Towards a relation ontology for the Semantic Web

Dagobert Soergel

With the help of Joseph Easterly

Department of Library and Information Studies
Graduate School of Education
University at Buffalo

Relations in linked data

The problem

- Linked data are built on relationships (properties) usually identified with a URI and a lexical label.
- Relationship types are mostly not standardized (only some are taken from standard name spaces such as DC or FOAF).
- Makes finding and linking data sets difficult.

Relations in linked data

Complementary approaches to a solution

1 **Harvest URIs that refer to the same relationship type** by
following *same-as* links between relationship types;
lexical matching on labels.

2a **Develop a relation ontology** as a support structure in the Semantic Web (**SRO**, also to include entity types), linked to large RO's (CYC, SUMO, FrameNet) with their logical, conceptual, and lexical information.

2b **Encourage description of datasets with** a schema using entity types and relationship types expressed in or mapped to **the SRO**.

3 **The SRO supports harvesting:**
directly: it provides multipe URIs for a relationship type;
indirectly: it provides lexical information that supports lexical matching.

History and future

- This is not a new idea. Relationship type registries have been talked about for a long time in the thesaurus community with no result.
- With the semantic Web the issue becomes more urgent. **One does not get semantics by syntax alone.**
- There are limited registries such as DCMI, FOAF, SKOS etc. but they do not begin to meet the need.

Note: Relationship type = RDF property
but relationships can (and often need to) have more than two arguments (slots).

Example: climb.dataincubator.org

GeographicEntity <*hasLabel*>	**Text*** * also used for Site
GeogrEntity <*hasXCoordinate*>	**GeoCoordinateNumber***[Y, Z]
GeogrEntity <*loggedThrough*>	**Document***
Site <*hasGuidebook*>	**Document**
Site <*hasSiteDescription*>	**Text**
Site <*isNearTo*>	**GeographicEntity**
Site <*geographicallyFaces*>	**CompassDirection**
Site <*hasTerrain*>	**TypeOfRock**
Site <*canBeReachedBy*>	**Route**
Route <*hasDifficulty*>	**DifficultyGrade**

Example: www4.wiwiss.fu-berlin.de/dailymed

Drug	***<hasName>***	***Text***
Drug	***<hasGenericVersion>***	***Drug***
Drug	***<hasActiveIngredient>***	***ChemicalSubstance***
Drug	***<hasInactiveIngredient>***	***ChemicalSubstance***
Drug	***<hasClinicalPharmacologyDescription>***	***Text***
Drug	***<hasIndicationDescription>***	***Text***
Drug	***<hasContraIndicationDescription>***	***Text***
Drug	***<hasAdverseReactionDescription>***	***Text***
Drug	***<hasBoxedWarning>***	***Text***
Drug	***<administeredVia>***	***RouteOfAdministration***

Example: www4.wiwiss.fu-berlin.de/drugbank

DBDrug <*hasName*>	**Text**
DBDrug <*hasGenericName*>	**Text**
DBDrug <has*CASRegistryNumber*>	**URI**
DBDrug <*hasAbsorptionDescription*>	**Text**
DBDrug <*hasBioTransformationDescription*>	**Text**
DBDrug <*hasPharmacologyDescription*>	**Text**
DBDrug <*hasProteinBindRate*>	**Percent**
DBDrug <*hasIndicationDescription*>	**Text**
DBDrug <*hasPossibleDiseaseTarget*>	**Disease**
DBDrug <*hasContraIndicationInsert*>	**Document**

Drug <*hasName*> Text

Drug <*hasGenericVersion*> Drug

Drug <*hasActiveIngredient*> ChemicalSubstance

Drug <*hasClinicalPharmacologyDescr*> Text

Drug <*hasIndicationDescr*> Text

Drug <*hasContraIndicationDescription*> Text

Drug <*administeredVia*> RouteOfAdministration

DBDrug <*hasName*> Text

DBDrug <has*Gen…VersionName*> Text

DBDrug <has*CASRegistryNumber*> URI

DBDrug <*hasAbsorptionDescr*> Text

DBDrug <*hasBioTransformDescr*> Text

DBDrug <*hasPharmacolDescr*> Text

DBDrug <*hasProteinBindRate*> Pct

DBDrug <*hasIndicationDescr*> Text

DBDrug <*hasPossibleDiseaseTarget*> Disease

DBDrug <*hasContraIndicationInsert*> Document

DBDrug <*hasDosageForm*> DosageForm

Building a comprehensive relation ontology

- The relation ontology (SRO) must
  - be **comprehensive and specific** to cover many LOD data sets;
  - be structured into a **well-formed hierarchy**;
  - give much information for each relationship type.

Relation type registry

- The relation ontology should be implemented as a **relation type registry**
  a type of **metadata registry**

- The registry and the relation ontology should be **developed and maintained collaboratively**

Some sources

- **Bottom up:** The linked data sets themselves
- **Top down:** Existing schemes, such as
  - SUMO www.ontologyportal.org
  - FrameNet https://framenet.icsi.berkeley.edu
  - OBO RO http://obofoundry.org/ro/
  - CYC http://opencyc.org/
  - UMLS Semantic Network
    www.nlm.nih.gov/research/umls/META3_current_semantic_types.html
    www.nlm.nih.gov/research/umls/META3_current_relations.html
  - Soergel 1967
  - DCMI and many similar schemes, markup languages

SUMO

part

. component

. piece

. interior part

. member

. sub collection

. subString

SUMO 2

part

The basic mereological relation. All other mereological relations are defined in terms of this one. (part ?PART ?WHOLE) simply means that the object ?PART is part of the object ?WHOLE. Note that, since part is a reflexive relation, every object is a part of itself.

