# Organizing knowledge in Semantic Web

Miroslav Blaško

miroslav.blasko@fel.cvut.cz

October 16, 2019

# Outline

# Knowledge Organization Systems

# Why do we need to organize knowledge ?

- Indexing
- Retrieval
- Organization and navigation

# Basics

# About KOS

**KOS** = **K**nowledge **O**rganization **S**ystems

- a generic term used to cover all types of schemes for organizing information and promoting knowledge management
- includes following subtypes [**hodge2000systems**] :
  - **Term lists** (authority files, glossaries, dictionaries, gazetteers)
  - **Classification and categories** (subject headings, classification schemes, taxonomies, categorization schemes)
  - **Relationship lists** (thesauri, semantic networks, ontologies)
- another classification [**hedden2010accidental**] defines **Controlled vocabularies** which include all types of KOS except of highly structured semantic networks and ontologies

# Types of KOS

# What is Controlled vocabulary ?

- **Controlled vocabulary** in its simplest form is a restricted list of words or terms for some specialized purpose
- it is mostly used for descriptive cataloging, tagging or indexing
- ”controlled” is used because
  - only terms from the list may be used for the subject area covered
  - only under certain specific conditions may controlled vocabulary change and grow which is responsibility of controlled vocabulary editor, not user

# Term lists – Authority file

- **Authority file** is a list of terms that are used to control the variant names for an entity or the domain value for a particular field (names for countries, individuals, and organizations)
- properties
  - might include non/preferred terms
  - generally does not include deep organization or complex structure
- examples
  - the Library of Congress Name Authority File
  - the Getty Geographic Authority File

# Term lists – Glossary

- **Glossary** is a list of terms, usually with definitions
- properties
  - the terms may be from a specific subject field or from a particular work
  - the terms are defined within a specific environment and rarely include variant meanings
- examples
  - the Environmental Protection Agency (EPA) Terms of the Environment

# Term lists – Dictionary

- **Dictionary** is an alphabetical list of words and their definitions
- properties
  - might include variant senses where applicable
  - more general in scope than glossary
  - might provide synonyms and related words (through the definitions) but it is not explicitly structured or grouped by concept

# Term lists – Gazetteer

- **Gazetteer** is a list of place names
- properties
  - each entry may also be identified by feature type (e.g. river, city, or school)
  - geospatially referenced gazetteer provides coordinates for locating the place
- examples
  - the U.S. Code of Geographic Names

# Classification and categories – Subject Heading

- **Subject Heading** is a scheme type providing a list of controlled terms to represent the subjects of items in a collection
- properties
  - the list of terms can be extensive and cover a broad range of subjects
  - the list's structure is generally very shallow
- examples
  - the Medical Subject Headings (MeSH)
  - the Library of Congress Subject Headings (LCSH)

| | |
|---|---|
| **Title:** | Anti-doping : the fight against ***performance-enhancing drugs*** in sport / [European Commission, Directorate-General for Research]. |
| **Publisher:** | Luxenbourg : Office for Official Publications of the European Communities, [2001?] |
| **Description:** | 1 sheet (folded) : col. ill. ; 21 cm. |
| **Series:** | European research in action |
| **Portion of title:** | Fight against ***performance-enhancing drugs*** in sport |
| **Notes:** | Cover title. |
| | "KI-38-01-601-EN-D." |
| **OCLC:** | (OCoLC)ocm48580907 |
| **Subjects:** | Athletes--Drug use. |
| | Doping in sports. |
| **Other:** | European Commission. Directorate-General for Research. |

Example of LCSH for performance enhancing drugs (see "Athletes–Drug" and "Doping in sports") taken from `http://http://madcat.library.wisc.edu/`

# Classification and categories – Taxonomy

- **Taxonomy** is collection of terms organized into a hierarchical structure
- properties
  - each term in a taxonomy is in one or more parent-child relationships to other terms in the taxonomy
  - parent-child relationship in a taxonomy can have different parent-child relationships in a taxonomy (e.g., whole-part, genus-species, type-instance)
  - generally each parent-child relationships have a single parent of the same type (otherwise it is called poly-hierarchy)
  - traditional taxonomies use "is a kind of" relationship

## Note

- the term "taxonomy" is sometimes used in broader sense to refer any means of organizing concepts of knowledge. To limit it to hierarchical structure within this context term "hierarchical taxonomy" is used.
- we will use this narrower sense when talking about taxonomies.

