ONTOCOMMONS AND DOME 4.0

ONTOSTYLOGY ALIGNMENT

Emanuele Ghedini (UNIBO)

Main Contributing Partners: UNIBO, CNR, SINTEF, UCL, ENIT, GCL, UKRI, ATB, UiO

This projects has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreements No. 953163 and No. 958371
DOME 4.0 relies on an **ontology based architecture** with a

- **Semantic Data Exchange Ontology** (called the **dataset ontology**, based on EMMO and DCAT) (Lead: UNIBO, UCL, SINTEF)

- **Ecosystem Ontology** (extending existing ontologies and vocabularies, such as European Virtual Marketplace Ontology and European Science Vocabulary) (Lead: UKRI)
Objectives of the Data Set Ontology:

1) To **develop an ontology for semantic exchange of data** between data providers and consumers. The semantic data exchange ontology will be lightweight in terms of logical complexity and number of entities and should be **based on existing established standards and ontologies** (e.g., EMMO).

2) Interactions with the **project funded from the NMBP-39-2020-CSA (OntoCommons)** call will provide guidelines for such development to provide a high level of generality and applicability, shared by a larger community.

3) Develop an **ontological syntactic representation** of data with an extensible, light-weight data structure ontology capable of mapping between syntactic representations and thereby supporting the exchange of data.

https://doi.org/10.5281/zenodo.7784934
DOME 4.0 and OntoCommons

Bare Data → RDFS Documented Data (keywords as semantic meaning providers) → OWL-DL Documented Data (data are part of an ontology framework that provides a richer semantic framework) → Data Interoperability (data are semantically placed into a large ontology framework, enabling interoperability)

Semantic Enhancement
From RDF to OWL 2 DL

- There exist several RDF vocabularies and schemas aimed to document data and their use in different scenarios, that are already widely used and understood by several communities. These RDF schemas includes the Dublin Core Metadata Initiative collection of terms (DCMI Metadata Terms), the Data Catalog Vocabulary (DCAT), the Friend of a Friend Vocabulary Specification (FOAF) and the PROV Data Model ontology (PROV-O).

- These schemas rely on RDF concepts, and in some cases on OWL 2 concepts and provide a very flexible way to document data and their usage. However, the permissivity of the RDF language prevents the introduction of more sophisticated axiomatisations to impose constraints that are commonly used in the definition of a highly expressive ontology.

- While such permissivity facilitates a fast deployment of metadata schemas developed ad hoc for the documentation of specific domain cases, it prevents the building a more semantically rich environment, that requires a language (e.g., OWL 2 DL) and some syntactic constraints to grant computability (i.e., reasoning).

- Moreover, it would be beneficial to embed such RDF vocabularies into a larger ontological environment, to use the information conveyed by such terms in an environment that connects the existing terms towards other knowledge domains.

from DOME 4.0 D3.1 - “Semantic data exchange ontology”, UNIBO, UCL, SINTEF
Basic Metadata

- Defined a list of data documentation concepts.
- Mapped the EMMO 1.0.0-beta4 to DCAT and other relevant RDFS vocabularies, to build a semantically enhanced data documentation environment, compatible with other H2020 initiatives (OntoCommons, OntoTrans, OpenModel, SimDOME)

