Deliverable D3.1 - “Semantic data exchange ontology”

Responsible Partner:	UNIBO	28.02.2022
Contributor(s):	Emanuele Ghedini (UNIBO) Adham Hashibon (UCL) Jesper Friis (SINTEF)	28.02.2022
Reviewer(s):	<Reviewer names (partner)>	28.02.2022
Coordinator:	CMCL Innovations	28.02.2022
Dissemination Level:	Public	
Due Date:	M15 - 28 February 2022	
Submission Date:	28.02.2022	

This document is part of a project that has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 953163. It is the property of the DOME 4.0 consortium and do not necessarily reflect the views of the European Commission.

Project Profile

Programme	Horizon 2020
Call	H2020-NMBP-TO-IND-2020-twostage
Topic	DT-NMBP-40-2020 Creating an open marketplace for industrial data (RIA)
Project number	953163
Acronym	DOME 4.0
Title	Digital Open Marketplace Ecosystem 4.0
Start Date	December 1st, 2020
Duration	48 months

Document History

Version	Date	Author	Remarks
V0.1	11/02/2022	Emanuele Ghedini	First Draft Structure
V0.2	18/02/2022	Emanuele Ghedini	Concept and Schema List
V0.3	18/02/2022	Jesper Friis Emanuele Ghedini Adham Hashibon	Mapping Description
V0.4	21/02/2022	Emanuele Ghedini	Incremental
V0.5	22/02/2022	Emanuele Ghedini	Incremental
V0.6	23/02/2022	Emanuele Ghedini	Incremental
V0.7	23/02/2022	Adham Hashibon	Review and outlook
V0.8	28/02/2022	Silvia Chiacchiera	Review
V0.8	28/02/2022	Emanuele Ghedini	Review and correction
V0.9	28/02/2022	Adham Hashibon	Final Review
V1.0	28/02/2022	Wieteke van Balen	Final reformatting and submission

Executive Summary

A semantic enhancement of existing metadata schemas for data documentation and exchange is here provided, to cover the fundamental concepts needed by the DOME4.0 project. Concepts from the DCAT RDF vocabulary (and from the RDF schema which DCAT depends on) for the description of datasets and data services (and catalogues) are here mapped in an enriched semantical framework provided by the EMMO. This is done by imposing constraints that enable the usage of the original DCAT concepts into an OWL 2 DL framework.

A mapping ontology from DCAT together with an ontology extension of the EMMO to deal with specific data concepts is provided in a public repository. Results of this mapping will be ported into the *OntoCommons* Top Reference Ontology. This will make it usable also by other ontological approaches (e.g. BFO, DOLCE). This work presents both means to bridge between the state of the art widely used top level ontologies and widely used web standard metadata schemas on the one hand, and a novel means of ontologising such metadata schemas and bringing them to a high-level semantic logic on the other hand.

A first example of combination of syntactic description of data and semantic mapping, is hereby therefore provided.

Table of Contents

Executive Summary 3
Table of Contents 4
List of Figures 5
List of Tables 5
1. Introduction 6
1.1 Objectives 6
1.2 Terminology 6
1.3 Methodology 6
2. RDFS Data Exchange Vocabularies 8
2.1 DCTERMS 8
2.2 DCAT 8
2.3 PROV-O 10
3. Data Set Basic Concepts 11
3.1 Concept List 11
4. EMMO Mapping 15
4.1 Methodological Approach 15
4.2 EMMO Perspectives 16
4.3 Concept Mappings 18
4.3.1 DataSet 18
4.3.2 Title 20
4.3.3 Keyword 20
4.3.4 Creator 23
4.3.5 Publisher 25
4.3.6 25
4.3.7 Issued 26
4.3.8 License 26
4.3.9 Source 27
4.3.10 URI 29
4.3.11 Homepage 30
4.3.12 Description 31
5. Future Developments 33
6. Syntactic Description 35
7. Networking Actions 36
8. Acknowledgement 37

List of Figures

Figure 1 Dependency diagram for the mapping axioms. 7
Figure 2 The DCAT (shown is Version 3) schema relies on DCTERMS, FOAF, SKOS, etc. Image from https://www.w3.org/TR/vocab-dcat-3/images/dcat-all-attributes.svg 9
Figure 3 The three Starting Point classes and the properties that relate them. From https://www.w3.org/TR/prov-o/#description-starting-point-terms. 10
Figure 4 The rdf:Property and their type restriction in OWL 2 DL as data, object or annotation properties 16
Figure 5 The EMMO class hierarchy up to the first level beyond Perspective class. 17
Figure 6 Example of semantic enhancement of basic data documentation. 18
Figure 7 EMMO mapping of dcat:Dataset 19
Figure 8 EMMO mapping of dcterms:title 20
Figure 9 EMMO mapping of dcat:keyword 22
Figure 10 EMMO mapping of dcterms:Agent and dcterms:creator 24
Figure 11 EMMO mapping of dcterms:Agent and dcterms:publisher 25
Figure 12 EMMO mapping of dcterms:issued 26
Figure 13 EMMO mapping of dcelements:rights and dcterms:license 27
Figure 14 EMMO mapping of dcterms:source 28
Figure 15 EMMO mapping of dcterms:identifier 29
Figure 16 EMMO mapping of foaf:homepage and foaf:Document 30
Figure 17 EMMO mapping of dcterms:description 32
Figure 18 An example of the top level information model of the IDS showing more intricate relations between erntities than e.g., DCAT. From 33
Figure 19 EMMO/DOME 4.0 enables seamless mapping between third party standards. The semantic enrichment enables seamless semantic interoperable exchange. 34
Figure 20 Syntactic description and semantic mapping of data sets. 35

List of Tables

Table 1 List of DOME4.0 data documentation concepts and their reference to existing RDF schemas. ... 14

1. Introduction

1.1 Objectives

The objectives of Task 3.1 of the DOME4.0 project are:

1. To develop an ontology for semantic FAIR 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 (e.g. IDS) and ontologies (e.g. EMMO)
2. To cater for FAIR-ness elements which are manifested by an exchange of the needed information to identify the source of data (findability), its type, application context, and access rights (accessibility), means to exchange and decipher the data (interoperability), and means to reuse it (reusability).
3. To Interact 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
4. To 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.

1.2 Terminology

We will use the word *concept* to refer to the abstract notion that we define and represent using RDFS/OWL entities. A concept is usually elucidated through *definitions*, i.e., articulated text aimed to explain in natural language the real-world entities that the concept addresses. The word *label* will indicate the short text used to tag a specific concept (e.g. through rdfs:label, skos:prefLabel, or directly in the IRI). The word *term* is usually used as a synonym for concept using the label as reference. For example, with the sentence "the term Agent in DCAT", we will refer to the concept in DCAT that has the word Agent in the IRI.

In principle, a concept is *not* necessarily related to a label, and a label can be used to address more than one concept. A term specifies a concept and its referred (preferred) label according to a specific namespace designation, for this reason the use of the word term *must* always refers to a specific vocabulary providing its definition.

1.3 Methodology

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). We will refer generically to such schemas using the abbreviation **RDF-DEV** (RDF-based Data Exchange Vocabulary).

