Multilingual Linked Open Data Patterns

Jose Emilio Labra Gayo
University of Oviedo, Spain

Joint work with:

Dimitris Kontokostas       Sören Auer
Universität Leipzig, Germany

More info: http://www.weso.es/MLODPatterns
MLOD Patterns

From best practices (Dublin) to patterns (Rome)

We propose a catalog of 20 MLOD patterns

Pattern = generic solution to a problem in a context

Common vocabulary
Patterns can be related to each other
Some patterns can contradict other patterns

There are already Linked data patterns
We focus on multilingual linked data patterns

Each pattern contains
Name
Description
Context
Example
Discussion
See also

Based on DBPedia I18n experience

More info: http://www.weso.es/MLODPatterns
MLOD Patterns

Patterns are classified by activity:

Naming:
- URI design, URIs, IRIs, etc.

Dereference:
- How is the content that we return affected by multilingualism

Labeling:
- Handling multilingual labels

Longer descriptions:
- Longer textual descriptions

Linking:
- Links between concepts in different languages?

Reuse:
- Vocabularies and multilingualism

More info: http://www.weso.es/MLODPatterns
## General overview

<table>
<thead>
<tr>
<th>Goal</th>
<th>Pattern</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Naming</td>
<td>Descriptive URIs</td>
<td>Use descriptive URIs with ASCII characters, % encoding extended characters</td>
</tr>
<tr>
<td></td>
<td>Opaque URIs</td>
<td>Use non human-readable URIs</td>
</tr>
<tr>
<td></td>
<td>Full IRIs</td>
<td>Use IRIs with unicode characters</td>
</tr>
<tr>
<td></td>
<td>Internationalized local names</td>
<td>Use Unicode characters only for local names</td>
</tr>
<tr>
<td></td>
<td>Language in URIs</td>
<td>Include language information in the URI</td>
</tr>
<tr>
<td>Dereference</td>
<td>Return Language independent data</td>
<td>Return the same triples independently of the language</td>
</tr>
<tr>
<td></td>
<td>Language content negotiation</td>
<td>Return different triples depending on user agent preferences</td>
</tr>
<tr>
<td>Labeling</td>
<td>Label everything</td>
<td>Define labels for all the resources</td>
</tr>
<tr>
<td></td>
<td>Multilingual labels</td>
<td>Add language tags to labels</td>
</tr>
<tr>
<td></td>
<td>Labels without language tag</td>
<td>Add labels without language tags in a default language</td>
</tr>
<tr>
<td>Longer descriptions</td>
<td>Divide longer descriptions</td>
<td>Replace long descriptions by more resources with labels</td>
</tr>
<tr>
<td></td>
<td>Add lexical information</td>
<td>Add lexical information to long descriptions</td>
</tr>
<tr>
<td></td>
<td>Structured literals</td>
<td>Use HTML/XML literals for longer descriptions</td>
</tr>
<tr>
<td>Linking</td>
<td>Identity links</td>
<td>Use owl:sameAs and similar predicates</td>
</tr>
<tr>
<td></td>
<td>Soft links</td>
<td>Use predicates with soft semantics</td>
</tr>
<tr>
<td></td>
<td>Linguistic metadata</td>
<td>Add linguistic metadata about the dataset terms</td>
</tr>
<tr>
<td>Reuse</td>
<td>Monolingual vocabularies</td>
<td>Attach labels to vocabularies in a single language</td>
</tr>
<tr>
<td></td>
<td>Multilingual vocabularies</td>
<td>Prefer multilingual vocabularies</td>
</tr>
<tr>
<td></td>
<td>Localize existing vocabularies</td>
<td>Translate labels of existing vocabularies</td>
</tr>
<tr>
<td></td>
<td>Create new localized vocabularies</td>
<td>Create custom vocabularies and link to existing ones</td>
</tr>
</tbody>
</table>
Motivating example

Juan is an armenian professor at the University of León.