<table>
<thead>
<tr>
<th>Label</th>
<th>Definition</th>
<th>RDFS Schema References</th>
</tr>
</thead>
<tbody>
<tr>
<td>DataSet</td>
<td>DCAT: A collection of data, published or curated by a single agent, and available for access or download in one or more representations.</td>
<td>dcat:Dataset (rdfs:Class) subclass of dcatalog:Resource (rdfs:Class)</td>
</tr>
<tr>
<td>Title</td>
<td>DCAT: A name given to the resource.</td>
<td>dcterms:title (rdf:Property) with range rdfs:literal</td>
</tr>
<tr>
<td>Keyword</td>
<td>DCAT: A keyword or tag describing the resource.</td>
<td>dcterms:keyword (rdfs:Property) with range rdfs:literal</td>
</tr>
<tr>
<td>Creator</td>
<td>DCAT: An entity responsible for making the resource.</td>
<td>dcterms:creator (rdfs:Property) with range dcterms:Agent (rdfs:Class)</td>
</tr>
<tr>
<td>Publisher</td>
<td>DCAT: An entity responsible for making the resource available.</td>
<td>dcterms:publisher (rdfs:Property) with range dcterms:Agent (rdfs:Class)</td>
</tr>
<tr>
<td>Issued</td>
<td>DCAT: Date of formal issuance of the resource.</td>
<td>dcterms:issued (rdfs:Property) with range rdfs:literal</td>
</tr>
<tr>
<td>License</td>
<td>DCAT: A legal document giving official permission to do something with the resource.</td>
<td>dcterms:license (rdfs:Property) with range dcterms:LicenseDocument (rdfs:Class)</td>
</tr>
<tr>
<td>Source</td>
<td>DCAT: A related resource from which the described resource is derived.</td>
<td>dcterms:source (rdfs:Property)</td>
</tr>
<tr>
<td>Homepage</td>
<td>FOAF/DCAT: The homepage property relates something to a homepage about it. (a public Web document usually available in HTML).</td>
<td>foaf:homepage (owl:ObjectProperty) with range foaf:Document (rdfs:Class)</td>
</tr>
<tr>
<td>Description</td>
<td>DCAT: An account of the resource.</td>
<td>dcterms:description (rdfs:Property)</td>
</tr>
</tbody>
</table>
Properties OWL 2 DL Restrictions

Several terms in the DCAT/DCTERMS/FOAF schemas are associated with the `rdf:Property` type, giving the user the freedom to choose the OWL 2 resource type (data, object or annotation) to which the property points.

For example, a `dcterms:creator` can refer to a textual annotation (e.g. “John Smith”) or to an individual of type `dcterms:Agent`.

However, to build an OWL 2 DL compliant mapping enabling reasoning, there is the need to specify one specific type of property between datatype, object, or annotation property. The mapping will then distinguish between the different types of properties according to the expected range and domain.

In OWL 2 Full, object properties and datatype properties are not disjoint. In OWL 2 DL the set of object properties and datatype properties are disjoint, to enable decidable reasoning. See https://www.w3.org/TR/2012/REC-owl2-syntax-20121211/#Typing_Constraints_of_OWL_2_DL.
Semantic Enhancement

Non-Logical Statement.
Reasoner does not make use of this information.

Non-Logical Statement.
Reasoner can make use of this information through data properties, but with limited expressivity.

Logical Statement.
Reasoner makes use of this information. Possible to connect this knowledge to other concepts and make inferences.

Semantic Enhancement metadata are ontological entities fully exploiting OWL 2 DL capabilities.
Dataset Mapping

This mapping enables a direct relationship between an EMMO and DCAT data concepts, whereby the `emmo:DataSet` is a restriction of the `dcat:Dataset` since it requires that at least two `emmo:Datum` are present in the dataset, while the `dcat:Dataset` is not clear about the definition of the term “collection”.

Within the EMMO, the distinction between data and datum terms, enables the use of the expressivity power of mereotopology for the representation of the content of a dataset.

The EMMO nominalistic approach requires that individuals of the `emmo:EncodedData` are actual material expressions of data, thus restricting the mapping to `dcat:Dataset` entities that refers to actual data material basis.
This design choice recognises that it is possible to provide data with **syntactic keywords**, giving complete descriptive freedom to the user, and **semantic keywords**, that are restricted to IRIs pointing to valid OWL 2 DL entities.

The `emmo:hasTypeKeyword` data property is aimed to define the type of the data, i.e. what the data physically is (e.g. a book, a csv file, a picture). This suggests that a dataset can take any physical form.

The `emmo:hasIsAboutKeyword` reflects something about the data via a semiotic process stating that the data “is about” something else. Here we make use of the EMMO semiotic approach with a domain `emmo:SemioticObject`.

The `emmo:hasToolKeyword` is the missing link between the seemingly thin metadata layer imposed both by DCAT dataset and the deep content of a dataset (i.e. the actual raw data stored in the dataset) referring to specific computational tools (e.g. a spreadsheet, or a simulation package, or a user provided script) that are able to decipher the syntactic information.
Creator Mapping

The EMMO mapping of `dcterms:creator` restricts the scope of the relation within the data field, restricting the domain to `emmo:Data`, and defining `emmo:Agent` as sub class of `dcterms:Agent`. We also introduce the `emmo:DataCreator` class to specify the type of agent involved in the data creation process, and the data creation process itself by the `emmo:Creation` class.

