LIRICS

Deliverable D2.3

Test Suite of ISO-conformant lexical entries

Project reference number	e-Content-22236-LIRICS
Project acronym	LIRICS
Project full title	Linguistic Infrastructure for Interoperable Resource and Systems
Project contact point	Laurent Romary, INRIA-Loria 615, rue du jardin botanique BP101. 54602 Villers lès Nancy (France) romary@loria.fr
Project web site	http://lirics.loria.fr
EC project officer	Erwin Valentini
Document title	Lexical Test Suites
Deliverable ID	D2.3
Document type	Report
Dissemination level	Public
Contractual date of delivery	M27
Actual date of delivery	17 September 2007
Status & version	Draft, version 1.6
Work package, task & deliverable responsible	CNR-ILC
Author(s) & affiliation(s)	Monica Monachini CNR-ILC
Additional contributor(s)	Gil Francopoulo INRIA, Antonio Toral CNR-ILC, Nuria Bel and Sergio Espeja, UPF, Marc Kemps-Snijders, Max Plank
Keywords	NLP Lexicon, lexical information, standard, XML

Document evolution

version	date	version	date
1.1	30 June 2006		
1.2	15 September 2007		

Table of Content

1

Introduction

One of the aims of LIRICS is the development of test suites, i.e. a set of resources in the form of practical examples associated to the international standards produced by the project to test the applicability and usability of the proposed concepts. The objectives of developing test suites in conjunction with the delivery of a standard are to provide a guide for those who wish to apply them to their resources and, above all, to test their viability in NLP implementations and systems. Test suites will accompany the standards, ensure both wide dissemination and demonstration, facilitate their implementation and capability of propagation during and after project life cycle. Finally, the development of test suites will allow implementers to combine a given standard proposal in the form of a meta-model with the relevant Data Categories taken from the Registry. They can thus be used as examples of the application of data categories themselves. Test suites also act as a reference to the best practices in the representation of those phenomena.

Test suites are designed according to a methodology based on the same principles specifications:

- *relevance* of linguistic concepts,
- *conciseness* with regards to the number of actual entries and
- *precision* in relation to what is actually represented in each entry of the test suite
- *conformity* to TC 37/SC4 design principles.

As a consequence of this, they are not intended to provide a large amount of cases, but should rather focus on the quality and relevance of the examples they provide.

This current deliverable constitutes the last step in the definition of a standard framework for NLP lexicons produced in Work Package 2 of the LIRICS project.

The first, D2.1 “Evaluation of existing standards for NLP lexica” (Monachini et al., 2005) gathered a set of linguistic information reliable and harmonized enough to be recognized as crucial for the description of a computational lexical entry; then proposed a draft inventory of lexical information as good candidates to become Data Categories with unified descriptors, short descriptions and exemplifications. A set of about 500 data categories taken form D2.1 and D3.1, indeed, is part of the ISO Data Category Registry in the Morphosyntactic Profile. Other data categories for other profiles will be continuously integrated in the ISO Registry as the contribution of CNR-ILC to standardization activities undertaken at the level of UNI (Italy) and ISO TC37/SC4. CNR-ILC is currently conducting corresponding work in Asia within the NEDO project and for the specialized bio-medical domain within the BOOTStrep project (see below).

The second set of deliverables, D2.2a (Monachini et al 2006) and D2.2b (Francopoulo and Monachini 2007) presented two different releases of the meta-model for lexicon description, the Lexical Mark-up Framework. The first one describes revision 9, the WD for CD ballot, finalized and submitted for balloting on March 2006. The ISO National delegations expressed their positive judgement on June 2006. The second deliverable, on the basis of the feedback emerged during the ballot phase, contains the CD standard for the DIS ballot presented on November 2006, revision 14, which also obtained a positive vote at the end of February 2007. On June 2007, a DIS version for the FDIS ballot has been produced and submitted on August 2007.

This present deliverable is a report on the development of lexical test suites (Task2.3). Each of the LIRICS partners took care of the construction of test suites in their respective languages by converting lexical entries extracted from their lexicons to the proposed format, the Lexical Markup Framework developed in Task 2.2. These test suites accompany the LMF standard, facilitating its acceptance and implementation and promoting the development of LMF conformant lexicons. Lexical test suites cover the following languages: Italian, French, Spanish and English.

The LMF DTD, which has guided the implementation of the LIRICS test suites, is provided in Appendix A.

It should also be taken into consideration that CNR-ILC is the major promoter of LMF in two on-going international projects. The first project is BOOTStrep (www.boostrep.com), where LMF is the basis for the development of the model of a large-scale lexico-terminological resource the BioLexicon (Monachini et al 2007; Quochi et al 2007) especially designed for text-mining applications in the biomedical domain. This resource presents some novelties: one of the most important is that it is the first resource in the field to be compliant to ISO standards. Thanks to conformity to standards, it accounts for interoperability and extendibility to other areas. Relationships and reciprocal impacts between the two lexical models, the ISO-LIRICS and the BioLexicon ones, go in two directions: the ISO-LIRICS model strongly influences the architecture and the policies of the BioLexicon model, but, vice-versa, the BioLexicon model constitutes both an extension and an implementation of the available standards, thus enhancing the lexical standards themselves. LMF plays a central role in the Japanese grant NEDO (Tokunaga et al 2006). This is a project for pushing European lexical standards in Asia and developing harmonized lexical resources for Asian languages. This LMF-compliant NEDO lexicon is expected to support cross-language information retrieval applications, developed by an Asian industrial partner in view of the Olympics, Beijing 08.

In the present deliverable, in order to show the degree of acceptance and the percolation power of LMF, some sample entries from those projects are included in the lexical test suites. The first is a small set of English entries taken from the biomedical domain, in particular, the gene regulation sub-domain. The other set is provided to show the application of LMF to Asian languages. The NEDO lexicon is totally compliant to the LMF DTD, whereas, for the BioLexicon lexical entries, Appendix B provides the relevant DTD, that can be seen as a "dialect" of the LMF DTD.

By the TA, the beginning of the activities of test suite building was scheduled at M18, that is mid project. Since the lexical standard was still provisional (revision 9, CD ballot phase), it was decided to postpone the construction of test suites to a later stage with a stable LMF version and to have instead a first proof-of-concept by translating into LMF some lexical entries taken from best practices. As already experimented in different standardization activities (Bertagna et al. 2004), mapping what exists to a standard has always a positive impact, thus helping to show the potentialities and capability of the standard itself. Examples were provided in both UML and XML format and when available, raw data were presented: the results can be found in the ISO/LIRICS auxiliary working document section http://lirics.loria.fr/doc_pub/.

The current document is organized as follows:

Section 2 and 3 present the Italian test suite and some sample LMF conformant entries.

Section 4 and 5 contain a brief description of the Spanish test suite and sample LMF conformant entries.

Section 6 and 7 are devoted to the French LMF test suite and relevant entries.

Section 8 and 9 present the English test suite used as a demo of the LEXUS tool developed in the framework of LIRICS to describe LMF complaint lexicons.

Section 10 and 11 introduce the NEDO lexicon, a resource totally conformant to LMF developed for Asian languages and provide sample entries for Japanese.

Section 12 and 13 contain the BioLexicon, a LMF implementation for the specialized biomedical domain.

The full xml file of LMF test suites is made available online on the LIRICS web site.

2 Italian Test Suite

In this section, the LMF conformant lexical entry samples have been selected from the PAROLE-SIMPLE-CLIPS lexicon (Ruimy et al. 2003).

