish and Catalan lexicons onto the lemon model following the LexInfo ontology. The mapping implied three main tasks: the lexicon format mapping (from DTD to lemon model), the

[19] Semantic relations and semantic roles are grouped. The object of ‘semantic relation’ triples is always another LexicalSense.

ontology mapping (from ‘descriptive’ elements to LexInfo ontology) and the mapping of lexical entries.

This work may help and encourage other PAROLE/SIMPLE lexicons to take the same way[20]. The lemon version of PAROLE model (DTD) is already mapped and all shared descriptive elements are integrated with the LexInfo ontology. Everything can be reused by other languages. In addition, new lexicons can benefit from conversion templates and only need to address language particular descriptions. Linked Open Data is the natural scenario for a multilingual resource such as the PAROLE/SIMPLE lexicons.

The resulting lexicons benefit from standardization and Linked Open Data principles. The fact that source data categories are mapped onto the LexInfo ontology which in turn is linked to ISOcat[21] is a step forward in standardization and interoperability.

From our experience we conclude that XML (essentially DTDs) is not well suited for modeling purposes as it allows for a number of syntactic alternatives and conveys semantic ambiguity. In addition, XML proves inefficient when relating resources. This is crucial in a scenario where references to external resources are essential to guarantee interoperability. RDF overcomes some of the problems met with XML. The use of RDF (especially URIs) proves essential for datasets such as original PAROLE/SIMPLE lexicons where interoperability was crucial. The resulting data is lighter and better suited for exploitation. In addition, it easies further extensions and links with external resources such as WordNet, lemonUby, DBpedia etc.

8. Acknowledgements

The resources reported in this paper were developed with the support of METANET4U: Enhancing the European Linguistic Infrastructure, (2011-2013), funded by UNER - Competitiveness and Innovation Framework Program, (CIP-PSP-270893).

We thank the Institut d’Estudis Catalans and the GilcUB from the University of Barcelona as the creators of the original Catalan and Spanish lexicons.

Finally, we also want to thank the reviewers of this paper for their valuable comments and suggestions.

[20] Most PAROLE/SIMPLE lexicons have been maintained and enlarged since their creation and are available at the META-SHARE nodes (ie http://metashare.elda.org/). These include the Catalan, Danish, Dutch, English, Greek, Italian, Portuguese and Spanish. The Swedish lexicon can be accessed at http://spraakdata.gu.se/parole/lexikon/swedish.parole.lexikon.html.

[21] http://www.isocat.org/

References

[1] Antoni-Lay, M.-H., Francopoulo, G., & Zayssr, L. (1994). A generic model for reusable lexicons: the GENELEX project. *Literary and linguistic computing*, 9(1), 47-54.

[2] Buitelaar, P., Cimiano, P., Haase, P., & Sintek, M. (2009). In *Proceedings of the 6th European Semantic Web Conference on The Semantic Web: Research and Applications* (pp. 111–125). Berlin, Heidelberg: Springer-Verlag. doi:10.1007/978-3-642-02121-3_12

[3] Chiarcos, Ch., McCrae, J., Cimiano, P., & Fellbaum, Ch. (2013). Towards open data for linguistics: Lexical Linked Data. In A. Oltramari, P. Vossen, L. Qin, & E. Hovy (Eds.), *New Trends of Research in Ontologies and Lexical Resources: Ideas, Projects, Systems* (pp. 7–25). Berlin, Heildelberg: Springer-Verlag.

[4] Cimiano, P., Buitelaar, P., McCrae, J., & Sintek, M. (2011). LexInfo: A declarative model for the lexicon-ontology interface. *Web Semantics: Science, Services and Agents On The World Wide Web*, 9(1), 29-51. doi:http://dx.doi.org/10.1016/j.websem.2010.11.001

[5] Francopoulo, G., Bel, N., George, M., Calzolari, N., Monachini, M., Pet, M., & Soria, C. (2007). Lexical Markup Framework: ISO standard for semantic information in NLP lexicons. In *Lexical semantic and ontological resources: Workshop of the GLDV Working Group on Lexicography at the Biennial Spring Conference of the GLDV; April 13-14, 2007, Tubingen, Germany; workshop proceedings*. Hagen: FernUniv. Retrieved from http://tagmatica.fr//publications/LMFPaperForTubingen17February2007.pdf

[6] McCrae, J., Spohr, D., & Cimiano, P. (2011). Linking Lexical Resources and Ontologies on the Semantic Web with Lemon. In *Proceedings of the 8th Extended Semantic Web Conference on The Semantic Web: Research and Applications - Volume Part I* (pp. 245–259). Berlin, Heidelberg: Springer-Verlag. Retrieved from http://dl.acm.org/citation.cfm?id=2008892.2008914

[7] Barnett, R., Calzolari, N., Flores, S., Hellwig, P., Kahrel, P., Leech, G., Melera, M., Montemagni, S., Odijk, J., Pirrelli, V., Sanfilippo, A., Teufel, S., Villegas, M., & Zaysser, L. (1996). *EAGLES: Preliminary Recommendations on Subcategorisation: EAG---CLWG---SYNLEX/P, version of Aug, 1996*. [Pisa]: Eagles. Retrieved from http://www.ilc.cnr.it/EAGLES96/synlex/synlex.html

[8] Villegas, M., & Bel, N. (2002). From DTD to relational dB. An automatic generation of a lexicographical station out off ISLE guidelines. In *Proceedings of the Third International Conference on Language Resources and Evaluation, LREC 2002* (pp. 694-700). Las Palmas, Canary Islands, Spain: European Language Resources Association. Retrieved from http://www.lrec-conf.org/proceedings/lrec2002/pdf/48.pdf