The semantic enhancement provided by the EMMO is related to the use of the Holistic and Persistence perspectives, that provide mereotopological relations to deal with the concepts of e.g. process, role, and participant.

These concepts are peculiar to most of the Top Level Ontologies that are not expressed in the existent RDF schemas for data documentation.
Benefit of an Ontology (in short)

**Reasoning:**
- possibility to apply constraints to data documentation improving the quality of your databased documentation (consistency)
- inferring new knowledge (e.g., types, relations) from existing one

**Interoperability:**
- between disciplines, providing a network of relations between entities, and placing them under different perspectives

**Expressivity:**
- taxonomy, annotations, and relations provides a way to express meaning for a dataset, much powerful than a simple keyword

An ontology bubbles up knowledge towards the user, reducing the need for data analysis and complex queries.
Benefit of an Ontology (EMMO)

Each perspective provides a covering axiom for all causal objects according to a specific categorisation.
EMMO perspectives draw a clear separation between data and their meaning.

**Information**, usually defined as “data with meaning” is an important concept for EMMO.

Mereotopology (not present in RDFS vocabularies, but in several TLOs) enables description of the syntax of a dataset.

EMMO forces the users to consider information only the data that has been semiotically connected to another entity.

*e.g. to connect physical properties to the relevant physical entity, extremely relevant for material characterization or modelling*
Example of Metadata Origin Documentation

Using parthood we can extract some relevant data from the source (e.g. title), while leaving the bulk of the data outside the ontology.

Using semiosis we can add data not part of the source (e.g. book rating) and documenting also who declared that particular data as relevant for the object.
Physical Quantities are represented as **syntactical structures** of numbers and strings, and stored in RDFS format.

QuantitativeProperties are physical quantities that are connected to a material through a **semitotic process** of simulation.

---

3.2 kg
Beyond the EMMO

The structure of the TRO-OWL ontology modules is summarized here, also including the Middle Reference Ontology, that hosts the alignments of a selected set of MLOs and bridge concepts. The Meta ontology is clearly depicted as the combination of the mappings between the three selected TLOs.
Protégé Environment

The TRO-OWL framework has been published publicly in the GitHub repository at:

https://github.com/OntoCommons/OntologyFramework

It will be continuously updated in the course of the OntoCommons Project:

- Accessible
- Easy to maintain
The framework is Protégé-compatible and will offer the end user the possibility to navigate through the TLOs, their mappings and the lower-level ontologies that depend on them.

The OWL mappings, derived from the more rigorous FOL alignment, connect TLOs’ taxonomic trees... and not just that.

The TRO-OWL will constitute a development framework for semantic web ontologists to build and test OntoCommons compliant ontologies.

The connected roots of the three TLOs
DOLCE/BFO mapping expressed in the OWL 2 DL language and visualised through the Protégé OWLViz Plugin.
The content of the template (now a table), can be expressed using more flexible formats (e.g. XML, JSON) and documented within the RDFS version of the ontology.
CHEBI to TLOs

CHEBI is a chemistry MLO, not aligned with a TLO.

Thanks to the ChemicalEntity and Atom Bridge Concepts we can align it to the TRO and hence to all TLOs.
Conclusions

• The DOME 4.0 Data Set Ontology can open the gate towards ontology frameworks that provides well documented ontology concepts to semantically enrich your data (OCES specifications).

• An ontologized data set can answer complex questions such as: “Which software may provide such data so that I can build a workflow?”, “Which real world object type this data stands for?”, “What do you mean with viscosity in this dataset?”, “Are there workflows that make use of this dataset type?”

• The Data Set Ontology is respectful of DCAT and people that does not like ontologies! You may or may not use it to enhance your dataset and live with simple syntactic keywords (with all their pros and cons).

The OCES comprises also Technical Specifications and Tools for optimal ontology development and documentation.
This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreements No. 953163 and No. 958371.