This lexicon is a multipurpose computational lexical database for Italian, obtained by extending the PAROLE-SIMPLE lexicon which shares with other eleven European lexica a common conceptual model, representation language and lexicon building methodology. The underlying theoretical model is grounded on the EAGLES project recommendations and, at semantic level, it implements and extends major aspects of Generative Lexicon (GL) theory; nevertheless, the lexicon is not strictly theory-dependent. The model enables a very fine-grained description to be performed, but allows a more shallow one too, in so far as the information provided meets the model requirements.

The lexicon consits of 53000 lemmas encoded at morphological level and phonological level (for a total of about 390000 word-forms), 51000 lemmas encoded at syntactic level, and 57000 semantically encoded word senses. Conformity of the data to the model is ensured by an XML DTD, whereas internal formal validation is performed by an XML parser.

In order to export entries from the PAROLE-SIMPLE-CLIPS Italian lexicon into LMF, we have used an API which was developed for the first. This API allows to perform queries to the lexical database from Java applications. This way, for each element of the entries to be exported, we apply queries from this API and encode it according to the LMF syntax.

As the database model is the same for the 12 lexicons (each for a different language) developed within the european project SIMPLE, the introduced procedures could be used out-of-the-box for SIMPLE lexicons for languages other than Italian.

The Italian test suite amounts to 100 entries. Entries are taken from noun, verb and adjective syntactic categories and are intended to implement (part of) the syntactic extension of LMF but, particularly, the semantic extesion with focus on Sense, SenseRelation, PredicativeRepresentation, SemanticPredicate objects of the meta-model. Also, the implementation of the level of correspondence between syntax and semantics is provided.

The LMF output entryset has been checked with xmlstarlet (see reference). Two types of validation have been performed: well-formedness and validation against DTD. While the well-formedness validation succeeds, the validation against the LMF DTD fails. However, all the errors that cause the failure of the validation are of the same type: "IDREFS attribute targets references an unknown ID". This is due to the fact that the 100 entries do not constitute a closed repository. Some entries from the entryset contain references to other entries not present in the entryset. Therefore, we can conclude that the entryset is compliant to the DTD but the validation fails because of the presence of references that are not present in the 100 entryset.

3 Examples of Italian LMF conformant lexical entries

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE LexicalResource SYSTEM "LMFNLP.dtd">
<LexicalResource dtdVersion="14">
<GlobalInformation>
         <feat att="label" val="ILC-CNR test suites number 1 for Italian"/>
</GlobalInformation>
<Lexicon>
         <feat att="language" val="Italian"/>
         <LexicalEntry id="LE_abbandonare">
                  <feat att="POS" val="V"/>
                  <Lemma>
                           <feat att="writtenform" val="abbandonare"/>
                  </Lemma>
                  <Sense id="USem73115abbandonare">
                           <PredicativeRepresentation predicate="PREDabbandonare_2"
correspondences="ISObivalent">
```

```
                                        <feat att="link" val="Master"/>
                                </PredicativeRepresentation>
                                <SenseRelation targets=" USem67171mollare
USem79703lasciare">
                                        <feat att="relation_type" val="Synonym"/>
                                </SenseRelation>
                                <SenseRelation targets=" USemD5371agire">
                                        <feat att="relation_type" val="Isa"/>
                                </SenseRelation>
                        </Sense>
                        <SyntacticBehaviour id="SB_SYNUabbandonareV" senses="
USem59592abbandonare USem73115abbandonare"/>
                </LexicalEntry>
                <SemanticPredicate id="PREDabbandonare_2">
                        <SemanticArgument id="ARG0abbandonare_2">
                                <feat att="role" val="Role_ProtoAgent"/>
                                <feat att="restriction_type" val="Notion"/>
                                <feat att="restriction" val="ArgHuman"/>
                        </SemanticArgument>
                        <SemanticArgument id="ARG1abbandonare_2">
                                <feat att="role" val="Role_ProtoPatient"/>
                                <feat att="restriction_type" val="SemType"/>
                                <feat att="restriction" val="Entity"/>
                        </SemanticArgument>
                </SemanticPredicate>
                <SynSemCorrespondence id="ISObivalent">
                        <SynSemArgMap synFeature="pos0" semFeature="arg0"/>
                        <SynSemArgMap synFeature="pos1" semFeature="arg1"/>
                </SynSemCorrespondence>
</Lexicon>
</LexicalResource>
```

4 Spanish Test Suite

The Spanish samples have been taken from the “Diccionari de processament del Corpus Tècnic de l’IULA”, a full form dictionary of the Institut Universitari de Lingüística Aplicada, for tagging the “Corpus Tècnic de l’IULA”. The full form dictionary contains about 80K lexical entries and has been migrated to LMF and stored in a LMF based database. The database allows a web service access, which is LMF compliant.

The lexical test suite is composed of about 100 different lexical entries, covering different parts of speech, and encoding morphosyntactic features.

5 Examples of Spanish LMF conformant lexical entries

```
<LexicalEntry>
          <feat att="partOfSpeech" val="commonNoun" />
          <Lemma>
                    <feat att="writtenForm" val="niaja"/>
          </Lemma>
          # niajas N5-FP
          <WordForm>
                    <feat att="writtenForm" val="niajas"/>
                    <feat att="grammaticalGender" val="feminine" />
                    <feat att="grammaticalNumber" val="plural" />
          </WordForm>
</LexicalEntry>

<LexicalEntry>
          <feat att="partOfSpeech" val="verb" />
          <Lemma>
                    <feat att="writtenForm" val="dictar"/>
          </Lemma>
          # dictabais VDA2P-
          <WordForm>
                    <feat att="writtenForm" val="dictabais"/>
                    <feat att="verbFormMood" val="indicative" />
                    <feat att="grammaticalTense" val="imperfect" />
                    <feat att="person" val="secondPerson" />
                    <feat att="grammaticalNumber" val="plural" />
          </WordForm>
</LexicalEntry>

<LexicalEntry>
          <feat att="partOfSpeech" val="verb" />
          <Lemma>
                    <feat att="writtenForm" val="asestar"/>
          </Lemma>
          # asestaría VDC6S-
          <WordForm>
                    <feat att="writtenForm" val="asestaría"/>
                    <feat att="verbFormMood" val="indicative" />
                    <feat att="grammaticalTense" val="conditional" />
                    <feat att="grammaticalNumber" val="singular" />
          </WordForm>
</LexicalEntry>

<LexicalEntry>
          <feat att="partOfSpeech" val="qualifierAdjective" />
          <Lemma>
                    <feat att="writtenForm" val="horripilador"/>
          </Lemma>
          # horripiladores JQ--MP
          <WordForm>
                    <feat att="writtenForm" val="horripiladores"/>
                    <feat att="grammaticalGender" val="masculine" />
                    <feat att="grammaticalNumber" val="plural" />
          </WordForm>
</LexicalEntry>

<LexicalEntry>
```

```
          <feat att="partOfSpeech" val="verb" />
          <Lemma>
                    <feat att="writtenForm" val="chirlar"/>
          </Lemma>
          # chirlase VJA6S-
          <WordForm>
                    <feat att="writtenForm" val="chirlase"/>
                    <feat att="verbFormMood" val="subjunctive" />
                    <feat att="grammaticalTense" val="imperfect" />
                    <feat att="grammaticalNumber" val="singular" />
          </WordForm>
</LexicalEntry>