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. Moreover, it would be beneficial to embed such RDF-DEV 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.

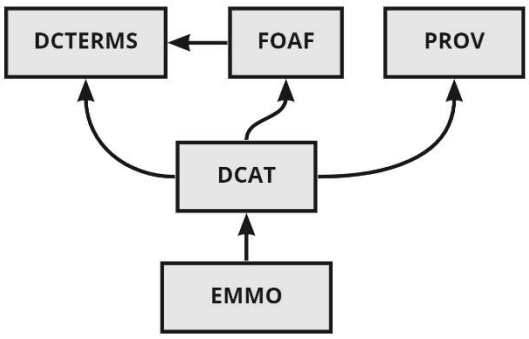

Figure 1
Dependency diagram for the mapping axioms.

The methodology adopted here is to rely on the existing RDF-DEV, but at the same time enrich them semantically by providing a mapping with a **Top-Level Ontology** that is part of the **OntoCommons Top Reference Ontology** level. This approach aims to facilitate the injection of data documentation that is compliant with the RDF-DEV, to facilitate the migration of already existing data documentations into a larger ontological framework (the EMMO), and at the same time to facilitate the usage and understanding by developers already trained and skilled on such material.

To achieve that, a one-way mapping based mainly on rdfs:subClassOf relations has been provided from EMMO to RDF-DEV concepts, meaning that the EMMO concepts are OWL 2 DL compliant restrictions of the wider RDF schemas ones. In this way every EMMO type will both refer to an existing RDF-DEV and will provide a semantical enhancement within the EMMO ontology.

Users that are not interested in a semantically enriched data documentation approach can use the EMMO concepts directly linked with the RDF-DEV without dealing with any higher-level mappings of these concepts into the EMMO, while users that want to exploit a semantically rich framework may use the EMMO concepts as part of a larger knowledge framework that will provide an RDF-DEV counterpart through mapping for the usage in less semantically demanding scenarios.

The RDF version of the mapping will be made available publicly under the DOME4.0 repository at https://github.com/DOME-4-0/data-set-ontology. It will be maintained and expanded according to the needs of the project and the evolution of the EMMO during the overall duration of DOME4.0

2. RDFS Data Exchange Vocabularies

Here we briefly list the RDFS-DEV from which we will select the terms relevant for the DOME4.0 scope.

2.1 DCTERMS

Dublin Core[1] is a set of properties (vocabulary) for associating metadata with resources. It was originally developed to describe library resources, particularly documents, video files, books etc., and later extended for web resources, and it has also been used to describe a variety of other physical and digital resources.

The Dublin Core metadata initiative includes the fifteen terms in the Dublin Core Metadata Element Set in addition to a larger set of properties, classes, datatypes, and schemes. Together, they are collectively referred to as "DCMI metadata terms" or "Dublin Core terms" (DCTERMS) for short.

DCTERMS are expressed in RDF vocabularies. Each term is identified with a Uniform Resource Identifier (URI), which is a global identifier usable in Linked Data. Built into the Dublin Core standard are definitions of each metadata element – like native content standard – that state what kinds of information should be recorded where and how. Associated with many of the data elements are data value standards such as the DCMI Type Vocabulary and ISO 639 language codes.

We will hereafter refer to the RDF representation of the http://purl.org/dc/terms/ namespace published on 2020-01-20, and available at https://www.dublincore.org/schemas/rdfs/.

2.2 DCAT

DCAT is an RDF vocabulary designed to facilitate interoperability between data catalogues published on the Web. It enables a publisher to describe datasets and data services in a catalogue using a standard model and vocabulary that facilitates the consumption and aggregation of metadata from multiple catalogues. This can increase the discoverability of datasets and data services. It also makes it possible to have a decentralised approach to publishing data catalogues and makes federated search for datasets across catalogues in multiple sites possible using the same query mechanism and structure. Aggregated DCAT metadata can serve as a manifest file as part of the digital preservation process.

As illustrated in Figure 1, DCAT relies on the FOAF and DCTERMS vocabularies. Note that while the current widely used DCAT version is 2.0, this work covers both the stable version 2.0 and the upcoming version 3.0. Whenever needed the explicit version will be mentioned.

We will refer to the official RDF representation of DCAT version 2 available at https://github.com/w3c/dxwg/blob/gh-pages/dcat/rdf/dcat2.ttl.

