

*KNOWDIVE*



**KGE - Knowledge Graph Engineering**

## **Formal Modelling**

Teleontology = Teleology + Ontology

**Fausto Giunchiglia**

# Contents

- 1 Teleology - Recap**
- 2 Ontology**
  - The notion - from the SoA
  - Types of ontologies - from the SoA
  - Ontologies - our own perspective
- 3 Teleontology**
- 4 Teleontology modeling in iTelos**
  - The process
  - An example
- 5 Standards and tools**
  - Standard and tools from the SoA
  - LiveKnowledge - The KGE catalog
  - The UKC - Language annotation
- 6 References**

# Teleology - Recap

- **Teleologies** are **enriched ER models** with the difference that they are designed to explicitly model the **purpose**:
  - 1 allow to define the **spatio-temporal context**.
  - 2 allow to model data and object properties as **objects, functions** and **actions**.
  - 3 Teleologies (equivalent to ER models) are extended to **teleontologies** (equivalent to EER models) only after the explicit selection of an ontology and with a clear purpose.
- Teleologies provide the basis for a general model for the **heterogeneity of knowledge**.

# ER Model Limitations - SOLVED !!!

The following two main weaknesses of ER models are solved via modelling a **teleology**:

- What situational context is the ER model modeling?

**Solved by making explicit the spatio-temporal context in which the entities of a teleology are contained.**

- Where do data and object properties come from?

**Solved by making explicit the functions and actions via which the entities of a teleology interact within the spatio-temporal context.**

# ER to EER Model - STILL UNSOLVED

The design of EER models, as extensions of ER models, should be driven by the need of facilitating reuse.

- Where do the **extra etypes of the EER model** come from?

The step from an ER model to an EER model is completely undefined.

The etypes in ER models do not conform to a general theory about *what exists* in the world around us.

As a result, the etype hierarchies in EER models are developed in a focus-less fashion, without a clear methodology.

This results in hindrance of:

- (i) data and knowledge reuse, and consequently
- (ii) lack of semantic interoperability.

# ER to EER Model - proposed solution

The solution proposed:

- 1 ...
- 2 ...
- 3 Teleologies (equivalent to ER models) are extended to **teleontologies** (equivalent to EER models) only after the explicit selection of an **ontology** and with a clear purpose.

How to do it?

- Make explicit reference to a general representation of the world (an ontology)
- Choose the best possible ontology among those which are available
- Extend the current teleology with information coming from the selected ontology, based on the purpose, thus generating a teleontology

# Contents

- 1 Teleology - Recap
- 2 **Ontology**
  - The notion - from the SoA
  - Types of ontologies - from the SoA
  - Ontologies - our own perspective
- 3 Teleontology
- 4 **Teleontology modeling in iTelos**
  - The process
  - An example
- 5 **Standards and tools**
  - Standard and tools from the SoA
  - LiveKnowledge - The KGE catalog
  - The UKC - Language annotation
- 6 **References**

# What is an Ontology (philosophy)?

“Ontology” has different meaning in different communities...

Ontology: philosophical discipline which deals with the nature and structure of “reality.”

- the science of “being qua being,” i.e., the study of attributes that belong to things because of their very nature (Aristotle), which focuses on the nature and structure of things per se,
- independently of any further considerations, and even independently of their actual existence
  - e.g. Ontology of unicorns and other fictitious entities



# What is an Ontology (Computer Science)?

- **An ontology:** in computer science, a special kind of information object or computational artifact
  - formally model the structure of a system, i.e., the relevant entities and relations that emerge from its observation, and which are useful to our purposes.
- **Example:** Provide an ontological representation of a company with all its employees and their interrelationships
  - entities organized in concepts (unary predicates) and relations (binary predicates) a taxonomy of concepts (generalization/specialization hierarchy)

E.g.:

- **Person, Manager, and Researcher**
  - Person “super concept” of Manager, and Researcher
- **Cooperates-with** can be considered a relevant relation holding between persons.
- A concrete person (e.g. Mario Rossi) working in a company would then be an instance of its corresponding concept.
- **Cooperates-with(Mario Rossi, Giorgio Bianchi)** states that Mario Rosso cooperates with Giorgio Bianchi in its work.

# What is an Ontology (The key aspects)?

- “explicit specification of a conceptualization” [Gruber, 1993]
- “formal specification of a shared conceptualization” [Borst, 1997]
- “An ontology is a formal, explicit specification of a shared conceptualization” [Studer et al., 1998]
- But....
  - What is a conceptualization?
  - What is a proper formal, explicit specification?
  - Why is ‘shared’ of importance?

