SKOS

Simple Knowledge Organization System

CS-E4410 Semantic Web, 30.01.2019

Petri Leskinen and Esko Ikkala

Aalto University, Semantic Computing Research Group (SeCo)
http://seco.cs.aalto.fi
firstname.lastname@aalto.fi

Overview

- RDF-based standard vocabulary for representing *simple* ontologies
- Provides primitives for representing commonly used structures in *legacy* vocabularies, classifications, and thesauri
  - *Can be extended using RDF(S) and OWL*
- Focus on representing vocabulary structures rather than on real world knowledge
  - *In SKOS: entries, e.g., "table" are instances of skos:Concept*
  - *In RDF(S): entries, e.g., "table" are classes of from which actual instances (of tables) are created*
  - *However, also SKOS structures are used for reasoning w.r.t. real world data*

SKOS specification

W3C®

SKOS Simple Knowledge Organization System Reference

W3C Recommendation 18 August 2009

This version:
http://www.w3.org/TR/2009/REC-skos-reference-20090818/
Latest version:
http://www.w3.org/TR/skos-reference
Previous versions:
http://www.w3.org/TR/2009/PR-skos-reference-20090615/
Editors:
Alistair Miles, STFC Rutherford Appleton Laboratory / University of Oxford
Sean Bechhofer, University of Manchester

Please refer to the **errata** for this document, which may include some normative corrections.

See also **translations**.

Concept schemes

Ontologies are instances of concept schemes

- Top concepts are pointed out

- ```
  ex:animalThesaurus rdf:type skos:ConceptScheme ;
      dct:title "Simple animal thesaurus" ;
      dct:creator ex:antoineIsaac ;
      skos:hasTopConcept ex:mammals ;
      skos:hasTopConcept ex:fish .
  ```

dct: <http://purl.org/dc/terms/> (Dublin Core terms)

Concepts and labels

Concepts (for machines)

- Instances of skos:Concept
- **`ex:animals rdf:type skos:Concept .`**

Labels (human-readable names of concepts)

- skos:prefLabel, skos:altLabel, skos:hiddenLabel
- For example:

  - *`ex:animals rdf:type skos:Concept ;`*
    `skos:prefLabel "animals"@en ;`
    `skos:altLabel "creatures"@en ;`
    `skos:prefLabel "animaux"@fr ;`
    `skos:altLabel "créatures"@fr .`

Documentary notes

- skos:note
  - *skos:scopeNote*
  - *skos:editorialNote*
  - *skos:changeNote*
  - *skos:historyNote*
- skos:definition
- skos:example

Semantic relationships

Broader/Narrower relationships

- Like in thesauri
- `ex:animals rdf:type skos:Concept ;`
  `skos:prefLabel "animals"@en ;`
  `skos:narrower ex:mammals .`
- skos:narrower / skos:broader are ***not*** transitive!
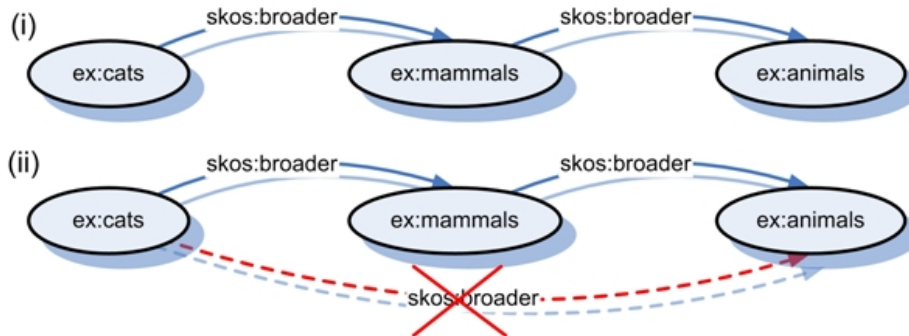
Modeling transitivity

skos:broaderTransitive and skos:narrowerTransitive

- skos:broader / skos:narrower are their subproperties
- Transitive versions are inferred, not asserted
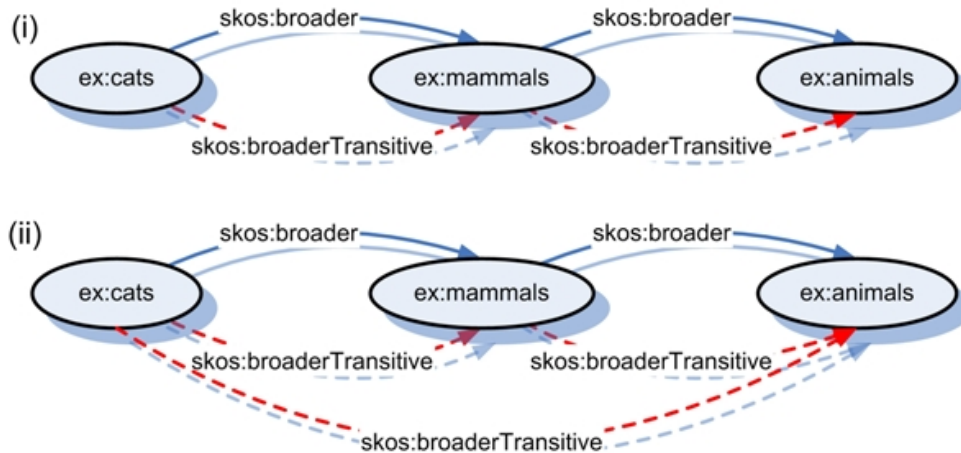
Other semantic relationships

Associative Relationships

- `ex:birds rdf:type skos:Concept ;`
    `skos:prefLabel "birds"@en ;`
    `skos:related ex:ornithology .`

skos:broader/narrower/related properties can be refined by introducing more specific properties using rdfs:subPropertyOf