# Relationship lists – Thesaurus

- **Thesaurus** is a list of controlled terms and relationships among them. The terms are organized in groups of synonyms representing concepts.
- properties
  - relationships commonly found in thesaurus include hierarchy (broader/narrower term), equivalence (synonym), and association or relatedness (related term).
  - dictionary-thesaurus includes all the associated terms that can be potentially used in place of term entry in various contexts
  - information retrieval thesaurus on the other hand is used in all contexts within covered domain of interest

**ABSORPTION**

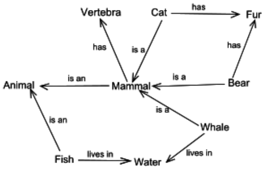
The retention and conversion into another form of energy of rays, waves, or particles by a substance.

UF ABSORPTIVE PROPERTIES
BT **SORPTION**
NT **BIOLOGICAL ABSORPTION**
**RESONANCE ABSORPTION**
**TWO PHOTON ABSORPTION**
**X RAY ABSORPTION ANALYSIS**

Representation of term "absorption" taken from DTIC Thesaurus

# Relationship lists – Semantic network

- **Semantic network** is organization of terms and concepts in a structure not as hierarchy but as a network (web)
- properties
  - the concepts are represented as nodes of the network while relationships are edges that connects them
  - the relationships are generally richer than in thesaurus. They may include specific whole-part, cause-efect, or parent-child relationships.
- examples
  - Princeton University's WordNet




Example of simple semantic network about Mammals taken from

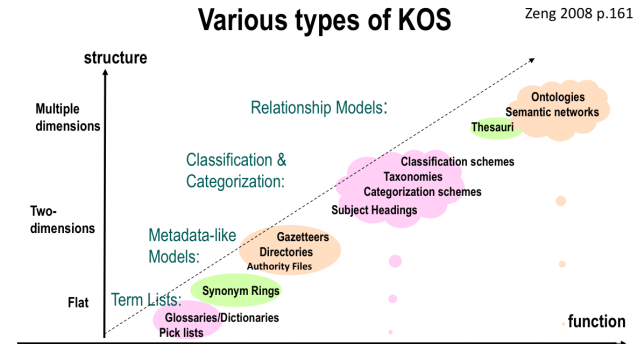
`http://en.wikipedia.org/wiki/Semantic_network`

# Relationship lists – Ontology

- **Ontology** is structural framework for organizing knowledge that uses controlled vocabulary expressed in an ontology representation language
- properties
  - the language has a grammar for using vocabulary terms to express something meaningful within a specified domain of interest
  - the grammar specifies formal constraints on how terms in the ontology's controlled vocabulary can be used together
  - concepts and relations are organized into "is kind of" hierarchies
  - compared to semantic network it can have axioms and rules
- examples
  - Descriptive Ontology for Linguistic and Cognitive Engineering
  - Basic Formal Ontology

**Note**

By the term "ontology" we refer here mainly to "formal ontology", i.e. ontology with main purpose to negotiate meaning.




Major functions

| | | | | | | |
|---|---|---|---|---|---|---|
| eliminating ambiguity | xxx | | xxx | xx | xxxx | xx |
| controlling synonyms | | xxxx | xxx | xx | xxxx | xx |
| establishing relationships: hierarchical | | | x | xxxx | xxx | xxx |
| establishing relationships: associative | | | | | xxxx | xxxxx |
| presenting properties | | | | | | xxxxx |

Figure taken from [**lei2008knowledge**] visualize types of KOS, arranged according to the degree of controls introduced (from natural language to controlled language) and the strength of their semantic structure (from weakly structured to strongly structured), corresponding to the major functions of KOS.