# What is a Conceptualization?

- Formal structure of (a piece of) reality as perceived and organized by an agent, independently of:
  - the vocabulary used
  - the actual occurrence of a specific situation
- Different situations involving same objects, described by different vocabularies, may share the same conceptualization
- "mela", "apple": different terms for the same conceptualization...

# Formal, Explicit Specification

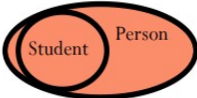
- We need to use a language to refer to the elements of a conceptualization
  - the language **commits** to a conceptualization
- Problem: a logical signature can be interpreted in arbitrarily many different ways
- Once we commit to a certain conceptualization, we have to make sure to only admit those **models** which are **intended** according to the conceptualization.
  - the intended models of a relation predicate will be those such that the interpretation of the predicate returns one of the various possible extensions (one for each possible world) of the conceptual relation denoted by the predicate.

# Why is SHARED of importance?

- Sharing whole conceptualizations may not be possible (private to the mind of the individuals)
- Sharing approximations of conceptualizations based on a limited set of examples, and showing the actual circumstances where a certain conceptual relation holds
- Without such minimal sharing, the benefits of having an ontology are limited
  - ontology may turn out useless if it is used in a way that runs counter the understanding of the primitive terms in the appropriate way.
- Any ontology will always be less complete and less formal than it would be desirable in theory.

# The Ontology Building Block: IS-A Relation

## Is-a Relation

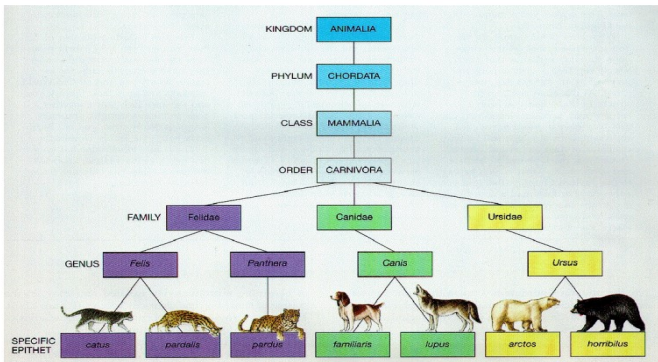
- **is-a relation:** binary relation between concepts (not individuals)
- **Examples: Student is-a Person, Air Pollutant is-a Pollutant**
  - Informal meaning: all the students are persons (or all the individuals that are students are also persons); if something is an air pollutant, it is also a pollutant
- In set-theoretical terms:

A Venn diagram consisting of two overlapping orange ovals. The left oval is labeled 'Student' and is entirely contained within the right oval, which is labeled 'Person'. This visualizes that every student is also a person.
- In FOL terms:
  - $\forall x(\text{student}(x) \rightarrow \text{person}(x))$

# IS-A Hierarchy: Example Taxonomy

## Is-a hierarchy

- **taxonomy**: a hierarchical organized subject-based classification system
  - typically depicted in a tree-like structure
- **is-a hierarchy**: taxonomy of concepts organized according to the is-a relation.

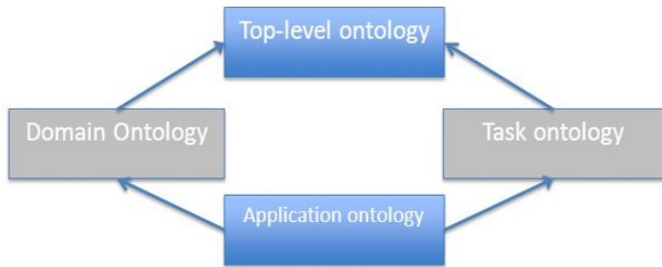


# Contents

- 1 Teleology - Recap
- 2 **Ontology**
  - The notion - from the SoA
  - Types of ontologies - from the SoA
  - Ontologies - our own perspective
- 3 Teleontology
- 4 **Teleontology modeling in iTelos**
  - The process
  - An example
- 5 **Standards and tools**
  - Standard and tools from the SoA
  - LiveKnowledge - The KGE catalog
  - The UKC - Language annotation
- 6 **References**



# Types of Ontologies



Guarino (1998) categorization

- Top-level ontologies describe very general concepts like space, Time, etc which are independent of a particular problem or domain.
- Domain ontologies and task ontologies describe, respectively, the vocabulary related to a generic domain (like medicine, or automobiles) or generic task or activity (like selling) by specializing the terms introduced in the top-level ontology.
- Application ontologies describe concepts depending both on a particular domain and task, which are often specializations of both the related ontologies.