<LexicalEntry>
          <feat att="partOfSpeech" val="qualifierAdjective" />
          <Lemma>
                    <feat att="writtenForm" val="pataruco"/>
          </Lemma>
          # patarucas JQ--FP
          <WordForm>
                    <feat att="writtenForm" val="patarucas"/>
                    <feat att="grammaticalGender" val="feminine" />
                    <feat att="grammaticalNumber" val="plural" />
          </WordForm>
</LexicalEntry>

<LexicalEntry>
          <feat att="partOfSpeech" val="verb" />
          <Lemma>
                    <feat att="writtenForm" val="peligrar"/>
          </Lemma>
          # peligrase VJA6S-
          <WordForm>
                    <feat att="writtenForm" val="peligrase"/>
                    <feat att="verbFormMood" val="subjunctive" />
                    <feat att="grammaticalTense" val="imperfect" />
                    <feat att="grammaticalNumber" val="singular" />
          </WordForm>
</LexicalEntry>

<LexicalEntry>
          <feat att="partOfSpeech" val="verb" />
          <Lemma>
                    <feat att="writtenForm" val="vendar"/>
          </Lemma>
          # vendármelo VI----_t
          <WordForm>
                    <feat att="writtenForm" val="vendármelo"/>
                    <feat att="verbFormMood" val="infinitive" />
          </WordForm>
</LexicalEntry>

<LexicalEntry>
          <feat att="partOfSpeech" val="verb" />
          <Lemma>
                    <feat att="writtenForm" val="enraigonar"/>
          </Lemma>
          # enraigonando VG----
          <WordForm>
                    <feat att="writtenForm" val="enraigonando"/>
                    <feat att="verbFormMood" val="gerund" />
          </WordForm>
</LexicalEntry>

<LexicalEntry>
          <feat att="partOfSpeech" val="verb" />
          <Lemma>
                    <feat att="writtenForm" val="titular"/>
          </Lemma>
          # titulármeles VI----_t
          <WordForm>
                    <feat att="writtenForm" val="titulármeles"/>
                    <feat att="verbFormMood" val="infinitive" />
          </WordForm>
</LexicalEntry>
```

6 French Test Suite

LMF test suites for French are intended, on the one hand, to show applicability of the model to simple adjectives and, on the other hand, to focus on the extension for MultiWords representation.

7 Examples of French LMF conformant lexical entries

```
<LexicalResource dtdVersion="14">
    <GlobalInformation
        <feat att="label"          val="LIRICS test suites number 1 for French"/>
        <feat att="comment"        val="Two adjectives are described: actif and
inactif. Each of them has two meanings. One of their meanings are linked together by
an antonymy. These two adjectives are morphologically described by the same paradigm
pattern. These adjectives are described in syntax. And the lexicon states by a set of
constraints that in French, adjectives vary in gender and number."/>
        <feat att="author"         val="Gil Francopoulo"/>
        <feat att="languageCoding" val="ISO 639-3"/>
    </GlobalInformation>
        <!--############################################################core section -->
    <Lexicon>
        <feat att="language" val="fra"/>
        <LexicalEntry paradigmPatterns="AsPassif">
            <feat att="partOfSpeech" val="adjective"/>
            <Lemma>
                 <feat att="writtenForm" val="actif"/>
            </Lemma>
            <Sense id="S1">
                 <feat att="definition" val="Qui agit ou implique une activité"/>
                 <SenseRelation targets="S3">
                     <feat att="label" val="antonym"/>
                     <feat att="comment" val="Actif est le contraire d'inactif"/>
                 </SenseRelation>
            </Sense>
            <Sense id="S2">
                 <feat att="definition" val="Propre à exprimer que le sujet est
considéré comme agissant"/>
                 <feat att="domain"     val="grammaire"/>
            </Sense>
            <SyntacticBehaviour subcategorizationFrameSets="SSRegPostAdj"/>
        </LexicalEntry>
        <LexicalEntry paradigmPatterns="AsPassif">
            <feat att="partOfSpeech" val="adjective"/>
            <Lemma>
                 <feat att="writtenForm" val="inactif"/>
            </Lemma>
            <Sense id="S3">
                 <feat att="definition" val="Qui n'a pas d'activité"/>
            </Sense>
            <Sense id="S4">
                 <feat att="definition" val="Qui n'a pas d'activité régulière, sans
être chômeur"/>
                 <feat att="domain"     val="juridique"/>
            </Sense>
            <SyntacticBehaviour subcategorizationFrameSets="SSRegPostAdj">
            </SyntacticBehaviour>
        </LexicalEntry>
        <!--############################################################syntax -->
        <SubcategorizationFrameSet id="SSRegPostAdj"
subcategorizationFrames="SFPostAdj SFAttr">
            <feat att="comment" val="The adjective may be used after the noun or as an
attribute"/>
        </SubcategorizationFrameSet>
        <SubcategorizationFrame id="SFPostAdj">
            <feat att="syntacticConstituent" val="NP"/>
            <feat att="comment"              val="The adjective can be used only after
the noun"/>
            <LexemeProperty>
                 <feat att="position" val="1"/>
                 <feat att="comment"  val"The current lexeme is after the noun"/>
            </LexemeProperty>
```

```
            <SyntacticArgument>
                 <feat att="partOfSpeech" val="noun"/>
            </SyntacticArgument>
        </SubcategorizationFrame>
        <SubcategorizationFrame id="SFAttr">
            <feat att="syntacticConstituent" val="VP"/>
            <feat att="comment"              val="Attributive formulation"/>
            <LexemeProperty>
                 <feat att="position" val="1"/>
                 <feat att="comment"  val"The current lexeme is after the verb"/>
            </LexemeProperty>
            <SyntacticArgument>
                 <feat att="partOfSpeech" val="verb"/>
            </SyntacticArgument>
        </SubcategorizationFrame>
        <!--############################################################paradigm patterns -
->
        <ParadigmPattern id="AsPassif">
            <feat att="comment"      val="Intended for adjectives with F ending"/>
            <feat att="partOfSpeech" val="adjective"/>
            <!--four values: the combination of masc/femi and sing/plur -->
            <TransformSet>
                <!-- masc/sing, the ending "f" is kept -->
                <Process>
                     <feat att="operator" val="addLemma"/>
                </Process>
                <GrammaticalFeatures>
                     <feat att="grammaticalGender" val="masculine"/>
                     <feat att="grammaticalNumber" val="singular"/>
                </GrammaticalFeatures>
            </TransFormSet>
            <TransformSet>
                <!-- masc/plur, an "s" is added -->
                <Process>
                     <feat att="operator" val="addLemma"/>
                </Process>
                <Process>
                     <feat att="operator" val="addAfter"/>
                     <feat att="stringValue" val="s"/>
                </Process>
                <GrammaticalFeatures>
                     <feat att="grammaticalGender" val="masculine"/>
                     <feat att="grammaticalNumber" val="plural"/>
                </GrammaticalFeatures>
            </TransformSet>
            <TransformSet>
                <!-- femi/sing, the ending "f" is transformed into "ve"
                <Process>
                     <feat att="operator" val="addLemma"/>
                </Process>
                <Process>
                     <feat att="operator" val="removeAfter"/>
                     <feat att="numValue" val="1"/>
                </Process>
                <Process>
                     <feat att="operator" val="addAfter"/>
                     <feat att="stringValue" val="ve"/>
                </Process>
                <GrammaticalFeatures>
                     <feat att="grammaticalGender" val="feminine"/>
                     <feat att="grammaticalNumber" val="singular"/>
                </GrammaticalFeatures>
            </TransformSet>
            <TransformSet>
                <!-- femi/plur, the ending "f" is transformed into "ves"
                <Process>
                     <feat att="operator" val="addLemma"/>
                </Process>
                <Process>
                     <feat att="operator" val="removeAfter"/>
                     <feat att="numValue" val="1"/>
                </Process>
                <Process>
                     <feat att="operator" val="addAfter"/>
                     <feat att="stringValue" val="ves"/>
                </Process>
```