[1] https://www.dublincore.org/

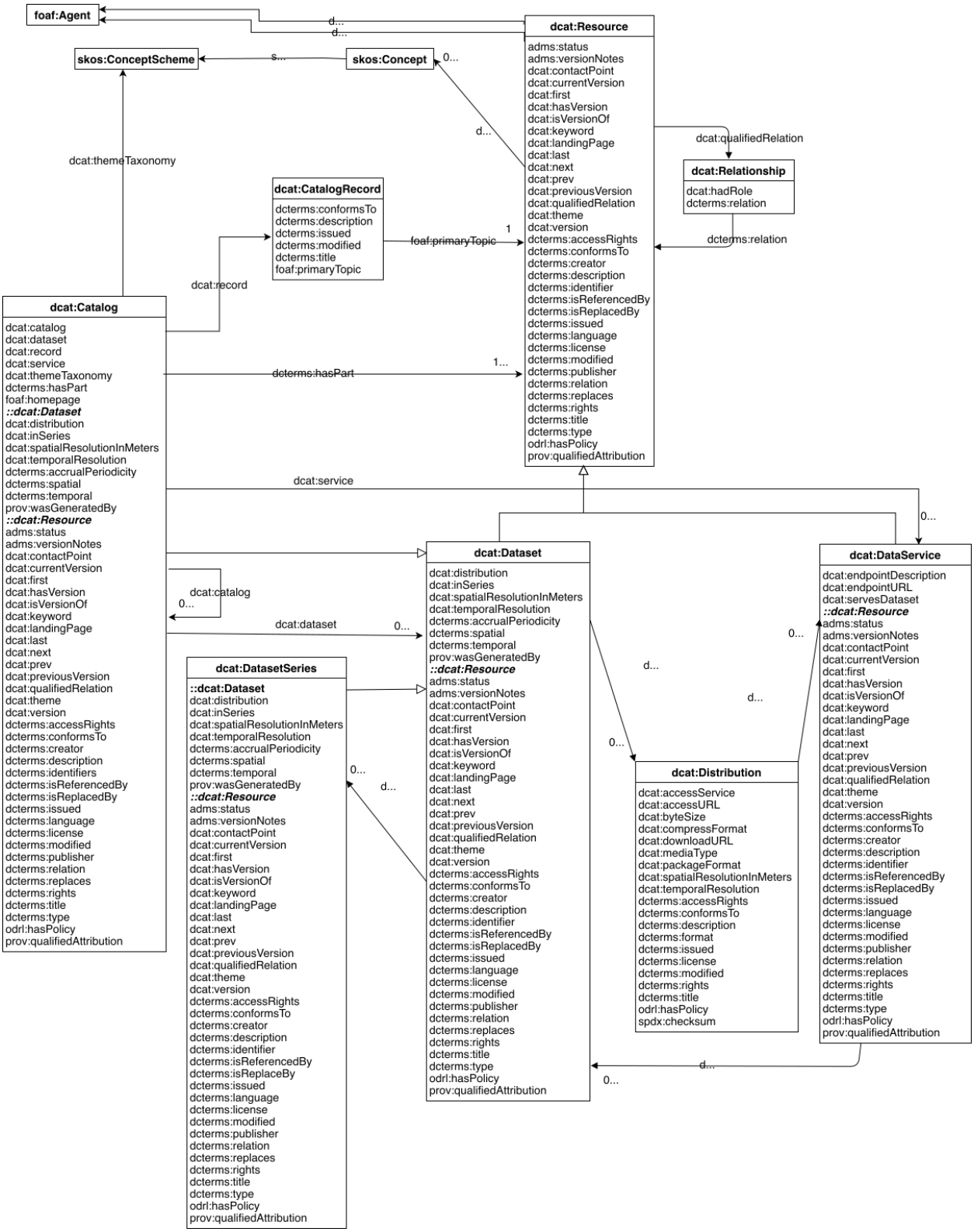

Figure 2 The DCAT (shown is Version 3) schema relies on DCTERMS, FOAF, SKOS, etc. Image from https://www.w3.org/TR/vocab-dcat-3/images/dcat-all-attributes.svg

2.3 PROV-O

The provenance ontology, PROV (PROV-O) expresses the so called PROV Data Model[2] using the OWL2 Web Ontology Language (OWL2). PROV-O aims to provide a set of classes, properties, and restrictions that can be used to represent and interchange provenance information generated in different systems and under different contexts. PROV-O has three different main parts, arranged from the most simple and fundamental terms (and concepts) needed for simple applications of provenance to more complex ones. These are the 1) Starting Point terms, 2) Expanded terms, and 3) terms for Qualifying relationships.

The **Starting Point classes** and properties provide the basis for the rest of the PROV Ontology and are used to create simple provenance descriptions. These include as shown in Figure 3 terms such as wasDerivedFrom, wasGeneratedBy, etc. These provide the minimal provenance elements.

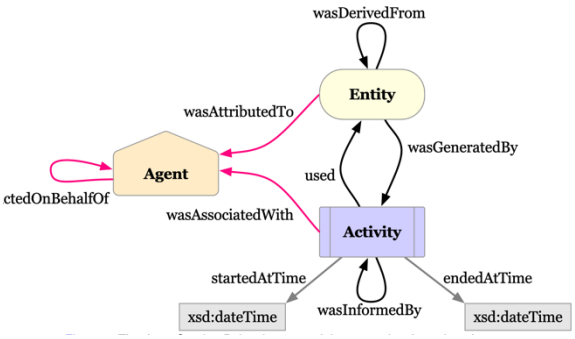

Figure 3 The three Starting Point classes and the properties that relate them. From https://www.w3.org/TR/prov-o/#description-starting-point-terms.

The **Expanded classes** and properties provide additional terms such as the special concepts that generate a dataset e.g., Person, or Organisation while the **Qualified classes** and properties provide elaborated information about binary relations asserted using Starting Point and Expanded properties. These include e.g., Start, End, Usage, of a data set and similar concepts. The entire PROV-O can be consumed by EMMO directly with the elementary mappings proposed here.

[2] http://www.w3.org/TR/2013/REC-prov-dm-20130430/

3. Data Set Basic Concepts

3.1 Concept List

Since DOME 4.0 aims at connecting various data providers with consumers without necessarily providing full access to the actual data, DOME4.0 needs sufficient information about data (i.e., the metadata) to the extent it allows the discovery (or findability) of the data sets based on:

i) general asserted criteria (i.e. most widely used concepts for data set documentation),
ii) the catering for options for accessing the data which are delegated to either external platforms or internal additional specialised apps (i.e. the accessibility),
iii) the ability to interpret the data in these data sets by bespoke tools (i.e., interoperability), and consequently, the ability to reuse the data in various applications which is delegated to other platforms (e.g. Marketplaces, OIP, OTEs, etc)

DOME4.0 facilitates FAIR-ness by providing the minimal essential criteria and components without necessarily a direct access to the entire datasets and without necessarily hosting the data itself. The novelty, or special position/stance of DOME 4.0 is the ability to connect data sets rapidly and semantically in various platforms, including those hosted by individual end users. Hence, DOME 4.0 requires a high-level dataset exchange ontology that covers primarily the top-level information criteria or metadata, creating a specific DOME 4.0 data model.

However, instead of just creating such a high-level data model, such as available already in DCAT or other similar initiatives, the need for deep semantic brokering and exchange in DOME 4.0 necessitates a true, logical, data set ontology rather than a metadata schema (like RDFs), or a set of keywords (like Dublin core). However, to remain compatible with the existing widespread practice that relies on such ad hoc information models, DOME 4.0 starts on the one hand by identifying and selecting its own basic vocabularies and concepts to be as close as possible to such models and extends these with a full ontology model on the other hand. The following criteria are used in the selection of the elementary keywords or vocabularies:

1) Support basic metadata that allow brokering activities mediating and connecting various providers and consumers
2) Stay minimal, only the most needed terms are considered (as more specialised tools will be able to “dig into the data” semantically or otherwise later, i.e., once the datasets have been identified
3) Support elementary keywords that hint and give information as much as possible on the content and nature of the data sets and enable choice of ontology-based keywords (e.g., based on existing terms and labels) that are semantically connected with a meaning.
4) Use or more accurately, reuse the same vocabularies when possible as DCAT (relying on version 2 and when needed on version 3)
5) Be as close to the potential RDFS schema vocabularies incarnations to make explicit reference to what we refer with the concept.

The table below shows the current selection of main terms from DCAT that DOME 4.0 adapts and integrates into a logically fully-fledged ONTOCOMMONS ontology, namely EMMO. Note that this initial data set may be updated later with additional terms as needed.

One of the main advances of such an approach is that a very broad, and loosely defined concept such as a dcat:keyword which can in principle have any arbitrary value (a string in DCAT) may, by means of integrating into a logically strong ontology, be extended into “semantic keywords”, namely it takes only values that themselves are ontological, in other words, a simple DCAT terms can then be enriched semantically to a much deeper expressiveness power enabling reasoning on the actual properties of the dataset itself beyond simply executing specific regex (regular expression) matching on random keywords.

Note however, that while this approach enables higher semantic reasoning, at the same time, there is no direct connection between the values of the keywords and the actual content of the dataset, in other words, a dataset can have a keyword referring for instance to say, mechanical properties of a metal, there is no guarantee that such information or data is indeed found or covered by the dataset. DOME 4.0, and in fact the entire community must rely on basic assumptions that such keywords deliver a trusted set of information about the content of the dataset. In DOME, specific tools for the provenance and reliability of the datasets are envisioned that will address such issues later. The main purpose of the dataset ontology is to allow the brokering on the one hand, and on the other, enable the development of such additional tools to handle and manipulate data sets.