# Domain Ontologies

- ❑ Domain ontologies are reusable in a given **specific domain** (medical, law, automobile, etc.).
- ❑ These ontologies provides **vocabularies** about concepts within a domain and their relationships, about the activities taking place in that domain.
- ❑ The concepts in domain ontologies are usually **specializations** of concepts already defined in top-level ontologies

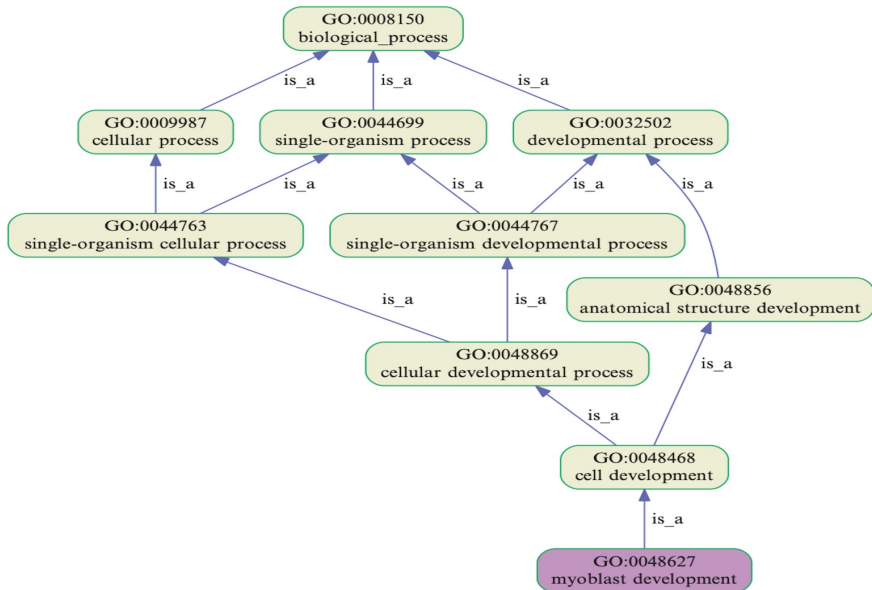
# Example: schema.org (fragment)

- Organization
  - Airline
  - Consortium
  - Corporation
  - EducationalOrganization
    - CollegeOrUniversity
    - ElementarySchool
    - HighSchool
    - MiddleSchool
    - Preschool
    - School
  - FundingScheme
  - GovernmentOrganization
  - LibrarySystem
  - LocalBusiness
    - AnimalShelter
    - ArchiveOrganization
    - AutomotiveBusiness
      - AutoBodyShop
      - AutoDealer

# Example: An Activity Ontology



# Example: The GENE Ontology



# Domain ontology - Characteristics

- *Sharing common understanding* of the structure of information in the selected domain (among people or software agents)
- Enabling *reuse of domain knowledge*
- Making *explicit the domain assumptions* underlying an implementation. (key feature: It is thus easier to change these assumptions easily if our knowledge about the domain changes).

# Contents

## 1 Teleology - Recap

## 2 **Ontology**

- The notion - from the SoA
- Types of ontologies - from the SoA
- Ontologies - our own perspective

## 3 Teleontology

## 4 Teleontology modeling in iTelos

- The process
- An example

## 5 Standards and tools

- Standard and tools from the SoA
- LiveKnowledge - The KGE catalog
- The UKC - Language annotation

## 6 References

# Ontologies - our own perspective

- The language of an ontology is written in a conceptual language which eliminates the ambiguities of NLP
- Given a language, an ontology is an entity type (etype) graph (ETG). An ETG is a set of etypes which associated with a set of objects and data properties (similarly to ER models)
- The ISA hierarchy is modeled as specialization hierarchy where more specific etypes inherit properties (similarly to EER models)
- Being designed with a more general purpose than a teleology (never made explicit in the SoA ontologies), an ontology provides a general view, and extra etypes and properties
- An ontology provides a more general view of the domain and extra etypes and properties (more general, but not only) which can be used to enrich a teleology, towards sharing and reusability.



# Contents

- 1 **Teleology - Recap**
- 2 **Ontology**
  - The notion - from the SoA
  - Types of ontologies - from the SoA
  - Ontologies - our own perspective
- 3 **Teleontology**
- 4 **Teleontology modeling in iTelos**
  - The process
  - An example
- 5 **Standards and tools**
  - Standard and tools from the SoA
  - LiveKnowledge - The KGE catalog
  - The UKC - Language annotation
- 6 **References**