```
                <GrammaticalFeatures>
                     <feat att="grammaticalGender" val="feminine"/>
                     <feat att="grammaticalNumber" val="plural"/>
                </GrammaticalFeatures>
            </TransFormSet>
        </ParadigmPattern>
        <!--##########################################################constraints -->
        <ConstraintSet>
            <feat att="label" val="grammaticalFeatureVariation"/>
            <Constraint>
                <feat att="label" val="forAdjectives"/>
                <feat att="comment" val="Valid for all French qualifying adjectives"/>
                <LogicalOperation>
                     <feat att="operator" att="logicalAnd"/>
                     <AttributeValuation>
                          <feat att="partOfSpeech" val="adjective"/>
                     </AttributeValuation>
                     <Constraint>
                          <feat att="label" val="genderNumber"/>
                          <LogicalOperation>
                               <feat att="operator" val="logicalOr"/>
                               <AttributeValuation>
                                    <feat att="grammaticalGender" val="masculine"/>
                                    <feat att="grammaticalNumber" val="singular"/>
                               </AttributeValuation>
                               <AttributeValuation>
                                    <feat att="grammaticalGender" val="masculine"/>
                                    <feat att="grammaticalNumber" val="plural"/>
                               </AttributeValuation>
                               <AttributeValuation>
                                    <feat att="grammaticalGender" val="feminine"/>
                                    <feat att="grammaticalNumber" val="singular"/>
                               </AttributeValuation>
                               <AttributeValuation>
                                    <feat att="grammaticalGender" val="feminine"/>
                                    <feat att="grammaticalNumber" val="plural"/>
                               </AttributeValuation>
                          </LogicalOperation>
                     </Constraint>
                </LogicalOperation>
            </Constraint>
        </ConstraintSet>
    </Lexicon>
</LexicalResource>


<LexicalResource dtdVersion="14">
    <GlobalInformation
        <feat att="label"          val="LIRICS test suites number 2 for French"/>
        <feat att="comment"        val="The multiword expression pomme de terre is
descibed"/>
        <feat att="author"         val="Gil Francopoulo"/>
        <feat att="languageCoding" val="ISO 639-3"/>
    </GlobalInformation>
        <!--##########################################################core section -->
    <Lexicon>
        <feat att="language" val="fra"/>
        <LexicalEntry mwePattern="NdeFixedN">
            <feat att="partOfSpeech" val="noun"/>
            <Lemma>
                 <feat att="writtenForm" val="pomme de terre"/>
            </Lemma>
            <ListOfComponents>
                 <Component entry="E1"/>
                 <Component entry="E2"/>
                 <Component entry="E3"/>
            </ListOfComponents>
        </LexicalEntry>
        <LexicalEntry id="E1" paradigmPatterns="AsTable">
            <feat att="partOfSpeech" val="noun"/>
            <Lemma>
                 <feat att="writtenForm" val="pomme"/>
            </Lemma>
        </LexicalEntry>
        <LexicalEntry id="E2" paradigmPatterns="AsDe">
            <feat att="partOfSpeech" val="preposition"/>
```

```
            <Lemma>
                 <feat att="writtenForm" val="de"/>
            </Lemma>
        </LexicalEntry>
        <LexicalEntry id="E3" paradigmPatterns="AsTable">
            <feat att="partOfSpeech" val="noun"/>
            <Lemma>
                 <feat att="writtenForm" val="terre"/>
            </Lemma>
        </LexicalEntry>
        <!--#######################################################paradigm patterns -->
        <ParadigmPattern id="AsTable">
            <feat att="comment"      val="Intended for feminine regular nouns"/>
            <feat att="partOfSpeech" val="noun"/>
            <!--two values: the combination of sing/plur -->
            <TransformSet>
                <!-- sing, the ending is kept -->
                <Process>
                     <feat att="operator" val="addLemma"/>
                </Process>
                <GrammaticalFeatures>
                     <feat att="grammaticalGender" val="feminine"/>
                     <feat att="grammaticalNumber" val="singular"/>
                </GrammaticalFeatures>
            </TransFormSet>
            <TransformSet>
                <!-- plur, an "s" is added -->
                <Process>
                     <feat att="operator" val="addLemma"/>
                </Process>
                <Process>
                     <feat att="operator" val="addAfter"/>
                     <feat att="stringValue" val="s"/>
                </Process>
                <GrammaticalFeatures>
                     <feat att="grammaticalGender" val="feminine"/>
                     <feat att="grammaticalNumber" val="plural"/>
                </GrammaticalFeatures>
            </TransformSet>
        </ParadigmPattern>
        <ParadigmPattern id="AsDe">
            <feat att="comment"      val="For fixed grammatical words"/>
            <feat att="partOfSpeech" val="preposition"/>
        </ParadigmPattern>
        <!--#############################################################MWE patterns -->
        <MWEPattern id="NdeFixedN">
            <MWENode>
                <feat att="syntacticConstituent"    val="NP">
                <MWELex>
                     <feat att="rank"               val="1"/>
                     <feat att="graphicalSeparator" val="space"/>
                     <feat att="structureHead"      val="yes"/>
                <MWELex>
                <MWELex>
                     <feat att="rank"               val="2"/>
                     <feat att="graphicalSeparator" val="space"/>
                <MWELex>
                <MWELex>
                     <feat att="rank"               val="3"/>
                     <feat att="graphicalSeparator" val="space"/>
                     <feat att="grammaticalNumber"  val="singular"/>
                <MWELex>
            </MWENode>
        </MWEPattern>
    </Lexicon>
</LexicalResource>
```

8 English Test Suite

The English test suite consists of a small sample of lexical entries that are used for demo purposes of the LEXUS tool, developed by Max Plank to describe LMF compliant lexicons.

9 Examples of English LMF conformant lexical entries

```
<?xml version="1.0" encoding="iso-8859-1"?>
<lexicon
xmlns="http://lux16.mpi.nl:80/mpi/lexus/schema/bGV4aWNvbi8yYzkwOTBjMjEyZmQzNTcxMDExMzI
wMWNlNGI4MDBmNQ==/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://lux16.mpi.nl:80/mpi/lexus/schema/bGV4aWNvbi8yYzkwOTBjMjEyZm
QzNTcxMDExMzIwMWNlNGI4MDBmNQ==/lexicon.xsd">
    <lexiconInformation>
        <creation_x0020_date>September 14, 2007</creation_x0020_date>
    </lexiconInformation>
    <lexicalEntry>
        <headword>crossroad</headword>
        <partOfSpeech>n</partOfSpeech>
        <sense>
            <definition>an intersection of two or more roads</definition>
            <synonym>junction</synonym>
            <gloss>A place where two or more roads meet</gloss>
        </sense>
        <form>
            <pronunciation>kraws-rohd, kros-</pronunciation>
            <phoneticForm>&#712;kr&#596;s&#716;ro&#650;d, &#712;kr&#594;s-
</phoneticForm>
            <multi_x0020_media>