The list of elementary metadata (terms) adopted for the dataset ontology is shown in

Table 1 with the corresponding definitions and RDFS reference. We will use such namespaces abbreviations:

- dcterms: http://purl.org/dc/terms
- dcat: http://w3.org/ns/dcat#
- foaf: http://cmlns/foaf/0.1
- prov: http://www.w3.org/ns/prov#
- xsd: http://www.w3.org/2001/XMLSchema#
- rdf: http://www.w3.org/1999/02/22-rdf-syntax-ns#
- rdfs: http://www.w3.org/2000/01/rdf-schema#
- owl: http://www.w3.org/2002/07/owl#

Table 1 List of DOME4.0 data documentation concepts and their reference to existing RDF schemas.

Label	Definition	RDFS Schema References
DataSet	DCAT: A collection of data, published or curated by a single agent, and available for access or download in one or more representations.	dcat:Dataset (rdfs:Class) subclass of dcat:Resource (rdfs:Class)
Title	DCTERMS/DCAT: A name given to the resource.	dcterms:title (rdf:Property) with range rdfs:literal
Keyword	DCAT: A keyword or tag describing the resource.	dcat:keyword (rdf:Property) with range rdfs:literal
Creator	DCTERMS/DCAT: An entity responsible for making the resource.	dcterms:creator (rdf:Property) with range dcterms:Agent (rdfs:Class)
Publisher	DCTERMS/DCAT: An entity responsible for making the resource available.	dcterms:publisher (rdf:Property) with range dcterms:Agent (rdfs:Class)
Issued	DCTERMS/DCAT: Date of formal issuance of the resource.	dcterms:issued (rdf:Property) with range rdfs:literal
License	DCTERMS/DCAT: A legal document giving official permission to do something with the resource.	dcterms:license (rdf:Property) with range dcterms:LicenseDocument (rdfs:Class)
Source	DCTERMS/DCAT: A related resource from which the described resource is derived.	dcterms:source (rdf:Property)
URI	RDF-XSD: xsd:anyURI represents an Internationalized Resource Identifier Reference (IRI). DCTERMS/DCAT: An unambiguous reference to the resource within a given context.	xsd:anyURI (rdfs:Datatype) dcterms:identifier (rdfs:Datatype)
Homepage	FOAF/DCAT: The homepage property relates something to a homepage about it. (a public Web document usually available in HTML).	foaf:homepage (owl:ObjectProperty) with range foaf:Document (rdfs:Class)
Description	DCTERMS/DCAT: An account of the resource.	dcterms:description (rdf:Property)

4. EMMO Mapping

4.1 Methodological Approach

The semantic enhancement is obtained by creating a mapping between existing DCAT/DCTERMS/FOAF terms addressing the concepts listed in

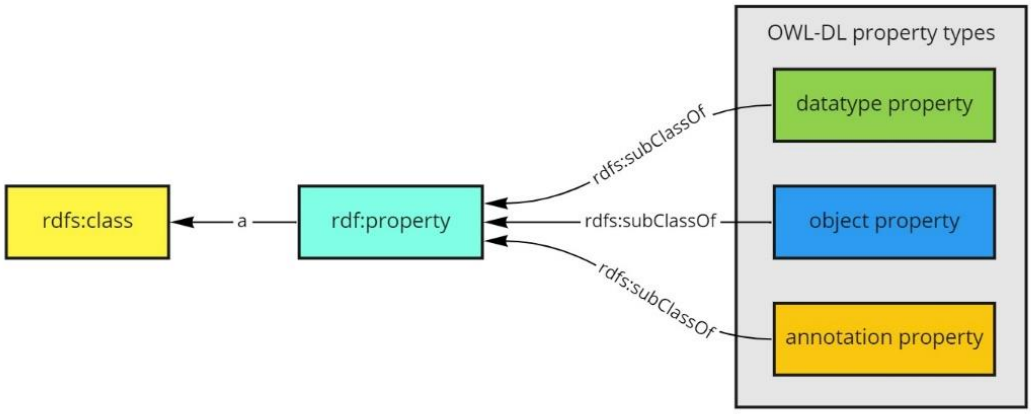

Figure 4 The rdf:Property and their type restriction in OWL 2 DL as data, object or annotation properties

Table 1 and the EMMO top and mid-level concepts, to grant the users the possibility of documenting their data under a wider knowledge framework. The target language for the semantic enhancement is OWL 2 DL, to enable the most powerful expressivity available nowadays by semantic web technologies.

RDF/RDFS schemas implement basic types and relations between entities[3], such as: rdf:type, rdf:Property, rdfs:Class, rdfs:range, rdfs:domain, rdfs:subClassOf, rdfs:subPropertyOf. While these types are useful to build taxonomies, they lack the expressivity to impose constraints that enable the representation of knowledge at a higher detail level.

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, there is the need to specify one specific type of property between datatype, object, or annotation property[4]. The mapping will then distinguish between the different types of properties according to the expected range and domain.

Reasoning in OWL 2 DL is based primarily on the object properties used by axioms to express semantic constraints between ontology entities. Data properties can also be used by axioms to express other data-related constraints. Data-based inferences are supported by most of the existing reasoners. On the contrary, annotations are not used by reasoners, and require *ad hoc* actions to be used semantically (e.g. through SPARQL construct queries digging through annotations and generating triplets according to some user-defined inferencing rules). For this reason, the EMMO concepts mapping the RDF-DEV will focus mostly on OWL 2 DL object and data properties.

[3] See: https://www.w3.org/TR/rdf-schema/

[4] 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.

4.2 EMMO Perspectives

Starting with the original RDF-DEV terms, the EMMO provides several directions for semantic enhancements (enrichments). The most fundamental enhancement is the embedding of the terms within a mereocausality framework, that enables the representation of parthood and causality relations, together with a causal-graph topology that constitute the foundation of space and time relations between objects.

Further semantic enhancements are based on the perspectives that constitute the EMMO Middle Level:

- **Data**: the data perspective of the EMMO defines entities according to the nature of the entity and the decoding criteria to be applied to the variation of the entity physical substrate. It provides means to distinguish between e.g. analog vs discrete data, classical vs quantum data, or formal languages, software code vs applications.
- **Semiotics**: the semiotic perspective enables the representation of the process of defining a meaning for the data, documenting the methodology for data generation, the subjectivity or objectivity of the process, the measurement- or modelling-based generation process.
- **Holistic**: the holistic perspective represents the relations between the whole together with the parts (roles) that makes it something more than a simple mereological sum.
- **Persistence**: the persistence perspective classifies things following the classical object/process dichotomy, and combined with the holistic perspective, provides concepts like participant, component, stage, constitutive process.
- **Physicalistic**: the physicalistic perspective represents things according to their physical form (e.g. matter, field, solid, liquid, crystal).
- **Reductionistic**: the reductionistic perspective provides means to describe composition through granularity levels, that can be used to syntactically describe the data. If combined with the semiotic perspective, it can provide a way to semantically map the data included in a dataset according to a syntactical structure.
- **Perceptional**: the perceptional perspective enables the representation of the data as they appear to the human end user perception (e.g. characters, pictures).