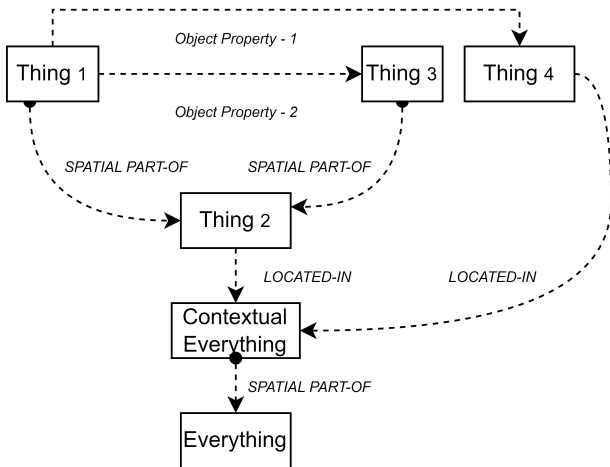
# Teleontology - Motivation

- Given a teleology modelled via objects, functions and actions, there is a distinct need to confer teleological concepts with ontological semantics. For example, which existence category does the teleological concept *park* or *student* belong to?
- In other words, there is a distinct need to embed abstract-level (domain) semantics during the process of building EER models from ER models.
- Conforming to a reference ontology will enable reusability and shareability of teleologies in similar spatio-temporal contexts.
- Conforming to a reference ontology will also enable semantic interoperability of teleologies in similar spatio-temporal contexts.

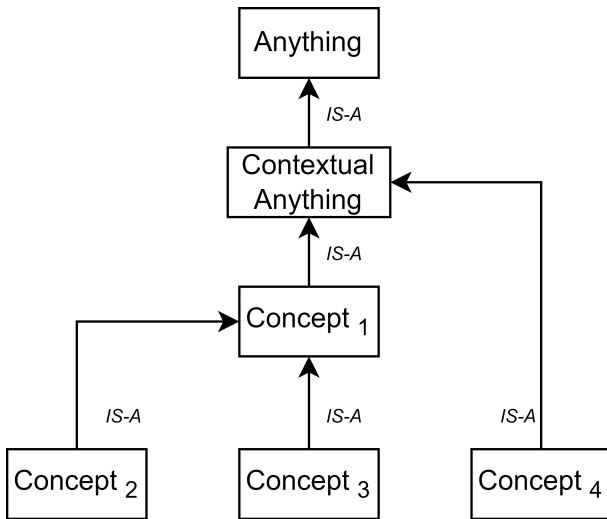
# Teleontology - Definition

- **Teleontology:** A teleontology is a teleology extended with more etypes which grounds it to what exists in the world around us. Given a teleology generated according to a purpose, a reference ontology is selected/designed to which the teleological concepts are aligned. Concretely axiomatized as an OWL RDF/XML file.
- It is, thus, a graph encoding **IS-A, SPATIAL PART OF, LOCATED IN** relations + functions and actions (in the form of object and data properties) amongst the potential concepts to be modelled.
- A teleontology models the specific concepts of a domain characterized by their (object and data)properties via the teleology.
- While at the same time, it semantically constrains the ontological meaning of the concepts in the teleology via their link to the chosen reference ontology.

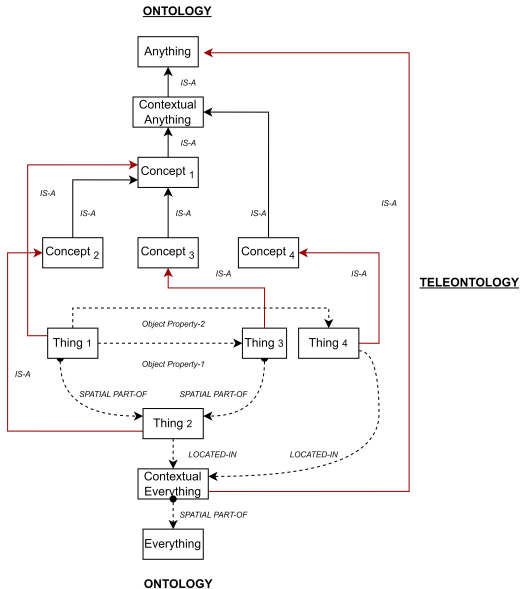
# Teleology - General Structure



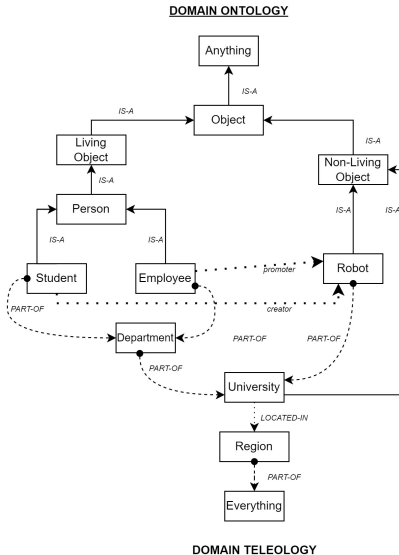
# Ontology - General Structure



# Teleontology = Teleology + Ontology



# Teleontology - Example



# Contents

- 1 Teleology - Recap
- 2 **Ontology**
  - The notion - from the SoA
  - Types of ontologies - from the SoA
  - Ontologies - our own perspective
- 3 Teleontology
- 4 **Teleontology modeling in iTelos**
  - The process
  - An example
- 5 **Standards and tools**
  - Standard and tools from the SoA
  - LiveKnowledge - The KGE catalog
  - The UKC - Language annotation
- 6 **References**



