Monnet Project

lemon: Linked Data, Lexicons and Data Category Registries

John McCrae[1] and Guadalupe Aguado-de-Cea[2]

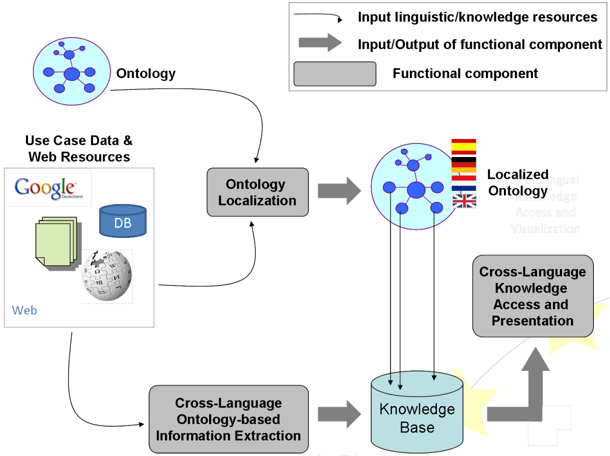
[1]Cognitive Interaction Technology Excellence Center, Universität Bielefeld

[2]Ontology Engineering Group, Universidad Politécnica de Madrid

Monnet: Main Objectives

- Multilingual Ontologies for Networked Knowledge
  - Linguistically enriched knowledge representation
  - Multilingual access to structured/networked knowledge: ontologies, knowledge bases, linked data
- Handling Information at the Semantic Level
  - Abstracting from language and form
  - Cross-lingual information
    1. Integration
    2. Aggregation
    3. Querying
    4. Presentation

Monnet



Input linguistic/knowledge resources
Input/Output of functional component
Functional component
Ontology
Use Case Data & Web Resources
Google
DB
Web
Ontology Localization
Localized Ontology
Cross-Language Knowledge Access and Presentation
Cross-Language Ontology-based Information Extraction
Knowledge Base

J. McCrae · G Aguado-de-Cea

WP2 Objectives

- Objectives
  - Define models, methods and tools for the localization of ontologies, by means of an **Le**xicon **M**odel for **On**tologies
  - We define **lemon** as a model for this
- Lemon as a lexicon
  - Lexicons provide linguistic data for NLP applications
  - Linked data is a way of sharing information on the (semantic) web
  - There are many categories of linguistic data and disagreement about the values, semantics and restrictions
  - Different granularity of linguistic information for different applications

lemon's origins

- Lexical Markup Framework (ISO 24613)
  - Standard for representing lexicons
  - XML
- LexInfo, LIR
  - Represent lexical information relative to an ontology
  - OWL
- SKOS (W3C Standard)
  - Designed for Taxonomy/Vocabulary representation
  - RDF

Design goals

- RDF(S)
- Minimalist
- Not prescriptive (i.e., uses data categories)
- Relative semantics (i.e., uses ontologies)
- Modular and extensible

Why lemon: RDF(S)

- RDF models are labelled directed graphs
  - Allows for smarter representation
- Each entry has a URI
  - Queriable on the web using standards
  - Clear ownership of data categories
- Linking possible between different lexica
  - Reuse of lexicon data
- Some induction possible (subproperties, classes etc.)

Why lemon: Minimalism

- Small models (i.e., fewer links, fewer kB)
- Easier to understand
- "Open-world": Not necessary to state all facts
    - Multiple points of view

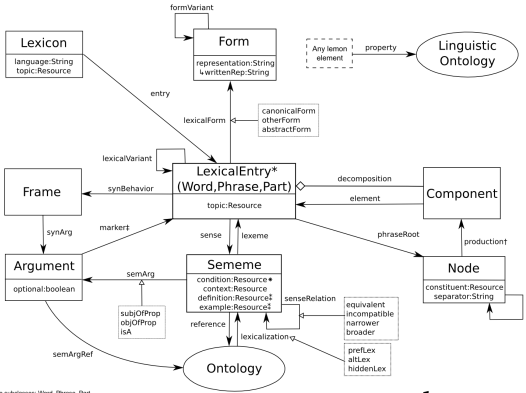
Why lemon: Relative semantics

- Meaning of a word given by reference
- Reference (generally an ontology) capable of representing more complex semantic information
- Disambiguation is performed relative to the ontology
- No (traditional) word senses
  - No clashing of word senses in cross-lingual mappings

Why lemon: Modular and extensible

- RDF(S) extensibility allows representation of
  - Subtle differences
  - Unexpected data categories
- Modularity
  - Different modules for different user requirements
  - New modules can be added later without affecting core