The hierarchy of the EMMO perspective level is shown in Figure 5. We will refer here to the EMMO 1.0.0-beta3 release, available at http://emmo.info/emmo/1.0.0-beta3/middle

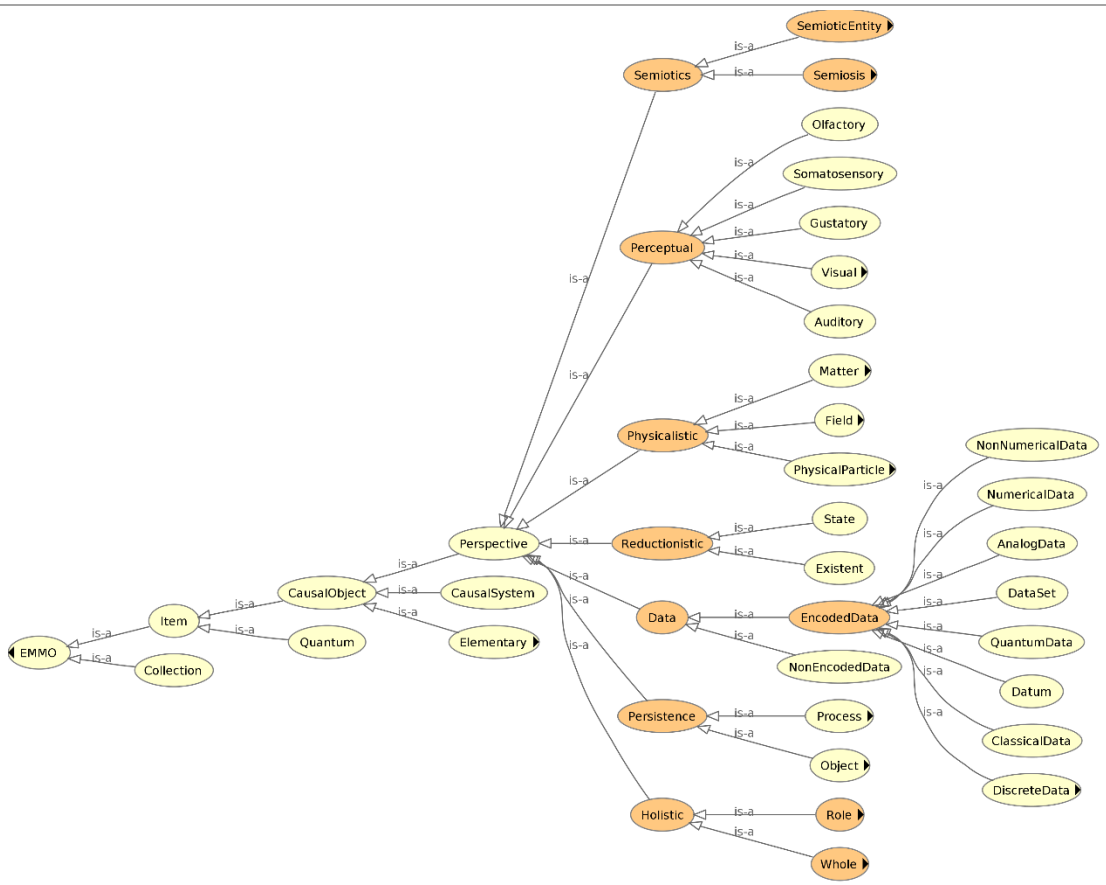

Figure 5 The EMMO class hierarchy up to the first level beyond Perspective class.

An example of semantic enhancement using the EMMO semiotics perspective is shown in Figure 6, where the basic data documentation of a document (i.e. title, author) are expressed mereologically as original part of the document. Moreover, the rating of the document according to an evaluator (the Librarian individual) is attached to the document as non-mereological metadata property (Metadata class) together with the knowledge of the rating schema (the ReadersRate class), and the process of generation of this metadata (the BookEvaluation class).

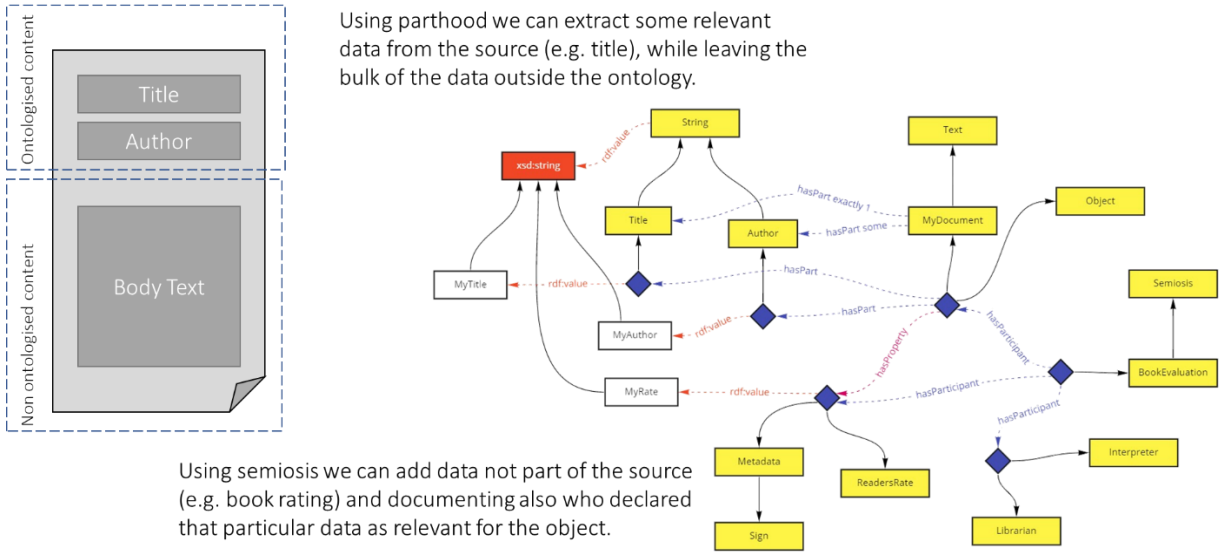

Figure 6 Example of semantic enhancement of basic data documentation.

4.3 Concept Mappings

4.3.1 DataSet

4.3.1.1 Reference Schemas

The **dcat:Dataset** is an **rdfs:Class** defined as: *"A collection of data, published or curated by a single agent, and available for access or download in one or more representations"* [5]. The term is aimed to represent the actual dataset as published in a repository and made accessible by the dataset provider.

Usage notes specify that *"This class describes the conceptual dataset. One or more representations might be available, with differing schematic layouts and formats or serializations"* and that *"This class describes the actual dataset as published by the dataset provider"*. While the first sentence seems to refer to an abstract entity, the second one is referring to actual published data, placing the entity not only in time and space, but also including the publishing process that requires manipulation of the data material basis. These notes makes the ontological status of a dataset not clear, especially when is interpreted according to a rigorous Top Level Ontology framework.