# Teleontology modeling - the process

- **Step (1):** Generate teleology [INFORMAL MODELLING]
- **Step (2):** Choose reference teleontology
- **Step (3):** Align teleology to (tele)ontology, possibly generating a new revised teleontology
- **Step (4):** Annotate the new terms with namespace information
- **Step (5):** Extract schema from teleontology

**STEP 2:** Resources from available catalogues (see lecture "W2.L2.Reuse Problem"). But see also **LiveKnowledge** catalog below

**STEP 2:** In this class knowledge resources suggested as part of the project definition)

**STEP 4:** See the **UKC** below.

# Teleontology modeling - step 2

## Step (2): Choose reference teleontology

- 1 Choose reference teleontology
  - If suitable teleontology exists return teleontology,
  - otherwise choose reference ontology
- 2 Choose reference ontology
  - If suitable ontology exists return teleontology built from ontology and teleology
  - otherwise choose choose reference standard
- 3 Choose reference standard
  - If suitable standard exists return minimal ontology built from standard and teleology
  - otherwise model customized minimal ontology (EER model) from teleology

**NOTICE** you might have to build your own customized ontology by suitably integrating relevant fragments (e.g., *person* from schema.org, *patient* from FHIR etc.) from several reference ontologies.

# Teleontology modeling - step 5

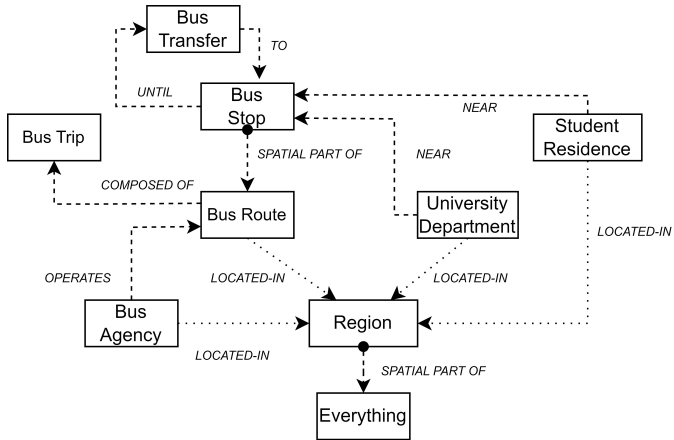
**Step (4):** Build schema by:

- 1 dropping the teleological root *Everything*
- 2 dropping the ontological root *Anything*
- 3 deleting entities/properties (e.g., entities and properties which are too general and/ or not relevant) from the teleontology according to specific application purpose. Typical candidates are common on even core etypes and properties.
- 4 adding entities/properties to the teleontology according to the specific application purpose. Typical examples are those contextual etypes and properties which were deleted before step 2 because very application specific - It would have been hopeless to try to match them with reference teleontologies. And also useless to save them in reusable teleontologies.

# Contents

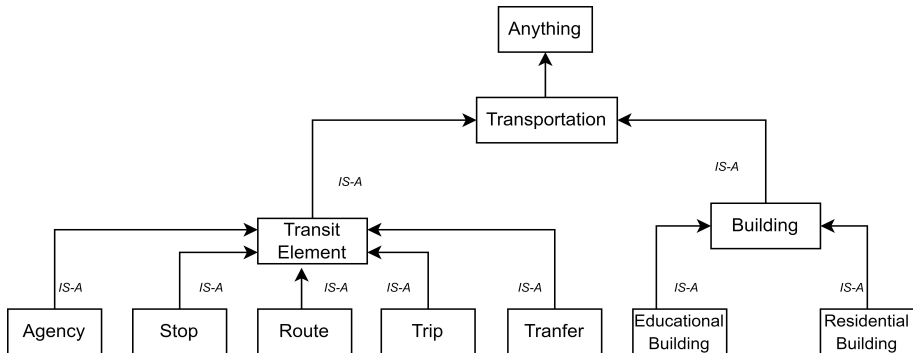
- 1 Teleology - Recap
- 2 **Ontology**
  - The notion - from the SoA
  - Types of ontologies - from the SoA
  - Ontologies - our own perspective
- 3 Teleontology
- 4 **Teleontology modeling in iTelos**
  - The process
  - An example
- 5 **Standards and tools**
  - Standard and tools from the SoA
  - LiveKnowledge - The KGE catalog
  - The UKC - Language annotation
- 6 **References**