The model



* LexicalEntry has three subclasses: Word, Phrase, Part
‡ definition and example are stated as a nodes with a value
⁎ condition has subproperties propertyDomain and propertyRange
† production can also refer to arguments
‡ marker can also refer to linguistic ontology

lemon

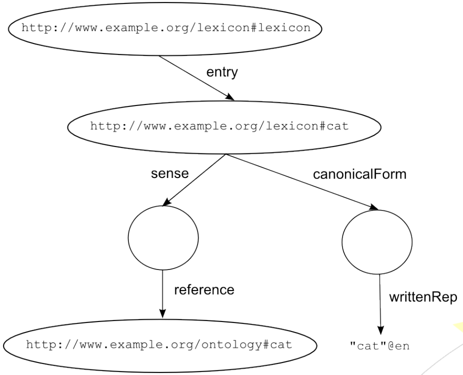
A simple example

```
@base <http://www.example.org/lexicon#>
@prefix ontology: <http://www.example.org/ontology#>
@prefix lemon: <http://www.monnet-project.eu/lemon#>

:lexicon lemon:language "en" ;
  lemon:entry :cat .

:cat a lemon:Word ;
  lemon:canonicalForm [ lemon:writtenRep "cat"@en ] ;
  lemon:sense [ lemon:reference ontology:cat ] .
```

A simple example



http://www.example.org/lexicon#lexicon
entry
http://www.example.org/lexicon#cat
sense
canonicalForm
reference
writtenRep
http://www.example.org/ontology#cat
"cat"@en

Adding a plural form: the ugly

```
:cat a lemon:Word ;
  lemon:canonicalForm [ lemon:writtenRep "cat"@en ] ;
  lemon:otherForm [ lemon:writtenRep "cats"@en ;
     lemon:property [ lemon:value "plural" ] ] .
```

- Does not indicate type of data category
- Different entity for each annotation
- Value could be misspelt or ambiguous

Adding a plural form: the bad

```
:cat a lemon:Word ;
  lemon:canonicalForm [ lemon:writtenRep "cat"@en ] ;
  lemon:otherForm [ lemon:writtenRep "cats"@en ;
                    :number :plural ] .

:number rdfs:subPropertyOf lemon:property .
```

- Property and value have unique name
- Must define properties for each lexical resource

Adding a plural form: the good

```
:cat a lemon:Word ;
  lemon:canonicalForm [ lemon:writtenRep "cat"@en ] ;
  lemon:otherForm [ lemon:writtenRep "cats"@en ;
                    dcr:number dcr:plural ] .
```

- Property and values standardized by DCR
- All lexicons refer to the property the same way

Representing variation

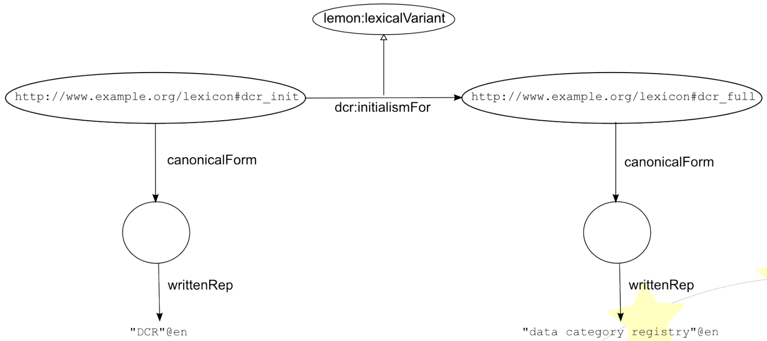
```
:dcr_init a lemon:Word ;
  lemon:canonicalForm [ lemon:writtenRep "DCR"@en ] .

:dcr_full a lemon:Phrase ;
  lemon:canonicalForm
    [ lemon:writtenRep "data category registry"@en ] .

:dcr_init dcr:initialismFor :dcr_full .

dcr:initialismFor
  rdfs:subPropertyOf lemon:lexicalVariant .
```

Representing variation


lemon:lexicalVariant
http://www.example.org/lexicon#dcr_init
dcr:initialismFor
http://www.example.org/lexicon#dcr_full
canonicalForm
canonicalForm
writtenRep
writtenRep
"DCR"@en
"data category registry"@en

ISOcat as DCR

- ISOcat is large
- Each entity has a unique identifier
- Distinguishes between properties (open) and values (closed)
- States ranges and dependencies
- Dereferenceable as RDF

Issues with ISOcat

- DCs are not clear to humans
  - `dcr:noun => isocat:1333 => http://www.isocat.org/rest/dc/1333`
  - `isocat6:noun => http://www.isocat.org/rest/profile/6#noun`
- RDF representation does not convert DCIF information
  - Open/Closed => Property/Resource
  - Domain values => Ranges
- Representation not aligned to lemon
  - Description
  - Representation
  - Relation

DCRs for Lemon

- Base DCR on ISOcat
- Publish only in RDF(S)
- Include references to lemon
- Add OWL constraints (where applicable)
- Reference DCR by use of `dcr:datcat` annotation

Conclusion

- lemon is an extensible model for linked data lexica
- Interacts with existing technologies
  - LMF conversion at `http://www.lexinfo.net/lemon2lmf`
- Data categories allow for representation of arbitrary linguistic information
- Importing from ISOcat is very useful for creating lemon lexica