A related term is provided by the **dcat:Distribution** class, defined as *"A specific representation of a dataset. A dataset might be available in multiple serializations that may differ in various ways…"*, that seems to define instances of datasets, discriminating them by e.g. natural languages, media type or format[6]. Moreover, the notes for **dcat:Distribution** term expand its scope also towards the informational equivalency level (e.g. lossy vs. lossless transformation), which is not inline with the concept of instance in ontological sense. How can a distribution partially instantiate an information concept represented by a specific dataset without referring to another dataset?

[5] https://www.w3.org/TR/vocab-dcat-3/#Class:Dataset
[6] https://www.w3.org/TR/vocab-dcat-2/#Class:Distribution

Finally, the notes delegate to application specific choice the definition of what are the requirements for a distribution to still refer to a dataset, introducing a further element of subjectivity.

The relation **dcat:distribution**, connecting a **dcat:Dataset** with a **dcat:Distribution**, is then so ontologically wide that encompasses at least type (e.g. is a distribution an instance of a dataset?) and semiotic (e.g. is the distribution another sign for the same object?) relations. This implies that the semantic extension of such fuzzy concepts within a more rigorous ontological framework would necessarily require a strong restriction of the original concepts.

4.3.1.2 EMMO Mapping

The mapping of **dcat:Dataset** is shown graphically in Figure 7. The crux of the mapping of DCAT into a proper ontology is largely catered for by the realisation that a **dcat:Dataset** is a superclass of **emmo:EncodedData** which is subclass of **emmo:Data**. While **emmo:Data** is a general class that can also describe wild data (non-generated by an agent), the DCAT datasets are more specific.

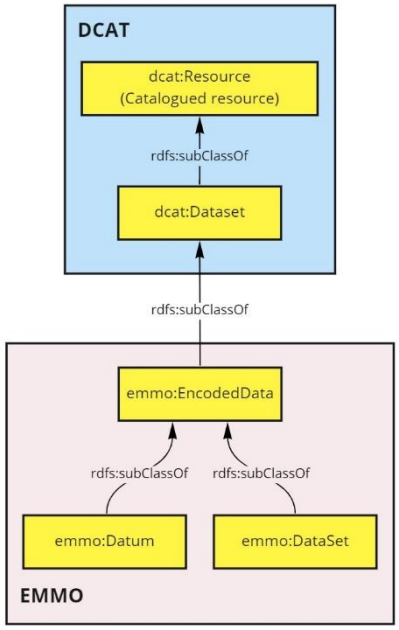

Figure 7 EMMO mapping of dcat:Dataset

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. The conceptual level to which the DCAT definition of a dataset refers to is provided in the EMMO by semiotic relations pointing to the object described by the informational content of the dataset. The EMMO semiotic relations include also to the methodology and the authorities for which such relations hold (see the emmo:hasIsAboutKeyword terms defined in the following sections) significantly improving the semantic description of the dataset scope.

The fact that a dataset individual is expressed following a particular syntactic format (e.g. XML, JSON) or within a material substrate (e.g. paper, CDROM, SSD), has introduced in the definition of the **dcat:Distribution** concept, can be expressed in the EMMO using classes defined under the data perspective that focus on the syntactic data structure or on the physical nature of the substrate. In this sense, **emmo:EncodedData** encompasses also **dcat:Distribution** concept and enables to distinguish

between the possible incarnation of a dataset (e.g. syntactic format, material basis) simply by using a **rdf:Type** relation pointing to specific classes (e.g. class of CSV files, class of CD-ROM entities).

The **dcat:distribution** relation can be mapped within the EMMO in both type and semiotic relations to better specify the connection between a dataset and its possible expressions[7]. This significantly enrich the semantic capabilities available with the original DCAT terms.

4.3.2 Title

4.3.2.1 Reference Schemas

The **dcterms:title** is a **rdf:Property** that is defined as: *"A name given to the resource"*[8]. The term is defined in DCTERMS as a generic property ranging to **rdfs:Literal**, where the actual metadata about the name of the resources is provided. The specification on the range makes it potentially either an OWL 2 DL data or annotation property.

4.3.2.2 EMMO Mapping

The EMMO mapping towards **dcterms:title** is shown in Figure 8. Since the information delivered by the term is an actual data, the EMMO mapping is simply provided by a data property **emmo:hasTitle** , that ranges towards **rdfs:Literal** from a **emmo:Data** domain.

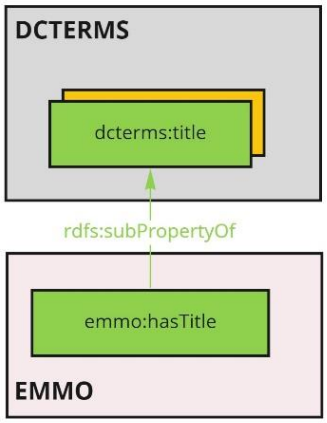

Figure 8
EMMO mapping of dcterms:title

4.3.3 Keyword

4.3.3.1 Reference Schemas

The **dcat:keyword** is defined as **rdf:Property** with range **rdfs:Literal**, and in the Turtle serialization[9] is specified as an **owl:DatatypeProperty**. The DCAT definition of this term is: *"A keyword or tag describing a resource"*[10].

This term allows to store as literal types a set of free-for-all textual data to enhance the semantic description of the dataset and is the only available DCAT approach to semantic enhancement of dataset.

4.3.3.2 EMMO Mapping

Figure 9 demonstrates how the **dcat:keyword** term can be expanded to provide semantical enhancement in the data set description. The **dcat:keyword** is a list of arbitrary strings used to designate or reflect some information for a human agent about the content of a dataset is semantically enriched to reflect in a more machine interpretable form the same information. The first step is to define **emmo:hasKeyword** as a sub

[7] The **dcat:distribution** relation has not been mapped here, due to its interpretation within the EMMO that crosses ABox and TBox.
[8] http://purl.org/dc/terms/title
[9] https://www.w3.org/ns/dcat2.ttl
[10] https://www.w3.org/TR/vocab-dcat-3/#Property:resource_keyword

property of **dcat:keyword**, with domain **emmo:Data**[11]. Then by splitting the latter into syntactic and semantic relation types enables to both support datasets described according to the current non semantic standards through the **emmo:hasSyntaticKeyword** data property and the new **emmo:hasSemanticKeyword** data relation, that ranges towards an IRI of an OWL 2 entity. The latter is particularly powerful thanks to three sub property types the connect the dataset entity with other ontological entities, given that the range of this data property is the IRI of a subclass of **emmo:EMMO**, i.e. the most generic ontological class. 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. More than one type can be defined for the data. 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** and a range **emmo:Property**.

[11] The DCAT 1 domain of dcat:keyword was dcat:Dataset. But this has been relaxed in DCAT 2 by removing it. In this mapping we consider more in line with the EMMO interpretation to reintroduce the emmo:Data domain.