<image>http://lux16.mpi.nl:80/mpi/lexus/resources/172.16.24.1331189762707438crossroads
.jpg</image>

<sound>http://lux16.mpi.nl:80/mpi/lexus/resources/172.16.24.1331189764416977crossr01.w
av</sound>
            </multi_x0020_media>
        </form>
    </lexicalEntry>
    <lexicalEntry>
        <headword>grid</headword>
        <partOfSpeech>n</partOfSpeech>
        <form>
            <phoneticForm>gr&#618;d</phoneticForm>
            <pronunciation>grid</pronunciation>
            <multi_x0020_media>

<image>http://lux16.mpi.nl:80/mpi/lexus/resources/172.16.24.1331189758819704hermannGri
dNegative.gif</image>

<sound>http://lux16.mpi.nl:80/mpi/lexus/resources/172.16.24.1331189764458976grid0001.w
av</sound>
            </multi_x0020_media>
        </form>
        <sense/>
        <sense>
            <gloss>gridiron</gloss>
            <definition>a network of horizontal and perpendicular lines, uniformly
spaced</definition>
            <example>
                <example_x0020_sentence>The city's streets form a
grid</example_x0020_sentence>
            </example>
        </sense>
    </lexicalEntry>
    <lexicalEntry>
        <partOfSpeech>n</partOfSpeech>
        <headword>horse</headword>
        <sense/>
        <sense>
            <semantic_x0020_domain>animal</semantic_x0020_domain>
```

```
            <definition>a large solid-hoofed herbivorous ungulate mammal (Equus
caballus, family Equidae, the horse family) domesticated since prehistoric times and
used as a beast of burden, a draft animal, or for riding</definition>
            <gloss>a large solid-hoofed herbivorous ungulate mammal</gloss>
            <example>
                <example_x0020_sentence>Before the advent of mechanized vehicles, the
horse was widely used as a draft animal and riding on horseback was one of the chief
means of transportation.</example_x0020_sentence>
            </example>
        </sense>
        <form>
            <phoneticForm>/h&#596;rs/</phoneticForm>
            <pronunciation>hawrs</pronunciation>
            <multi_x0020_media>

<image>http://lux16.mpi.nl:80/mpi/lexus/resources/172.16.24.1331189759334428horse.jpg<
/image>

<sound>http://lux16.mpi.nl:80/mpi/lexus/resources/172.16.24.1331189759473836horse001.w
av</sound>
            </multi_x0020_media>
            <inflected_x0020_form>
                <grammaticalNumber>plural</grammaticalNumber>
                <orthography>horses</orthography>
            </inflected_x0020_form>
        </form>
    </lexicalEntry>
    <lexicalEntry>
        <partOfSpeech>n</partOfSpeech>
        <headword>lexicon</headword>
        <form>
            <pronunciation>lek-si-kon, -kuhn</pronunciation>
            <phoneticForm>&#712;l&#603;ks&#618;&#716;k&#594;n, -
k&#601;n</phoneticForm>
            <inflected_x0020_form>
                <grammaticalNumber>plural</grammaticalNumber>
                <orthography>lexica</orthography>
            </inflected_x0020_form>
            <multi_x0020_media>

<sound>http://lux16.mpi.nl:80/mpi/lexus/resources/172.16.24.1331189764504074lexico08.w
av</sound>
            </multi_x0020_media>
        </form>
        <sense>
            <synonym>glossary</synonym>
            <gloss>dictionary</gloss>
            <synonym>dictionary</synonym>
            <definition>the vocabulary of a person, language or branch of knowledge
</definition>
            <example>
                <example_x0020_sentence>The size of the english
lexicon</example_x0020_sentence>
            </example>
        </sense>
    </lexicalEntry>
    <lexicalEntry>
        <partOfSpeech>v</partOfSpeech>
        <headword>overrule (to)</headword>
        <form>
            <phoneticForm>&#716;o&#650;v&#601;r&#712;rul</phoneticForm>
            <pronunciation>oh-ver-rool</pronunciation>
            <multi_x0020_media>

<sound>http://lux16.mpi.nl:80/mpi/lexus/resources/172.16.24.1331189764535934overru01.w
av</sound>
            </multi_x0020_media>
            <inflected_x0020_form>
                <orthography>-ruled</orthography>
                <tense>past </tense>
            </inflected_x0020_form>
        </form>
        <sense/>
        <sense>
            <synonym>alter</synonym>
            <gloss>cancel</gloss>
```

```
            <definition>to rule against or disallow the arguments of (a
person)</definition>
            <synonym>veto</synonym>
            <example>
                <example_x0020_sentence>The senator was overruled by the committee
chairman.</example_x0020_sentence>
            </example>
        </sense>
    </lexicalEntry>
    <lexicalEntry>
        <headword>swarm</headword>
        <partOfSpeech>n</partOfSpeech>
        <sense>
            <synonym>horde</synonym>
            <definition>a body of honeybees that emigrate from a hive and fly off
together, accompanied by a queen, to start a new colony</definition>
            <gloss>large or dense group of insects</gloss>
            <semantic_x0020_domain>animals</semantic_x0020_domain>
        </sense>
        <sense/>
        <form>
            <pronunciation>swawrm</pronunciation>
            <phoneticForm>sw&#596;rm</phoneticForm>
            <multi_x0020_media>

<sound>http://lux16.mpi.nl:80/mpi/lexus/resources/172.16.24.1331189764553578swarm001.w
av</sound>

<image>http://lux16.mpi.nl:80/mpi/lexus/resources/172.16.24.1331189763755461swarm.jpg<
/image>
            </multi_x0020_media>
        </form>
    </lexicalEntry>
</lexicon>
```

10 NEDO Test Suite

NEDO has been launched as a two year project to create a common standard for Asian language resources. The project is comprised of the following four research items. (1) building a description framework of lexical entries (2) building sample lexicons (3) building an upper-layer ontology (4) evaluation of the proposed framework through an application Figure 1 illustrates the relations among these research items. The description framework of lexical entries which fits with as many Asian languages as possible, has been built in the framework of ISO-TC37/SC4 activities. LMF has been used to describe several lexical entries of several Asian languages. Through building sample lexicons, we will find problems of the existing standard, and extend it so as to fit with Asian languages.

In the following section, the Japanese lexical entry “Âô” (English “to sell”) is provided. It should be noted that the NEDO entries are totally conformant to the LMF DTD.

11 Examples of LMF conformant lexical entries from NEDO

```
<Lexicon>
<!-- Lexical Entry of 'Âô' -->
<LexicalEntry>
    <feat att="partOfSpeech" val="verb"/>
    <Lemma>
        <feat att="writtenForm" val="Âô"/>
    </Lemma>
    <SyntacticBehaviour subcategorizationFrame="regularSVO"/>
    <Sense id="Âô" synset="Âô">
        <PredicativeRepresentation predicate="SP1"
            correspondences="SVO_XY"/>
    </Sense>
</LexicalEntry>


<!-- subcategorization frame of regular SVO -->
<SubcategorizationFrame id="regularSVO">
    <SyntacticArgument id="synArgX">
        <feat att="function" val="subject"/>
        <feat att="syntacticConstituent" val="NP"/>
    </SyntacticArgument>
    <SyntacticArgument id="synArgY">
        <feat att="function" val="object"/>
        <feat att="syntacticConstituent" val="NP"/>
    </SyntacticArgument>
</SubcategorizationFrame>