# Step (1) - Generate Teleology



**TELEOLOGY**

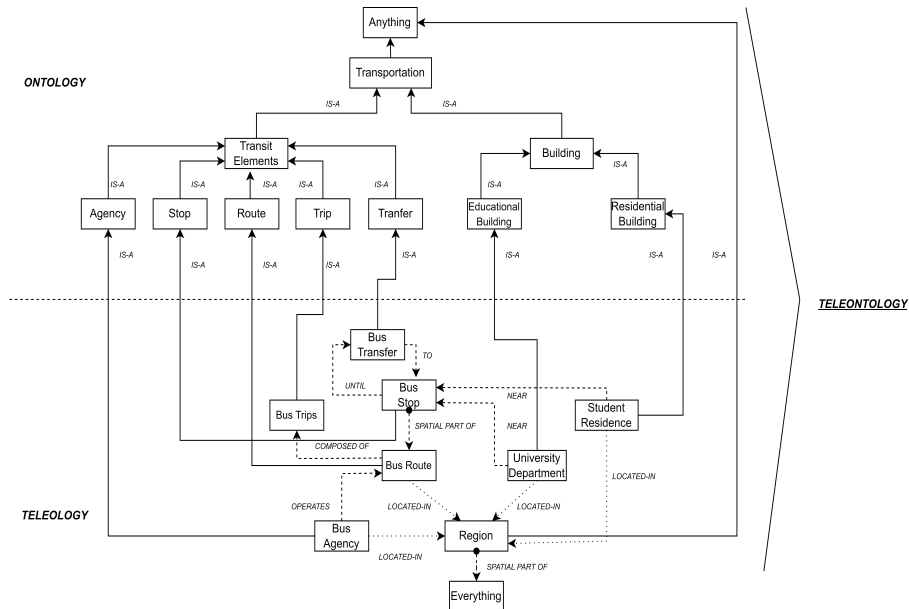
# Step (2) - Choose/Build (Tele)Ontology



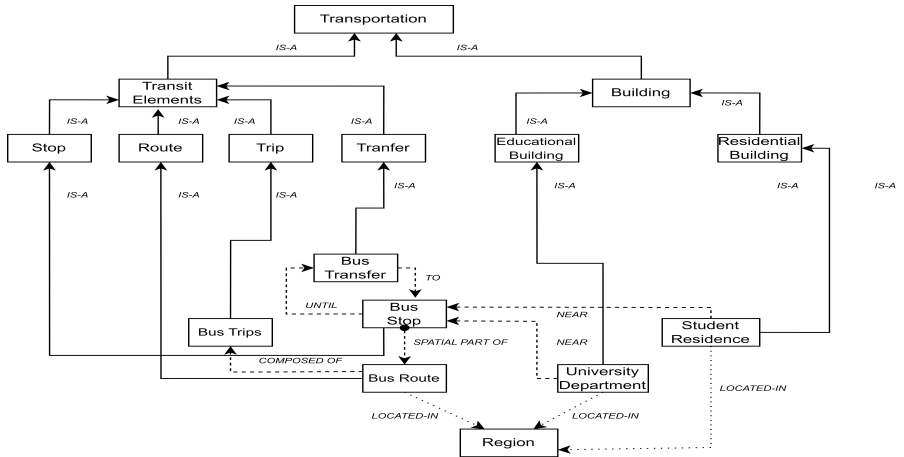
## ONTOLOGY

**NOTE:** The above ontology is customized with the *Transit Element* fragment from the GTFS Ontology and the *Building* fragment from the INSPIRE Directive.

# Step (3) - Align to form Teleontology



# Step (4) - Build Schema



**NOTE:** 'Bus Agency' deleted from previous teleontology following application-specific purpose + Anything & Everything Dropped.



# Contents

- 1 Teleology - Recap
- 2 **Ontology**
  - The notion - from the SoA
  - Types of ontologies - from the SoA
  - Ontologies - our own perspective
- 3 Teleontology
- 4 Teleontology modeling in iTelos
  - The process
  - An example
- 5 **Standards and tools**
  - Standard and tools from the SoA
  - LiveKnowledge - The KGE catalog
  - The UKC - Language annotation
- 6 References

# Standards and Tools

- **Namespaces** ensuring unique names (we use existing ones plus our own namespace for new terms you will introduce).
- **RDF** (Resource Description Framework): The W3C endorsed triple-based data modeling language (sub, property, obj). All information is stored and represented in the RDF.
- **OWL** (Web Ontology Language): The schema language, or knowledge representation (KR) language, of the ontologies. OWL enables you to define concepts composably so that these concepts can be reused as much and as often as possible.
- **SPARQL** (SPARQL Protocol and RDF Query Language): The W3C endorsed language employed to query ontologies. It is specifically designed to query data across various systems.
- **Protégé 5.5.0**: The open source software tool to model and edit ontologies.