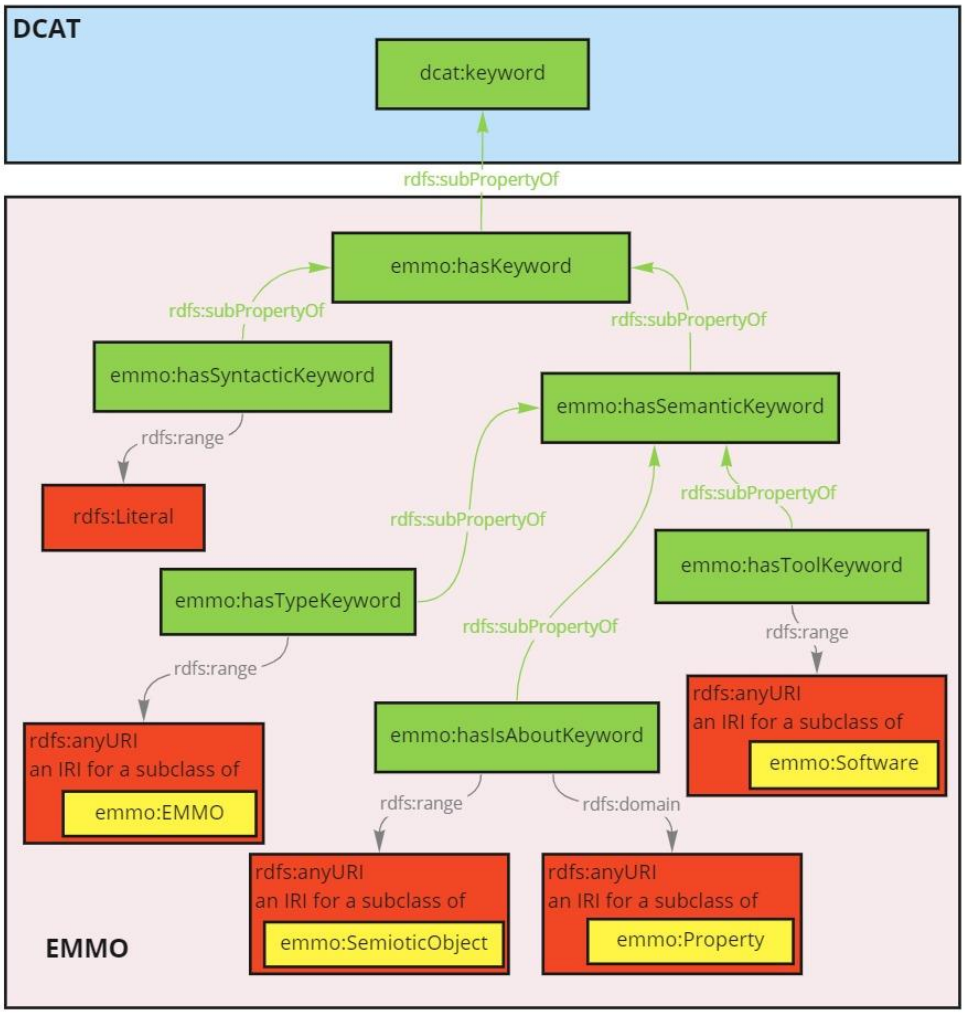

Figure 9 EMMO mapping of dcat:keyword

In this case of course, additional tools need to be developed to assert such a relation as described above within an RDFS framework. With the lack of deep semantic expressiveness (i.e. without forcing each and every dataset in the entire domain of discourse to be described by an ontology down to the individual bits) such tools are inevitable. The semantic enrichment of the keywords enables automated assertion tools, which is an advancement to the state of the art where no such assertion can be usually made via a machine. Thus, this approach taken here opens the route for applications of AI tools which DOME is planning in the near future.

For example, the SPARQL query:

```
SELECT ?x ?y
WHERE {?x emmo:hasTypeKeyword ?y}
```

can be used to resolve the user defined type data properties into strong axioms that will place the data within a specific semantic position within the ontology, by creating `rdfs:subClassOf` triplets from the query results. Similarly, the query:

```
SELECT ?x ?y
WHERE {?x emmo:hasIsAboutKeyword ?y}
```

may be used to express that a data is about another ontologically represented object, by creating rdfs:`hasProperty` triplets from the query results.

The last bit of semantic enrichment is the eminent **emmo:hasToolKeyword** which 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). This term provides a list of keywords 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 (or in fact, also any semantic formats defined according to any other standard). Future work will add keywords that directly link such tools to the dataset (which will be part of the provenance ontology being developed currently in DOME 4.0).

4.3.4 Creator

4.3.4.1 Reference Schemas

The term **dcterms:creator** is an **rdf:Property** whose range includes the class **dcterms:Agent**. The definition is: *"An entity responsible for making the resource"*[12]. DCTERMS specifies that is an OWL 2 equivalent property with respect to **foaf:maker**[13], making it an OWL 2 object property. The domain is any possible ontologically represented entity (**owl:Thing**).

4.3.4.2 EMMO 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.

[12] https://www.dublincore.org/specifications/dublin-core/dcmi-terms/#http://purl.org/dc/terms/creator
[13] http://xmlns.com/foaf/spec/#term_maker