More info: http://www.weso.es/MLODPatterns
Naming

Selecting a URI scheme for Armenia

Descriptive URIs

- Human-readable
- Good tool support

Opaque URIs

- Only ASCII, %-encode non-ASCII characters

More info: http://www.weso.es/MLODPatterns
Naming

Selecting a URI scheme for Armenia

Descriptive URIs
http://example.org/Armenia

Opaque URIs
http://example.org/I23AX45

Independence between concept and natural language representation

Non Human-readable
Difficult to handle by developers

More info: http://www.weso.es/MLODPatterns
Naming

Selecting a URI scheme for Armenia

Descriptive URIs
http://example.org/Armenia

Opaque URIs
http://example.org/I23AX45

Full IRIs
http://օրինակ.օրգ#Հայաստան

More natural for non-Latin based languages

Subject to visual spoofing attacks
Not so good tool support

More info: http://www.weso.es/MLODPatterns
Naming

Selecting a URI scheme for Armenia

Descriptive URIs
http://example.org/Armenia

Opaque URIs
http://example.org/I23AX45

Full IRIs
http://օրինակ.օրգ#Հայաստան

Internationalized local names
http://example.org#Հայաստան

Avoids domain name spoofing
More human-friendly identifiers

Visual spoofing attacks can still possible

More info: http://www.weso.es/MLODPatterns
Naming

Selecting a URI scheme for Armenia

Descriptive URIs

http://example.org/Armenia

Opaque URIs

http://example.org/I23AX45

Full IRIs

http://օրինակ.օրգ#Հայաստան

Internationalized local names

http://example.org#Հայաստան

Include language in URIs

http://hy.example.org#Հայաստան
http://en.example.org#Armenia

Independent development of datasets by language

Adding language info to the URI can become unwieldy
Example: languages & sublanguages

hy-Latin-IT-arevela

Where should we put the language tag?

More info: http://www.weso.es/MLODPatterns
Dereference

Language based content negotiation?

No language content negotiation

http://example.org/Armenia

Always returns the same data

Easy to develop
Consistency of data

Clients have to filter triples in other languages
Computation & network overhead

More info: http://www.weso.es/MLODPatterns
Dereference

Language based content negotiation?

No language content negotiation

http://example.org/Armenia

Always returns the same data

Language content negotiation

http://example.org/Armenia

Returns different data depending on Accept-language

Accept-language:en


Accept-language:hy

:Armenia rdfs:label "Հայաստան"@hy .

More info: http://www.weso.es/MLODPatterns

- Improves clients performance
- Less network overhead

- Difficult to implement
- Semantic equivalence between data
Labeling

Label everything

```
:Armenia rdfs:label "Armenia" .
:juan rdfs:label "Juan López" .
:position rdfs:label "Job position" .
```

User agents can show labels instead or URIs
Labels can be used for searching

Not always feasible, which labels?
Labels are for humans
Avoid machine-oriented notations

More info: http://www.weso.es/MLODPatterns
Labeling

Label everything

:Armenia rdfs:label "Armenia" .
:juan rdfs:label "Juan López" .
:position rdfs:label "Job position" .

Multilingual Labels

:juan :position "Professor"@en .
:juan :position "Catedrático"@es .

SPARQL can be more difficult

```
SELECT * WHERE {
  ?x :position "Professor" .
}
```

Multilingual labels are part of RDF Model

More info: http://www.weso.es/MLODPatterns
Labeling

Label everything

```turtle
:Armenia rdfs:label "Armenia" .
:juan rdfs:label "Juan López" .
:position rdfs:label "Job position" .
```

Multilingual Labels

```turtle
:juan :position "Professor"@en .
:juan :position "Catedrático"@es .
```

Labels without language tag

```turtle
:juan :position "Professor"@en .
:juan :position "Catedrático"@es .
:juan :position "Professor" .
```
Longer descriptions

Divide long descriptions

:juan :jobtitle "Professor at the University of León"@en .