# Contents

- 1 Teleology - Recap
- 2 Ontology
  - The notion - from the SoA
  - Types of ontologies - from the SoA
  - Ontologies - our own perspective
- 3 Teleontology
- 4 Teleontology modeling in iTelos
  - The process
  - An example
- 5 Standards and tools**
  - Standard and tools from the SoA
  - LiveKnowledge - The KGE catalog
  - The UKC - Language annotation
- 6 References

# LiveKnowledge - The KGE catalog

- LiveKnowledge, currently [LiveSchema](#), is a high-quality catalog of reference ontologies, teleologies and teleontologies.
- Currently, it aggregates ontological schemas from several state-of-the-art repositories such as [Linked Open Vocabulary](#), [FINTO](#) etc.
- You may find the reference ontologies (for **STEP (2)**) from the LiveSchema catalog via filtering through *domain* (e.g., geography, healthcare).
- The teleologies, ontologies and teleontologies that you produce will be published in the LiveSchema/LiveKnowledge catalog.

## Search Datasets



Popular tags

[metadata](#)[methods](#)[geography](#)[services](#)

## Catalogs

- [Linked Open Vocabulary](#)

LOV started in 2011, in the framework of a French research project (<http://datalift.org>). Its main initial objective was to help...

[720 datasets](#)

- [DERI Vocabularies](#)

DERI Vocabularies is a URI space for RDF Schema vocabularies and OWL ontologies maintained at DERI, the Digital Enterprise Research...

[81 datasets](#)

- [KnowDive Vocabularies](#)

## Services



Dataset Uploader



Knowledge Embedder



FCA Generator



Cue Generator



Visualization Generator



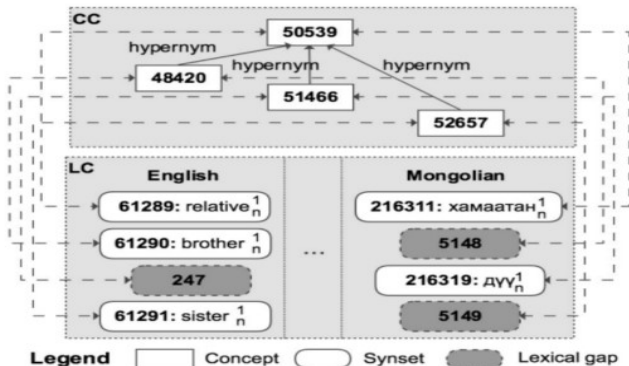
Query Catalog

# Contents

- 1 Teleology - Recap
- 2 **Ontology**
  - The notion - from the SoA
  - Types of ontologies - from the SoA
  - Ontologies - our own perspective
- 3 **Teleontology**
- 4 **Teleontology modeling in iTelos**
  - The process
  - An example
- 5 **Standards and tools**
  - Standard and tools from the SoA
  - LiveKnowledge - The KGE catalog
  - The UKC - Language annotation
- 6 **References**

# The UKC - Universal Knowledge Core

## UKC Architecture



Giunchiglia, Batsuren, Freihart. In Proceedings of CiCling (2018)

# UKC - Definitions

## Definitions

- 1 Synset:** Multiple meanings of a word are codified as a lexicalized concept, also called a synset, consisting of a (possibly incomplete) set of synonymous words.
- 2 Word Sense:** In linguistics, a word sense is one of the meanings of a word. For example, the word *play* has 50 different meanings.
- 3 Concept Core (CC):** The Concept Core (CC) is the UKC representation of the world and it consists of a semantic network where the nodes are language independent concepts.



# UKC - Definitions (Contd.)

## Definitions

- 1 Language Core (CC):** We talk of the Language Core (LC), meaning the component that, in the UKC, corresponds to the set of words, senses, synsets, glosses and examples supported by the UKC.
- 2 GID:** Global Identifier (GID) is the unique identifier assigned by the UKC CC to a concept.
- 3 Princeton WordNet (PWN):** PWN is a large lexical database of English. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. Synsets are interlinked by means of conceptual-semantic and lexical relations.

## Step (4) - Language Annotation

The UKC enabled Annotation transforms the informal terms of the teleontology to formal concepts. Three main objectives:

- Representing each informal concept in the teleontology with the unique identifier (GID) from UKC, thus, rendering each concept formal and absorbing conceptual diversity.
- The teleontology vocabulary has both already existing concepts in the UKC CC, and, new concepts absent in the UKC CC. We align the existing concepts with their equivalents in the UKC CC and extend the UKC CC by adding (only) the new concepts (with new GIDs), thus, absorbing conceptual as well as language diversity.
- Due to the LC, each concept can also be rendered multilingually, thus opening up the possibility to adapt the teleontology in any language or culture.