<!-- correspondence between syntactic and semantic argument -->
<SynSemCorrespondence id="SVO_XY">
    <SynSemArgMap synFeature="synArgX" semFeature="X"/>
    <SynSemArgMap synFeature="synArgY" semFeature="Y"/>
</SynSemCorrespondence>

<!-- semantic argument of `sell' and `buy' -->
<SemanticPredicate id="SP1">
    <feat att="label" val="human_ACT_product"/>
    <SemanticArgument>
        <feat att="label" val="X"/>
          <feat att="semanticRole" val="agent"/>
          <feat att="restriction" val="human"/>
    </SemanticArgument>
    <SemanticArgument>
        <feat att="label" val="Y"/>
          <feat att="semanticRole" val="patient"/>
          <feat att="restriction" val="product"/>
    </SemanticArgument>
</SemanticPredicate>

<SemanticPredicate id="SP2">
    <feat att="label" val="human_ACT_symbol"/>
    <SemanticArgument>
        <feat att="label" val="X"/>
          <feat att="semanticRole" val="agent"/>
```

```
          <feat att="restriction" val="human"/>
    </SemanticArgument>
    <SemanticArgument>
        <feat att="label" val="Y"/>
          <feat att="semanticRole" val="patient"/>
          <feat att="restriction" val="symbol"/>
    </SemanticArgument>
</SemanticPredicate>

<!-- semantic class of 'Âô' -->
<SynSet id="Âô" source="traditional_american_dictionary">
    <Definition>
          <feat attr="text" val="To exchange or deliver for money or its
equivalent."/>
    </Definition>
</SynSet>

<!-- semantic class of 'Âò' -->
<SynSet id="Âò" source="traditional_american_dictionary">
    <Definition>
          <feat attr="text" val="To acquire in exchange for money or its
equivalent."/>
    </Definition>
</SynSet>
</Lexicon>
```

12 BioLexicon test suite

The BioLexicon is a large-scale resource that combines terminological data coming from the various available databases (mostly UniProt, Swiss-Prot, ChEBI, the BioThesaurus and the NCBI taxonomy) enriched with lexical information extracted from texts. Morphological, syntactic and lexical semantic properties of terms are represented for each term and term variant. It is conceived as a resource that integrates features of both terminologies and lexicons and as an extendable resource that can be incremented with entries and lexical properties for bio-terms and bio-events automatically extracted from texts. Since the aim is semantic interoperability in the biology community, the ISO Lexical Markup Framework was chosen as the reference meta-model for the structure of the BioLexicon and the ISO Data Categories as the main building blocks for the representation of the entries. Most of the Data Categories used in the BioLexicon are in fact drawn from the standardized Data Category Registry (Francopulo et al 2006).

The BioLexicon is modeled into an XML DTD according to the LMF core model plus objects taken from the three NLP extensions for the representation of morphological, syntactic and (lexical) semantics properties of terms. The set of lexical objects and relations can be seen as the skeleton of the lexical entries. The content that allows for their actual representation is formed by the set of Data Categories, i.e. features used to decorate such objects. The set of Data Categories to be used in the BioLexicon is created both by drawing them from the standard sets of the ISO Data Category Registry and by creating specific ones for the Bio domain. In doing this, we also aim at establishing a standard set of Data Categories for this domain.

The BioLexicon as described has then been implemented in a MySQL relational database, which is therefore LMF compliant. Moreover, java-based loading software procedures have been realized for the automatic population of the database based on XML exchange format.

13 Examples of LMF conformant lexical entries from the BioLexicon

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE LexicalResource SYSTEM "BioLexicon.dtd" >
<LexicalResource>
         <GlobalInformation >
                  <DC att="Authors" val="PISA-GROUP"></DC>
         </GlobalInformation>
         <Lexicon>
                  <DC att="name" val="Bio-lexicon"/>
                  <DC att="language" val="eng"/>
<!--##########Lexical entry starts: it carries PoS info and SouceId PoS is implemented
as a special subType of DC; SourceId belongs to the set of DC  -->
                  <LexicalEntry  ID="LE_interleukin-2">
                           <POSDC POSAtt="POS" POSVal="Noun"/>
                                <SOURCEDC SourceAtt="Source" SourceVal="Q4U313a"/>
<!--###The morphological component. Lemma starts: it carries an attribute for
preferred basename.
                        The object can be decorated with other info taken from the
DatCat Repository -->
                      <Lemma ID="LM_interleukin-2"
                           basename="interleukin-2" >
                           <DC att="GRAMMATICALNUMBER" val="Singular"/>
                           <DC att="GRAMMATICALGENDER" val="Feminine"/>
<!--##################RepresentationFrame carries info concerning othographical
variants -->
                           <RepresentationFrame ID="RF_interleukin-2"
writtenform="interleukin-2">
                                    <DC att="RFDC1" val="RFDCVAL1"/>
                                    <VariantDC VariantAtt="VariantType"
VariantVal="FullForm"/>
                           </RepresentationFrame>
                           <RepresentationFrame ID="RF_IL2" writtenform="IL2">
                                    <DC att="RFDC1" val="RFDCVAL1"/>
                                    <VariantDC  VariantAtt="VariantType"
VariantVal="achronym"/>
                           </RepresentationFrame>
                           <RepresentationFrame ID="RF_IL-2" writtenform="IL-2">
```

```
                                        <VariantDC  VariantAtt="VariantType"
VariantVal="orthographic"/>
                                </RepresentationFrame>
                            </Lemma>
                                <SyntacticBehaviour id="SB_interleukin-2"
subcategorizationFrames="N0">
                                </SyntacticBehaviour>
                                <Sense id="S_interleukin-2">
                                        <SenseRelation targets="S_T-Cell_growth_factor">
                                                <DC att="semanticrelation"
val="synonym"/>
                                        </SenseRelation>
                                </Sense>
                        </LexicalEntry>
                        <LexicalEntry  ID="LE_T-Cell_growth_factor">
                                <POSDC POSAtt="POS" POSVal="Noun"/>
                                <SOURCEDC SourceAtt="Source" SourceVal="Q4U313b"/>
<!--###The morphological component. Lemma starts: it carries an attribute for
preferred basename.
                                        The object can be decorated with other info
taken from the DatCat Repository -->
                                <Lemma ID="LM_T-Cell_growth_factor"
                                        basename="T-Cell_growth_factor" >
                                        <DC att="GRAMMATICALNUMBER" val="Singular"/>
<!--###################RepresentationFrame carries info concerning othographical
variants -->
                                <RepresentationFrame ID="RF_T-Cell_growth_factor"
writtenform="T-Cell_growth_factor">
                                                <DC att="RFDC1" val="RFDCVAL1"/>
                                                <VariantDC  VariantAtt="VariantType"
VariantVal="FullForm"/>
                                        </RepresentationFrame>
                                        <RepresentationFrame ID="RF_TCGF"
writtenform="TCGF">
                                                <DC att="RFDC1" val="RFDCVAL1"/>
                                                <VariantDC  VariantAtt="VariantType"
VariantVal="Achronym"/>
                                                <SCOREDC ScoreAtt="Confidence"
ScoreVal="0.8"/>
                                </RepresentationFrame>
                                </Lemma>
                                <WordForm ID="WF_T-Cell_growth_factors"
                                inflectedform="T-Cell_growth_factors" >
                                        <DC att="GRAMMATICALNUMBER" val="Plural"/>
                                        <RepresentationFrame ID="RF_T-
Cell_growth_factors" writtenform="T-Cell_growth_factors">
                                                <VariantDC  VariantAtt="VariantType"
VariantVal="InflForm"/>
                                        </RepresentationFrame>
                                        </WordForm>
                                <SyntacticBehaviour id="SB_T-Cell_growth_factor"
subcategorizationFrames="N0">
                                </SyntacticBehaviour>
                                <Sense id="S_T-Cell_growth_factor">
                                </Sense>
                        </LexicalEntry>
                <SubcategorizationFrame id="N0">
                </SubcategorizationFrame>
                </Lexicon>
</LexicalResource>
```