- `ex:partOf rdfs:subPropertyOf skos:broader .`

SKOS ontology example
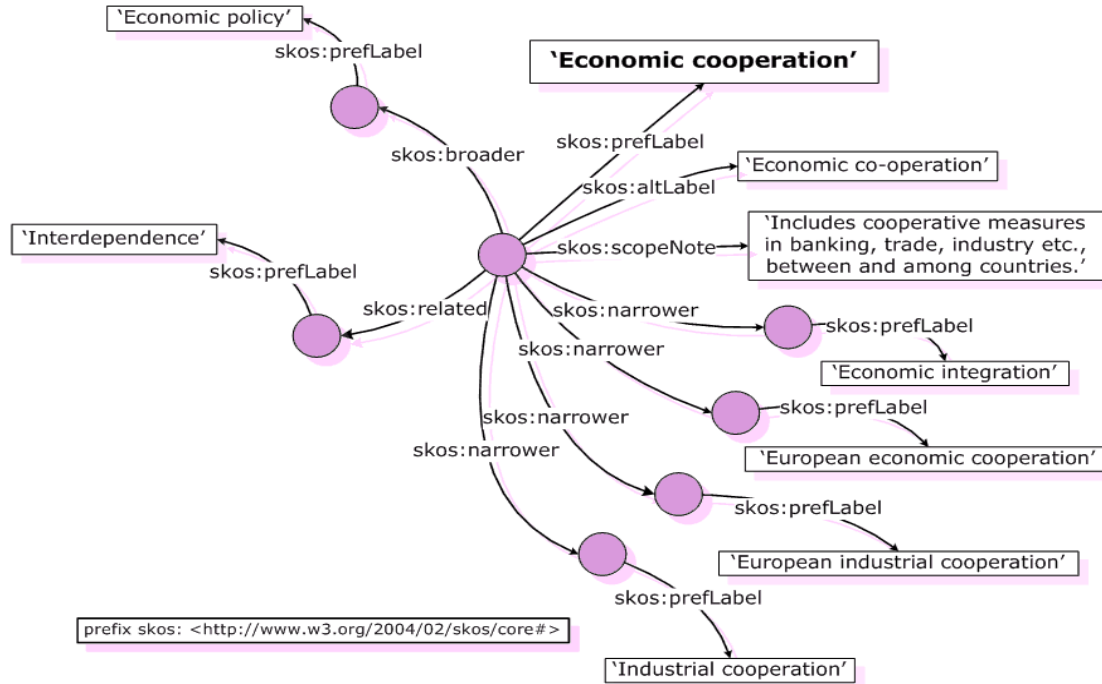
(SKOS Core Guide, 2005)

Mapping concept schemes

- Properties for aligning concepts in different ontologies
- Equality (between schemes)
  - *skos:exactMatch*
  - *skos:closeMatch*
- Other semantic relations (subproperties of concept properties)
  - *skos:broadMatch (< skos:broader)*
  - *skos:narrowMatch (< skos:narrower)*
  - *skos:relatedMatch (< skos:related)*

Other features

skos:Collection

- *For grouping concepts by a criterion. Collections are **not** used for indexing and can be blank nodes:*

- `ex:milk rdf:type skos:Concept ;`
  `            skos:prefLabel "milk"@en .`

- `ex:cowMilk rdf:type skos:Concept ;`
  `               skos:prefLabel "cow milk"@en ;`
  `               skos:broader ex:milk .`

- `ex:goatMilk rdf:type skos:Concept ;`
  `                skos:prefLabel "goat milk"@en ;`
  `                skos:broader ex:milk .`

- `_:b0 rdf:type skos:Collection ;`
  `        skos:prefLabel "milk by source animal"@en;`
  `        skos:member ex:cowMilk ;`
  `        skos:member ex:goatMilk .`

milk
  \<milk by source animal\>
    cow milk
    goat milk

(van Assem, Isaac, 2005)

Other features (2)

skos:OrderedCollection

- Same idea as with collections, but members are represented as an ordered list (instance of class rdf:List)
  - *Ordering information of subconcepts is often needed in, e.g., user interfaces, and collections do not represent it*

skos:notation

- String of characters used to uniquely identify a concept within a concept scheme
  - `ex:semanticWebCourse skos:notation "ME-E4300" .`

Extension for richer label modeling

Modeling labels as resources: SKOS-XL

- Labels are instances of skosxl:Label
  - *Then labels can have properties, too*
- Label instances are referred to using
  - *skosxl:prefLabel*
  - *skosxl:altLabel*
  - *skosxl:hiddenLabel*

SKOS-XL example

```
<Love>
  skosxl:prefLabel <A> ;
  skosxl:altLabel <B> ;
  skosxl:hiddenLabel <C> .

<A> rdf:type skosxl:Label ;
        skosxl:literalForm "love"@en ;
        dct:created "2006-10-03"^^xsd:date .

<B> rdf:type skosxl:Label ;
        skosxl:literalForm "adoration"@en ;
        dct:created "2006-10-03"^^xsd:date .

<C> rdf:type skosxl:Label ;
        skosxl:literalForm "luv"@en ;
        dct:created "2015-05-14"^^xsd:date .
```

More information

Namespace IRIs of SKOS and SKOS-XL contain the specifications in RDF for 1) classes and 2) properties

- http://www.w3.org/2004/02/skos/core#
  - *Direct link to RDF serialization:* http://www.w3.org/2004/02/skos/core.rdf
- http://www.w3.org/2008/05/skos-xl#
  - *Direct link to RDF serialization:* http://www.w3.org/2008/05/skos-xl.rdf

SKOS home page and documentations for a full list of features

For more elaborate ontology modeling there is the OWL standard (topic of the lecture next week)