## Step (4) - Annotation Methodology

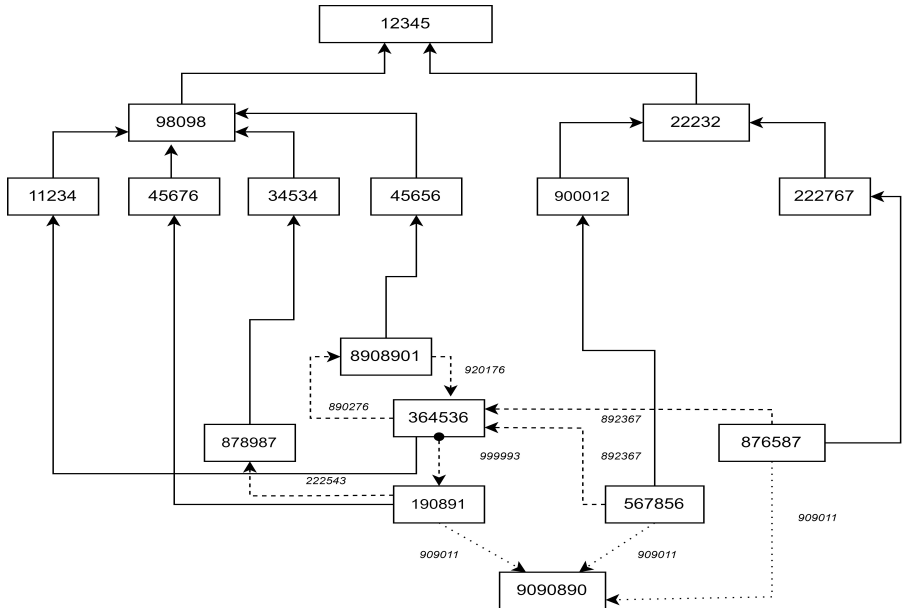
The general methodology for language annotation, as semi-automatically performed via an annotator tool is as follows:

- Each term (concept) is selected from amongst the classes, relations and attributes (all terms from all these hierarchies are considered; one at a time) from the teleontology
- The term is (semantically) searched in the UKC Knowledge Base (UKC KB) via the annotator tool, and the step will result in one of the following two scenarios:-
  - Scenario 1 (S1): Synonymous Match between the teleontology concept and a synonymous concept found in the UKC KB
  - Scenario 2 (S2): No Existing Semantic Match between the teleontology concept and any concept in the UKC KB (that means, New Concept !!!)

## Step (4) - Annotation Methodology (Cont.)

- (S1) In case of a Synonymous Match, the concept is mapped to the semantically synonymous concept in the UKC CC, and thus is assigned the latter's GID. The key step is to choose the semantically synonymous concept amongst the options suggested by the annotator tool.
- (S2): In case of a No Existing Semantic Match, the concept, via the annotator tool, is declared as a new concept with respect to the UKC CC.
  - This step is incremental as all concepts from all hierarchies are sequentially examined.
  - It results in alignment of all the concepts with their equivalent counterparts in the UKC KB, and thus each concept annotated with a unique GID
  - As a result, multiple (tele)ontologies are mapped to the single UKC (the single, unified background knowledge).

# Step (5) - Example



# Contents

- 1 **Teleology - Recap**
- 2 **Ontology**
  - The notion - from the SoA
  - Types of ontologies - from the SoA
  - Ontologies - our own perspective
- 3 **Teleontology**
- 4 **Teleontology modeling in iTelos**
  - The process
  - An example
- 5 **Standards and tools**
  - Standard and tools from the SoA
  - LiveKnowledge - The KGE catalog
  - The UKC - Language annotation
- 6 **References**

# References

- Giunchiglia, Fausto, and Mattia Fumagalli. "Teleologies: Objects, actions and functions." International conference on conceptual modeling. Springer, Cham, 2017.
- Giunchiglia, Fausto, and Mattia Fumagalli. "Concepts as (Recognition) Abilities." In FOIS, pp. 153-166. 2016.
- Noy, Natalya F., and Deborah L. McGuinness. "Ontology development 101: A guide to creating your first ontology." (2001).
- <https://cambridgesemantics.com/blog/semantic-university/intro-semantic-web/>
- <http://protegeproject.github.io/protege/>
- W3C Ontology Standards Page - [https://www.w3.org/2001/sw/wiki/Main\\_Page](https://www.w3.org/2001/sw/wiki/Main_Page)



KGE - Knowledge Graph Engineering



**Fausto Giunchiglia**



**Formal Modelling**

Teleontology = Teleology + Ontology