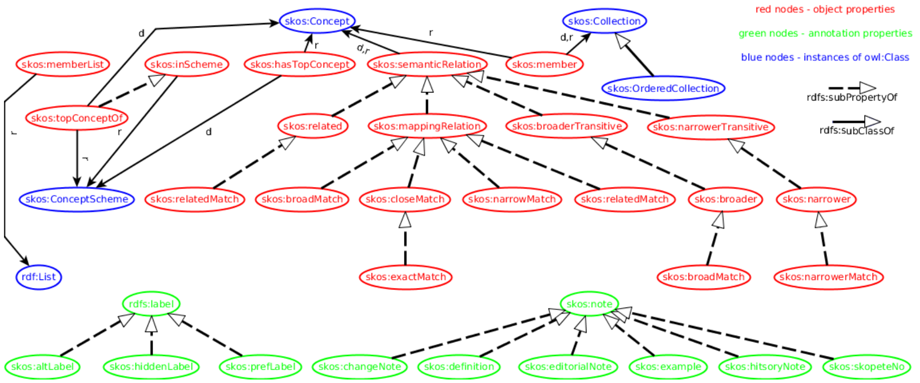
# SKOS

# What is SKOS ?

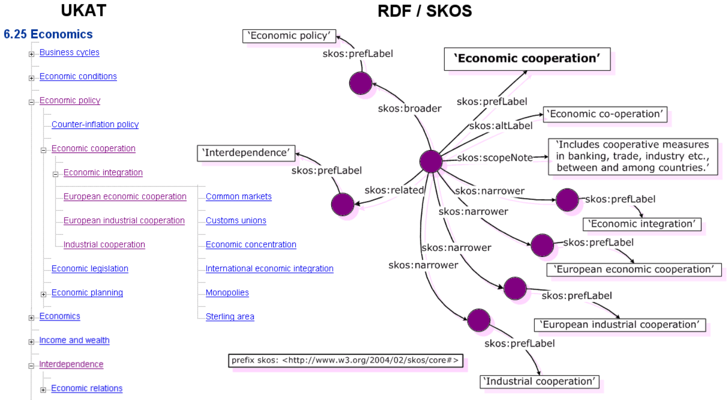
**SKOS** = **S**imple **K**nowledge **O**rganization **S**ystems

- knowledge organization system that become official W3C recommendation in August 2009
- it allows to express thesauri, classification systems, subject headings, lists ..
- it represent KOS in RDF in simple way
- SKOS data model is formally defined as an OWL Full ontology

# Partial SKOS Model – Axiomatic Triples



# Example of SKOS representing thesaurus




Portion of UK Archival Thesaurus (UKAT) and its relevant representation in SKOS taken from `http://www.mkbergman.com/374/an-intrepid-guide-to-ontologies/`.

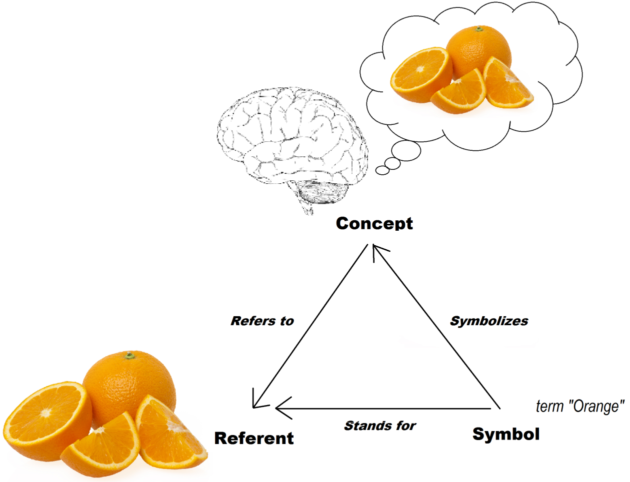
# SKOS relevant materials

- SKOS homepage
  `http://www.w3.org/2004/02/skos/`
- SKOS Primer
  `http://www.w3.org/TR/skos-primer/`
- SKOS Reference
  `http://www.w3.org/TR/skos-reference/`
- SKOS Use Cases and Requirements
  `http://www.w3.org/TR/2009/NOTE-skos-ucr-20090818/`