More complexity of the model
Not always possible

Fine-grained data is more amenable to semantic web apps
Apps can generate more readable information

More info: http://www.weso.es/MLODPatterns
Longer descriptions

Provide lexical information

```
:unileón a lemon:LexicalEntry ;
  lemon:decomposition ( 
    [ lemon:element :University ]
    [ lemon:element :Of ]
    [ lemon:element :León ]
  );
  rdfs:label "University of León"@en .

:University a lemon:LexicalEntry ;
  lexinfo:partOfSpeech lexinfo:commonNoun ;
  rdfs:label "University"@en ;
  rdfs:label "Universidad"@es .

:Of a lemon:LexicalEntry ;
  lexinfo:partOfSpeech lexinfo:preposition ;
  rdfs:label "of"@en ;
  rdfs:label "de"@es .

:León a lemon:LexicalEntry ;
  lexinfo:partOfSpeech lexinfo:properNoun ;
  rdfs:label "León"
```

More info: http://www.weso.es/MLODPatterns
Structured literals

Leverage existing I18N techniques
   Bidi, Ruby, Localization notes, ...

Interaction between 2 abstracion models
   RDF vs XML/HTML
   Large portions of structured literals can hinder LD

More info: http://www.weso.es/MLODPatterns
Linking

Inter-language identity links

<http://hy.example.org#Հայաստան> 
owl:sameAs 
<http://en.example.org#Armenia> .

owl:sameAs is a well known property
Already supported by linked data applications

Too strong semantics of owl:sameAs
Concepts may not be the same
Contradictions

More info: http://www.weso.es/MLODPatterns
Linking

Inter-language identity links

<http://hy.example.org#Հայաստան> 
owl:sameAs 
<http://en.example.org#Armenia> .

Soft Inter-language links

<http://hy.example.org#Հայաստան> 
rdfs:seeAlso 
<http://en.example.org#Armenia> .

Several predicates
- rdfs:seeAlso
- skos:related
- dbo:wikiPageLanguageLink

No standard property
No support for inference

More info: http://www.weso.es/MLODPatterns
Linking

Inter-language identity links

<http://hy.example.org#Հայաստան> owl:sameAs <http://en.example.org#Armenia> .

Soft Inter-language links

<http://hy.example.org#Հայաստան> rdfs:seeAlso <http://en.example.org#Armenia> .

Link linguistic meta-data

:Catedrático
  lexvo:means wordnet:Professor ;

Links between multilingual labels
Can declare language of a dataset

No standard practice
Semantic equivalence between concepts

More info: http://www.weso.es/MLODPatterns
Reuse

Monolingual vocabularies

FOAF, Dublin Core, OWL, RDF Schema, ... are only in English

- Easier to control vocabulary evolution
- Avoid bad translations, ambiguities

Monolingual vocabularies in multilingual applications require a translation layer

More info: http://www.weso.es/MLODPatterns
Reuse

Monolingual vocabularies

FOAF, Dublin Core, OWL, RDF Schema, ... are only in English

Multilingual vocabularies

```prolog
:position a owl:DatatypeProperty ;
  rdfs:domain :UniversityStaff ;
  rdfs:label "Position"@en ;
  rdfs:label "Puesto"@es .

:UniversityStaff a owl:Class ;
  rdfs:label "University staff"@en ;
  rdfs:label "Trabajador universitario"@es .
```

Elegant solution in multilingual contexts
More control over translations

Some common vocabularies are monolingual
Maintenance is more difficult

More info: http://www.weso.es/MLODPatterns
Reuse

Localize existing vocabularies

```
dc:contributor rdfs:label "Colaborador"@es .
```

- Transparently select label in preferred language
- Polluting well known vocabularies = controversial

Principle AAA
Anyone can say Anything about Any topic

More info: http://www.weso.es/MLODPatterns
Reuse

Localize existing vocabularies

```
dc:contributor rdfs:label "Colaborador"@es .
```

Create new localized vocabularies

```
dc:contributor
  owl:equivalentProperty :colaborador .
:colaborador
  rdfs:label "Colaborador"@es .
```

Freedom to taylor vocabulary to specific needs

More difficult to humans/agents to recognize new properties/classes

More info: http://www.weso.es/MLODPatterns
Future work

The catalog is not closed

Other issues & patterns
  Microdata & RDFa
  Other l18n topics
  Handle big datasets
  Localization workflows

Feedback from the community
  Best practices, Patterns, Anti-patterns?

More info: http://www.weso.es/MLODPatterns