Appendix A - The LMF DTD

```
<?xml version='1.0' encoding="UTF-8"?>
            <!-- DTD for LMFNLP packages-->
            <!--#################### Core package-->
<!ELEMENT LexicalResource (feat*, GlobalInformation, Lexicon+, SenseAxis*,
TransferAxis*, ExampleAxis*)>
<!ATTLIST LexicalResource
    dtdVersion CDATA     #FIXED "14">
<!ELEMENT GlobalInformation (feat*)>
<!ELEMENT Lexicon (feat*, LexicalEntry+,  SubcategorizationFrame*,
SubcategorizationFrameSet*, SemanticPredicate*, Synset*,
                                  SynSemCorrespondence*, ParadigmPattern*,
TransformClass*, MWEPattern*, ConstraintSet*)>
<!ELEMENT LexicalEntry (feat*, Lemma, WordForm*, StemOrRoot*, DerivedForm*,
ReferredRoot*, ListOfComponents?, Sense*, SyntacticBehaviour*)>
<!ATTLIST LexicalEntry
    id       ID #IMPLIED
    paradigmPattern IDREFS #IMPLIED
    mwePattern  IDREF #IMPLIED>
<!ELEMENT Sense (feat*, PredicativeRepresentation*, SenseExample*, SemanticDefinition*,
SenseRelation*,
                                MonolingualExternalRef*)>
<!ATTLIST Sense
    id       ID #IMPLIED
    inherit  IDREFS #IMPLIED
 synset IDREF #IMPLIED>
            <!--#################### Package for Morphology -->
<!ELEMENT Lemma (feat*, FormRepresentation*)>
<!ELEMENT WordForm (feat*, FormRepresentation*)>
<!ELEMENT StemOrRoot (feat*, FormRepresentation*, MorphologicalFeatures*)>
<!ELEMENT FormRepresentation (feat*)>
<!ELEMENT DerivedForm (feat*, FormRepresentation*)>
<!ATTLIST DerivedForm
    targets IDREFS #IMPLIED>
<!ELEMENT ReferredRoot (feat*, FormRepresentation*)>
<!ATTLIST ReferredRoot
    targets IDREFS #IMPLIED>
<!ELEMENT ListOfComponents (feat*, Component+)>
<!ELEMENT Component (feat*)>
<!ATTLIST Component
    entry IDREF #REQUIRED>
<!ELEMENT MorphologicalFeatures (feat*)>
            <!--#################### Package for Syntax -->
<!ELEMENT SyntacticBehaviour (feat*)>
<!ATTLIST SyntacticBehaviour
    id                                          ID #IMPLIED
    senses                                      IDREFS #IMPLIED
    subcategorizationFrames  IDREFS #IMPLIED
    subcategorizationFrameSets        IDREFS #IMPLIED>
<!ELEMENT SubcategorizationFrame (feat*, LexemeProperty?, SyntacticArgument*)>
<!ATTLIST SubcategorizationFrame
    id                ID #IMPLIED
    inherit          IDREFS #IMPLIED>
<!ELEMENT LexemeProperty (feat*)>
<!ELEMENT SyntacticArgument (feat*)>
<!ATTLIST SyntacticArgument
    id                ID #IMPLIED
    target           IDREF #IMPLIED>
<!ELEMENT SubcategorizationFrameSet (feat*, SynArgMap*)>
<!ATTLIST SubcategorizationFrameSet
    id                                          ID #IMPLIED
    subcategorizationFrames  IDREFS #IMPLIED
    inherit                                     IDREFS #IMPLIED>
<!ELEMENT SynArgMap (feat*)>
<!ATTLIST SynArgMap
    arg1  IDREF #REQUIRED
    arg2  IDREF #REQUIRED>
            <!--#################### Package for Semantics -->
<!ELEMENT PredicativeRepresentation (feat*)>
<!ATTLIST PredicativeRepresentation
   predicate             IDREF #REQUIRED
   correspondences IDREFS #REQUIRED>
```

```
<!ELEMENT SemanticPredicate (feat*, SemanticDefinition*, SemanticArgument*,
PredicateRelation*)>
<!ATTLIST SemanticPredicate
    id               ID #REQUIRED>
<!ELEMENT SemanticArgument (feat*)>
<!ATTLIST SemanticArgument
    id               ID #IMPLIED>
<!ELEMENT SynSemCorrespondence (feat*,SynSemArgMap*)>
<!ATTLIST SynSemCorrespondence
    id              ID #REQUIRED>
<!ELEMENT SynSemArgMap (feat*)>
<!ATTLIST SynSemArgMap
   synFeature        CDATA #REQUIRED
   semFeature       CDATA #REQUIRED>
<!ELEMENT PredicateRelation (feat*)>
<!ATTLIST PredicateRelation
    targets          IDREFS #IMPLIED>
<!ELEMENT SenseExample (feat*)>
<!ATTLIST SenseExample
    id               ID #IMPLIED>
<!ELEMENT SemanticDefinition (feat*, Statement*)>
<!ELEMENT Statement (feat*)>
<!ELEMENT Synset (feat*, SemanticDefinition*, SynsetRelation*,
MonolingualExternalRef*)>
<!ATTLIST Synset
    id               ID #IMPLIED>
<!ELEMENT SynsetRelation (feat*)>
<!ATTLIST SynsetRelation
    targets          IDREFS #IMPLIED>
<!ELEMENT MonolingualExternalRef (feat*)>
<!ELEMENT SenseRelation (feat*)>
<!ATTLIST SenseRelation
    targets  IDREFS #REQUIRED>
            <!--#################### Package for Multilingual notations -->
<!ELEMENT SenseAxis (feat*, SenseAxisRelation*, InterlingualExternalRef*)>
<!ATTLIST SenseAxis
    id             ID #IMPLIED
    senses         IDREFS #IMPLIED
    synsets        IDREFS #IMPLIED>
<!ELEMENT InterlingualExternalRef (feat*)>
<!ELEMENT SenseAxisRelation (feat*)>
<!ATTLIST SenseAxisRelation
    targets          IDREFS #REQUIRED>
<!ELEMENT TransferAxis (feat*, TransferAxisRelation*, SourceTest*, TargetTest*)>
<!ATTLIST TransferAxis
    id               ID #IMPLIED
    syntacticBehaviours IDREFS #IMPLIED>
<!ELEMENT TransferAxisRelation (feat*)>
<!ATTLIST TransferAxisRelation
    targets          IDREFS #REQUIRED>
<!ELEMENT SourceTest (feat*)>
<!ATTLIST SourceTest
    syntacticBehaviours IDREFS #REQUIRED>
<!ELEMENT TargetTest (feat*)>
<!ATTLIST TargetTest
    syntacticBehaviours IDREFS #REQUIRED>
<!ELEMENT ExampleAxis (feat*, ExampleAxisRelation*)>
<!ATTLIST ExampleAxis
    id               ID #IMPLIED
    examples IDREFS #IMPLIED>
<!ELEMENT ExampleAxisRelation (feat*)>
<!ATTLIST ExampleAxisRelation
    targets          IDREFS #REQUIRED>
            <!--#################### Package for paradigm patterns -->
<!ELEMENT ParadigmPattern (feat*, TransformSet*, Affix*, PrefixSlot*, InfixSlot*,
SuffixSlot*)>
<!ATTLIST ParadigmPattern
    id       ID #REQUIRED>
<!ELEMENT TransformSet (feat*, Process*, MorphologicalFeatures*)>
<!ELEMENT Process (feat*, Condition*, MorphologicalFeatures*)>
<!ELEMENT Condition (feat*)>
<!ATTLIST Condition
    id      ID #IMPLIED>
<!ELEMENT Affix (feat*, FormRepresentation*, Condition*, AffixAllomorph*,
MorphologicalFeatures*)>
<!ELEMENT AffixAllomorph (feat*, FormRepresentation*)>
```

```
<!ATTLIST AffixAllomorph
    conditions IDREFS #IMPLIED>
<!ELEMENT PrefixSlot (feat*, Affix*)>
<!ELEMENT InfixSlot (feat*, Affix*)>
<!ELEMENT SuffixSlot (feat*, Affix*)>
<!ELEMENT TransformClass (feat*)>
<!ATTLIST TransformClass
    id       ID #REQUIRED>
            <!--#################### Package for MWE patterns -->
<!ELEMENT MWEPattern (feat*, MWENode*)>
<!ELEMENT MWENode (feat*, MWEEdge*, MWELex)>
<!ELEMENT MWEEdge (feat*, MWENode*)>
<!ELEMENT MWELex (feat*)>
            <!--#################### Package for Constraint expression -->
<!ELEMENT ConstraintSet (feat*, Constraint*)>
<!ELEMENT Constraint (feat*, LogicalOperation*)>
<!ATTLIST Constraint
    id       ID #IMPLIED>
<!ELEMENT LogicalOperation (feat*, AttributeValuation*)>
<!ATTLIST LogicalOperation
   constraints  IDREFS #IMPLIED>
<!ELEMENT AttributeValuation (feat*)>
            <!--#################### for datcat adornment: feat stands for feature-->
<!ELEMENT feat EMPTY>
            <!-- att=constant to be taken from the DCR -->
            <!-- val=free string or constant to be taken from the DCR-->
<!ATTLIST feat
    att      CDATA #REQUIRED
    val      CDATA #REQUIRED>
```