# Ontology

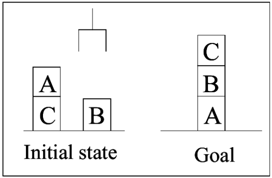
# Basics

# What is concept ?




The meaning triangle according to Ogden&Richards, 1969

# What is conceptualization ?




**Conceptualization-1**

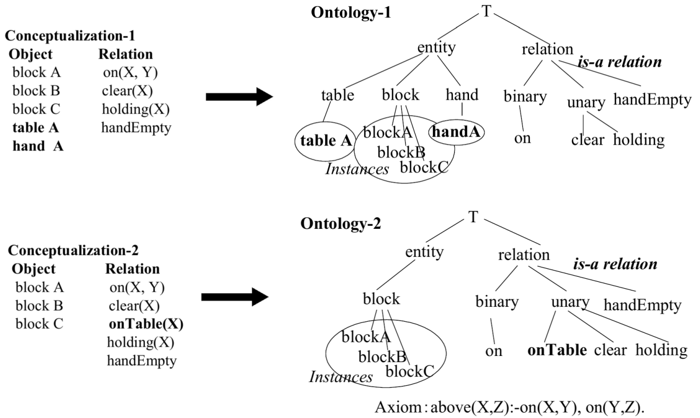
| Object | Relation |
|---|---|
| block A | on(X, Y) |
| block B | clear(X) |
| block C | holding(X) |
| **table A** | handEmpty |
| **hand A** | |

**Conceptualization-2**

| Object | Relation |
|---|---|
| block A | on(X, Y) |
| block B | clear(X) |
| block C | **onTable(X)** |
| | holding(X) |
| | handEmpty |

The block world problem and its two different conceptualizations taken from [**mizoguchi2003part**]. Objects **table A** and **hand A** does not exist in **Conceptualization-2** as they are substituted by **onTable** relation.

# Ontology based on conceptualization




Ontologies created based on **Conceptualization-1** and **Conceptualization-2** taken from [**mizoguchi2003part**].

# Definitions

- **Conceptualization** is set of objects which an observer thinks that they exist in target world (world of interest). It provides backbone of the conceptual structure of such world.
- **Ontology** is “explicit specification of conceptualization” [**gruberOntology**].
    - It contains hierarchically organized structure of concepts and relations between them. Such structure defines meaning of objects appearing in the target world.
    - It is type of KOS where each term from controlled vocabulary is put into some specific place within its complex structure.
    - It is declarative description of fundamental understanding of the target world.

## Note

In case when OWL2 is used to represent an ontology – OWL classes represents concepts, OWL individuals represent concrete objects, and combination of OWL classes and OWL properties represent relations of the target world. **From now on we will use OWL2 terminology whenever it is possible**.

# Using ontology in the most correct way

# But what conceptualizations are the best to use with ontology ?

- There aren't the best conceptualizations ! There are only conceptualization that fit the best for the given use of the ontology and its formal language capabilities.
- On the other hand, ontological engineering is currently viewed as a challenge to enabling *knowledge sharing* and *reuse* which other fields of AI failed to realize. In this sense we will call ontology more "**ontological**" or more "**ontologically correct**" if it suits better for this purpose.
- To create correct ontologies first we need to understand fundamental issues of building ontologies such as distinction between *is-a* and *part-of* relations, distinction between *classes*, *instances* and is-a relation etc.

# Importance of is-a and part-of relation

- Is-a hierarchy of classes allows us to use terms with different specificity. But can we count or identify objects using hierarchy properly ? Consider query that we want to ask an ontology :
- **How many blue things are in this room ?**

# Importance of is-a and part-of relation

- Is-a hierarchy of classes allows us to use terms with different specificity. But can we count or identify objects using hierarchy properly ? Consider query that we want to ask an ontology :
- **How many blue things are in this room ?**
  - Blue table has many blue things on it.
  - Property "blue" does not allows us to count objects.
  - **To answer this query we need proper categorization of things.**
  - We need to understand what are object *wholes* and what are object *parts*.

# Importance of is-a and part-of relation

- Is-a hierarchy of classes allows us to use terms with different specificity. But can we count or identify objects using hierarchy properly ? Consider query that we want to ask an ontology :
- **How many blue things are in this room ?**
    - Blue table has many blue things on it.
    - Property "blue" does not allows us to count objects.
    - **To answer this query we need proper categorization of things.**
    - We need to understand what are object *wholes* and what are object *parts*.
- **How many tables are in this room ? How many furniture are in this room ?**

# Importance of is-a and part-of relation

- Is-a hierarchy of classes allows us to use terms with different specificity. But can we count or identify objects using hierarchy properly ? Consider query that we want to ask an ontology :
- **How many blue things are in this room ?**
  - Blue table has many blue things on it.
  - Property "blue" does not allows us to count objects.
  - **To answer this query we need proper categorization of things.**
  - We need to understand what are object *wholes* and what are object *parts*.
- **How many tables are in this room ? How many furniture are in this room ?**
  - Classes can be organized into hierarchies according to different properties.
  - **To answer this query we need proper way of class organization.**
  - We need to understand how is *identity* of an object propagated through the use of *is-a* links.