SUMO 3

component

A specialized common sense notion of part for heterogeneous parts of complexes. (component ?COMPONENT ?WHOLE) means that ?COMPONENT is a component of ?WHOLE. Examples of component include the doors and walls of a house, the states or provinces of a country, or the limbs and organs of an animal. Compare piece, which is also a subrelation of part.

piece

A specialized common sense notion of part for arbitrary parts of substances. Quasi-synonyms are: chunk, hunk, bit, etc. Compare component, another subrelation of part.

FrameNet

Being_included

. Be_subset_of

. Part_piece

. . . Part_whole

. . . Part_edge

. . . Part_inner_outer

. . . Part_ordered_segments

. . . Part_orientational

. . . Rest

. . . Shaped_part

. . . Temporary_group

. . . Vehicle_subpart

FrameNet 2

Being_included

Definition:

A Part is profiled as being a subset or constituent subpart of a Whole. This is in contrast to Inclusion frame, wherein the same relation is seen from the point of view of the Whole.

Core:

Part [par] The Part is either a subset or a constituent part of the Whole.

Whole [who] The Whole is the larger entity that includes the Part.

Non-Core:

Inherits from:
Is Inherited by: Be_subset_of, Part_piece

FrameNet 3

Part_piece

This frame is concerned with transparent nouns denoting a Piece of a Substance.

Lexical units: chunk n, clod n, clump n, flake n, fragment n

Part_whole

This frame is concerned with nouns denoting a part or parts of a Whole entity . The Part is not defined relative to the Whole's orientation, center, or edge and is not ordered. We also annotate properties of the Part, e.g. relating to its size.

FrameNet 4

Shaped_part

An entity is a shaped Part of a larger Whole. The Part may correspond in form and function to a part of another entity. For instance, the leg of a table supports the weight of a table like the leg of an animal or person supports their weight. The Orientation of the Part relative to the Whole may be expressed.

OBO Relation Ontology

is_a
part_of
integral_part_of
proper_part_of
located_in
contained_in
adjacent_to

transformation_of
derives_from
preceded_by
has_participant
has_agent
instance_of

OBO Relation Ontology

part_of

Relation properties

[transitive] [reflexive] [anti-symmetric]

Definition

For continuants: C part_of C' if and only if: given any c that instantiates C at a time t, there is some c' such that c' instantiates C' at time t, and c *part_of* c' at t. For processes: P part_of P' if and only if: given any p that instantiates P at a time t, there is some p' such that p' instantiates P' at time t, and p *part_of* p' at t. (Here *part_of* is the instance-level part-relation.)

OBO Relation Ontology

proper_part_of

Relation properties

[transitive]

Definition

As for part_of, with the additional constraint that subject and object are distinct

Soergel 1967

Subsumptive and appurtenance relationships

State, condition, circumstances

Dependence and process, determinative

Motivative aspects

Evaluative aspects

Comparison

Time, space

Soergel 1967

Subsumptive and appurtenance relationships

. **Subsumptive**

. . Type/Kind

. . . Principle / manifestation

. . . Inclusion. Class / element (member) (RT Property)

. . . Implication

. . Whole / part + composite/ constituent

. . . Whole/Part

. . . . Organism/organ

. . . . Organization / subdivision

. . . Object /Material from which it is made. Composite / constituent

. **Property (attribute, quality) / having the property**.

A characterizes B. Capable of

Structure of the registry

Relationships with slots that correspond to roles

A <*hasPart*> B

A has the role of *whole*

B has the role of *part*

ChemicalSubstance A <*treats*> Disease B <*in*> Gender C <*withEffectiveness*> D

A has the role of *agent*

B has the role of *target*

C has the role of *qualifier (population)*

D has the role of *effectSize*

Structure of the registry

The following are the most important relationships needed to represent data about relationship types Rel (identified by a RelID)

- Rel <*hasLabel*> Text (preferred, alternate, language)
- Rel <*hasNumberOfRoles*> Number
- Rel <*hasRole*> (Role, Position)
- Rel <*hasDefinition*> Text
- Rel <*hasAxiom*> Axiom
- Rel <*similarInMeaning*> Rel   special case: data vs pointer to text
- Rel <*subsumes*> Rel (the hierarchy of relationship types)
- There are more kinds of data, see SUMO and FrameNet

Similar data for roles

Registry refers to ontologies

Rel <*occursIn*> (Ontology, Label)

This makes available

Lexical information: many terms in FrameNet

Axioms in the ontology for reasoning, for example

IF A <*givesTo*> (B, C) THEN B <*owns*> C

The registry as basis for an index to datasets

Refer to datasets that use a relationship type

<*relUse*> (RelType, Dataset, Label)

<*relUse*> (<*administeredVia*>*,*DrugBank, *DosageForm*)

Once the relationship URI and/or label is mapped to a relationship in SRO it can be used for retrieval and the information/axioms found about the relationship type can be used for more powerful inference.

Implementation

- Need a **“seed registry”** developed as described above – this would take funding.
- **Introduce as good practice:**
  Data set owners submit <*relUse*> data to the extended registry system, mapping the relationship types they use to the proper relationships in the registry.

  If a relationship type is not found, they submit a new relationship to the registry.

  Some will do this, some will not
- **Need an editing community**

A small start

Developing a small registry of relationship types needed for KOS (30 -50 (-100)) and representing them in RDF ready to use with SKOS

Two ways of collaborating on the Web

1 **Indirect collaboration:**
Use whatever people put on the Web either by itself or combined with other things other people put on the Web.

2 **Direct collaboration:**
Have places where multiple people make contributions within a systematic framework as in social tagging, Wikipedia, or the proposed relationship type registry.

Conclusion

A relationship type registry would bring the promise of linked open data closer to reality.

We need a Wikipedia-type organization to get there.