Appendix B – The BioLexicon DTD

```
 <?xml version='1.0' encoding="UTF-8"?>
            <!-- DTD for LMFNLP packages-->
            <!--#################### Core package-->
            <!--DTD only for Morphologic entities -->
            <!--Some other entities have to be included-->
<!ELEMENT LexicalResource (DC*, GlobalInformation, Lexicon+)>
<!ATTLIST LexicalResource
    dtdVersion CDATA     #FIXED "1.1">
<!ELEMENT GlobalInformation (DC*)>
<!ELEMENT Lexicon (DC*, LexicalEntry+,SubcategorizationFrame*)>
<!ELEMENT LexicalEntry ( POSDC+,DC*,Lemma,SyntacticBehaviour*,Sense*)>
<!ATTLIST LexicalEntry
    ID       ID      #REQUIRED  >
<!ELEMENT Lemma (DC* ,RepresentationFrame*)>
<!ATTLIST Lemma
       ID                          ID #REQUIRED
       BASENAME CDATA #REQUIRED >
 <!ELEMENT RepresentationFrame (DC*,VariantDC*)>
<!ATTLIST RepresentationFrame
     ID     ID #IMPLIED>
 <!ELEMENT SyntacticBehaviour (DC*)>
<!ATTLIST SyntacticBehaviour
    id                                          ID #IMPLIED
    senses                                      IDREFS #IMPLIED
    subcategorizationFrames  IDREFS #IMPLIED
    subcategorizationFrameSets        IDREFS #IMPLIED>
   <!ELEMENT Sense (DC*, SenseRelation*)>
<!ATTLIST Sense
    id       ID #IMPLIED>
<!ELEMENT SenseRelation (DC*)>
<!ATTLIST SenseRelation
    targets  IDREFS #REQUIRED>

 <!ELEMENT SubcategorizationFrame (DC*)>
<!ATTLIST SubcategorizationFrame
    id                ID #IMPLIED >
<!ELEMENT DC EMPTY>
            <!-- att=constant to be taken from the DCR -->
            <!-- val=free string or constant to be taken from the DCR-->


<!ATTLIST DC
    att      CDATA #REQUIRED
    val      CDATA #REQUIRED>

   <!ELEMENT VariantDC EMPTY>
            <!-- att=constant to be taken from the DCR -->
            <!-- val=free string or constant to be taken from the DCR-->
<!ATTLIST VariantDC
    writtenForm      CDATA #REQUIRED
    VariantType CDATA #REQUIRED>

    <!ELEMENT POSDC EMPTY>
            <!-- att=constant to be taken from the DCR -->
            <!-- val=free string or constant to be taken from the DCR-->
<!ATTLIST POSDC
    POS_att CDATA #REQUIRED
    POS_val CDATA #REQUIRED>
```

References

Bertagna F., Lenci A., Monachini M., Calzolari N. 2004- Content Interoperability of Lexical Resources: Open Issues and "MILE" Perspectives - LREC 2004: Fourth International Conference on Language Resources and Evaluation, held in Memory of Antonio Zampolli. Lisbon, Portugal, 26th, 27th & 28 May 2004. Proceedings, Volume I, Paris, The European Language Resources Association (ELRA). 131-134.

Francopoulo G., Declerck T., Monachini M., Romary L. 2006 - The relevance of standards for research infrastructures - LREC 2006: 5th International Conference on Language Resources and Evaluation. Genoa, Italy, 24-25-26 May 2006. Proceedings, Paris, The European Language Resources Association (ELRA). CD-ROM, 19-22.

Monachini M., Quochi V., Ruimy N., Calzolari N. 2007 - Lexical Relations and Domain Knowledge: The BioLexicon Meets the Qualia Structure - In Proceeding of the 4th International Conference on Generative Approaches to the Lexicon, 10-11 May 2007, Paris.

Quochi V., Del Gratta R., Sassolini E., Monachini M, Calzolari N. 2007 – Toward a Standard Lexical Resource in the Bio Domain. In Proceedings of the 3rd Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics, October 5-7, 2007, Poznań, Poland.

Ruimy N., Monachini M., Gola E., Calzolari N., Del Fiorentino M.C., Ulivieri M., Rossi S. 2003 - A computational semantic lexicon of Italian: SIMPLE - In A. Zampolli, N. Calzolari, L. Cignoni, (eds.), Computational Linguistics in Pisa - Linguistica Computazionale a Pisa. Linguistica Computazionale, Special Issue, XVIII-XIX, (2003). Pisa-Roma, IEPI. Tomo II, 821-864.

Tokunaga T., Sornlertlamvanich V., Charoenporn T., Calzolari N., Monachini M., Soria C., Huang C., Prevot L., Xia Y., Yu H., Kiyoaki S. 2007 - Infrastructure for standardization of Asian language resources - In Proceedings of COLING/ACL 2006 Main Conference Poster Sessions, Sydney, Australia. 827-834.

http://xmlstar.sourceforge.net/