# Proper representation of is-a hierarchy

- *is-a hierarchy* should not be simple classification of classes, but rather represent inheritance of *essential property* of each class.
- correct is-a hierarchy reveals the intrinsic structure of the target world which help people to understand in-depth the class without its definition.
- ad-hoc classification are usually purpose-dependent thus less shareable. "Correct ontology" should not be knowledge base for problem solving but rather foundation of knowledge base for various purposes.
- is-a hierarchies should be shareable and stable backbone of the knowledge structure.
- with proper is-a hierarchy we can safely propagate properties of super-class to properties of its sub-classes

# Proper representation of part-of relations

- *part-of relation* is used to represent a thing as a whole which is composed of few parts and is usually transitive.
- There are at least 5 types of *part-of relations* [**mizoguchi2004part**] :
    - Functional *part-of*, where contribution of the part to the whole is functional (e.g. wheel is *part-of* bike)
    - Qualification *part-of*, where instances of the relation must have qualification/role in order to become part of whole. (e.g. husband *part-of* married-couple)
    - Spatial/temporal relation *part-of*, where instances of the relation must satisfy spatial/temporal constraints to a become part of the whole. (e.g. tree *part-of* forest, mountain part-of mountains)
    - Staff *part-of*, where the whole is stuff. (e.g. a piece of pie *part-of* pie)
    - Material *part-of*, where instances are materials of the whole. (e.g. glass *part-of* cup)

**Note**

Although *part-of relation* is essential for building ontologies, OWL2 does not provide vocabulary to represent it. Thus, semantics of *part-of relation* must be defined within the ontology.

# Difference between part-of and is-a relationship

- It does not make sense for two classes be in relation part-of and is-a relation at the same time.
- Consider following axioms in in context of plant:
  1. normal operation *is-a* operation;
     restoration operation *is-a* operation
  2. normal operation *part-of* operation;
     restoration operation *part-of* operation
- **How can we resolve this issue ?**
  (Hint : differentiate between event/process and action)

# Difference between instance-of and is-a relationship

- Class represent set of concrete things in target world
- Instance is element of those sets, i.e. they represent undividable elements
- Thus it does not make sense for an ontological entity to be instance and class within ontology at the same time.
- Consider following axioms about cars:
  my personal car *instance-of* Ford;
  Ford *instance-of* car brand
- **How can we resolve this issue ?**
  (Hint : differentiate between Ford within both axioms, e.g. who is owner of Ford for each axiom)
- What happens if we represent above axioms within *OWL2* ontology ?

# Classification of ontologies

# Different types of ontologies

- There are many classification of ontologies. We will distinguish only four types :
  - **Top-level ontologies –** describe very general concepts such as event, object, action. They are independent of a particular problem or domain.
  - **Domain ontologies –** describe the vocabulary related to a generic domain such as cultural tourism, or medicine.
  - **Task ontologies –** describe the vocabulary related to a generic task or activity such as selling, or diagnosing.
  - **Application ontologies –** describe classes that depend on both a specific task and a specific domain. The classes within ontology often correspond to roles played by domain entities performing some task.
- Domain and task ontologies typical specializes top-level ontologies.
- Application ontologies typically specializes both task and domain ontologies.

# Study materials

- Tutorial on Ontological Engineering (from Mizoguchi Laboratory) http://www.ei.sanken.osaka-u.ac.jp/japanese/tutorial-j.html
  - Part "What is an ontology?" – written by T.Gruber, contains definition of ontology from multiple perspectives
  - Part 1 – details about conceptualizations, definition of ontology, types of ontologies
  - Part 3 – details about "ontologicaly correct“ ontologies (part-of, is-a relations etc.)