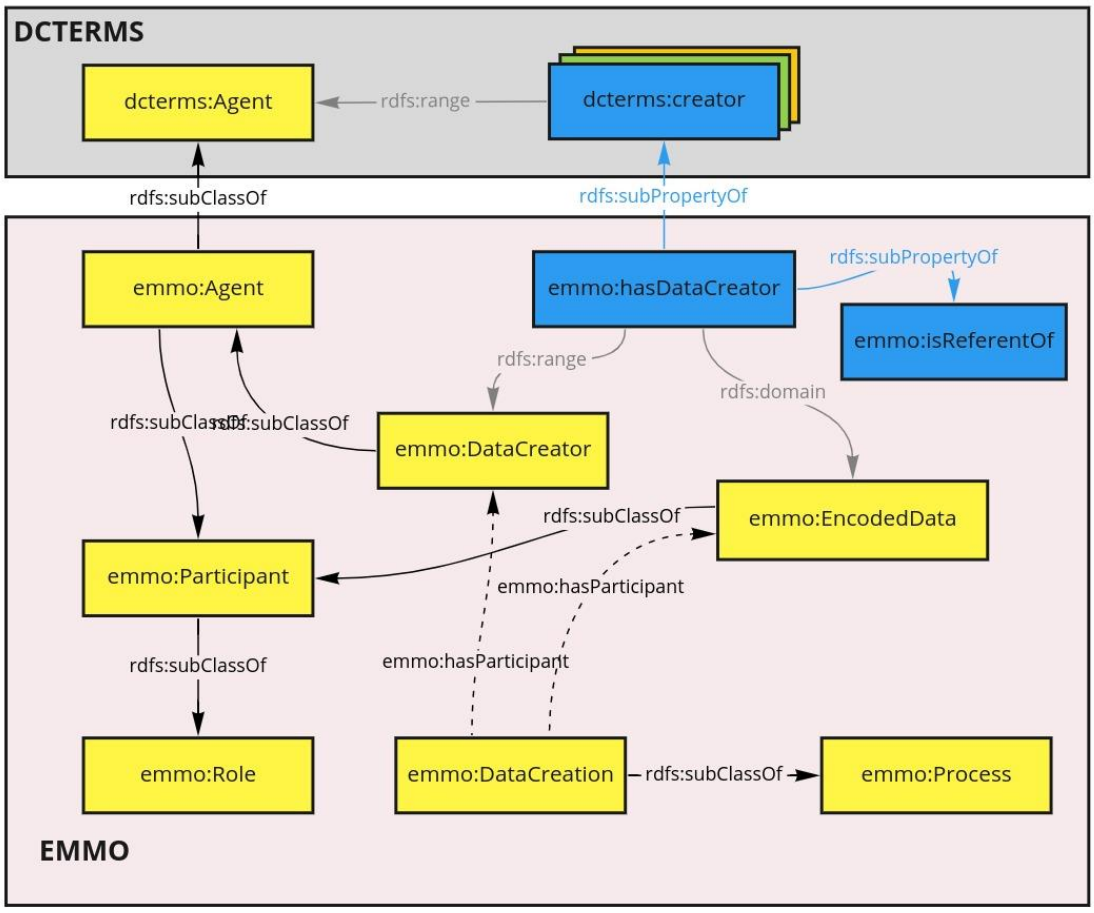

Figure 10 EMMO mapping of dcterms:Agent and dcterms:creator

4.3.5 Publisher

4.3.5.1 Reference Schemas

The term **dcterms:publisher** is an **rdf:Property** whose range is the class **dcterms:Agent**. The definition is: *"An entity responsible for making the resource available"*[14].

4.3.5.2 EMMO Mapping

The EMMO mapping provides a structure like the DCTERMS creator term mapping.

4.3.6

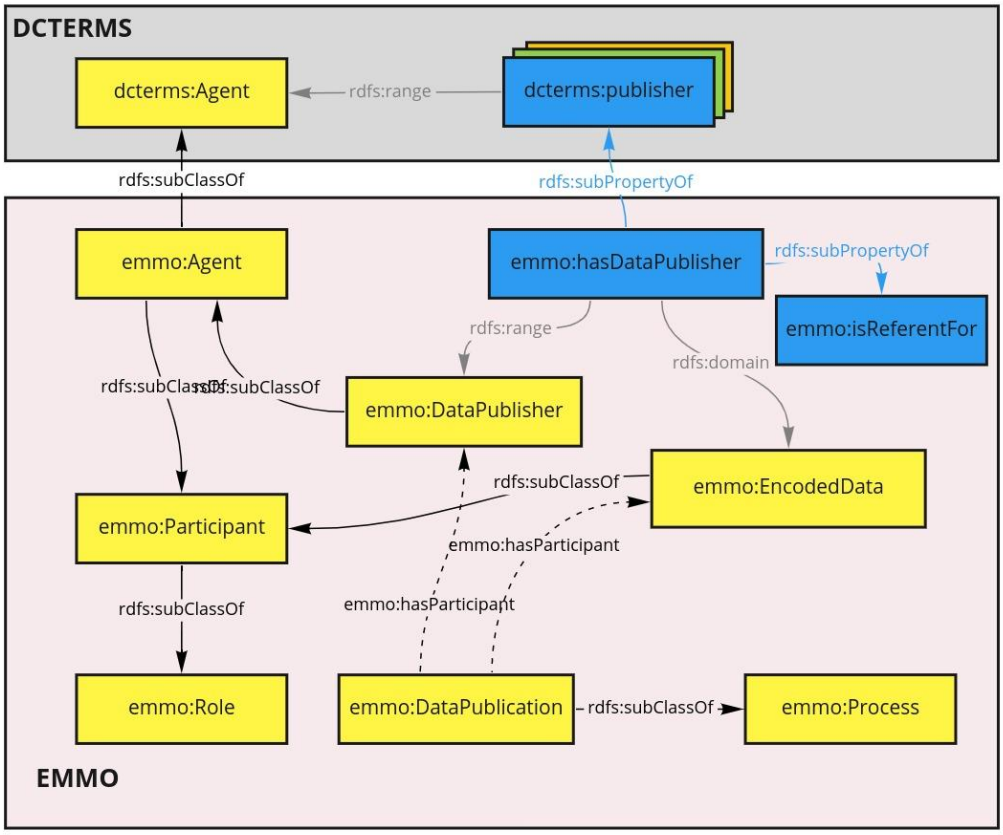

Figure 11 EMMO mapping of dcterms:Agent and dcterms:publisher

[14] https://www.dublincore.org/specifications/dublin-core/dcmi-terms/#http://purl.org/dc/terms/source

4.3.7 Issued

4.3.7.1 Reference Schemas

The **dcterms:issued** is a **rdf:Property** that is defined as: *"Date of formal issuance of the resource"*[15]. The term is defined in DCTERMS as a generic property ranging to **rdfs:Literal**, where the actual metadata about the issue date of the resources is provided. The specification on the range makes it potentially either an OWL 2 DL data or annotation property.

4.3.7.2 EMMO Mapping

The EMMO mapping towards **dcterms:issued** is shown in Figure 12. Since the information delivered by the term is an actual data, the EMMO mapping is simply provided by a data property **emmo:hasIssueDate** , that ranges towards **rdfs:Literal** from a **emmo:Data** domain.

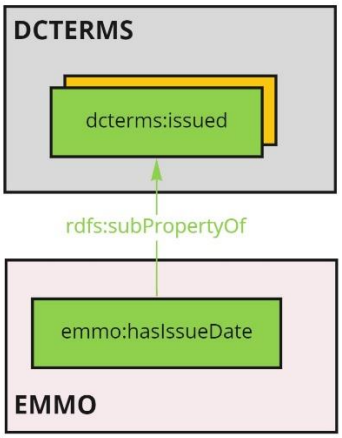

Figure 12
EMMO mapping of dcterms:issued

4.3.8 License

4.3.8.1 Reference Schemas

The **dcterms:license** is a **rdf:Property** that is defined as: *"A legal document giving official permission to do something with the resource"*[16]. The term is defined in DCTERMS as a generic property with a range that includes the class **dcterms:LicenseDocument** defined as: *"A legal document giving official permission to do something with a resource"* [17]. The specification on the range makes it potentially either an OWL 2 DL data, object, or annotation property. The superclass of **dcterms:LicenseDocument** is the class **dcterms:RightsStatement**, referring to *"A statement about the intellectual property rights (IPR) held in or over a resource, a legal document giving official permission to do something with a resource, or a statement about access rights"*[18].

The comment on the DCTERMS license term recommends identifying the license document with a URI, or with a literal value that identifies the license. In the first case the term can be an OWL 2 object property (if referred to an entity IRI), while in the second case can be an OWL 2 data or annotation property.

4.3.8.2 EMMO Mapping

The EMMO mapping enrich the **dcterms:license** providing the object sub property **emmo:hasLicense** referring to the a license document (e.g. the full specification of the GPL3) and the **emmo:hasRights** that refers to any statement claiming rights about a resource (e.g. a license document but also a generic sentence such as "This document is released under GPL3"). Using this approach, it is possible to reproduce

[15] http://purl.org/dc/terms/issued
[16] http://purl.org/dc/terms/license
[17] http://purl.org/dc/terms/LicenseDocument
[18] http://purl.org/dc/terms/RightsStatement

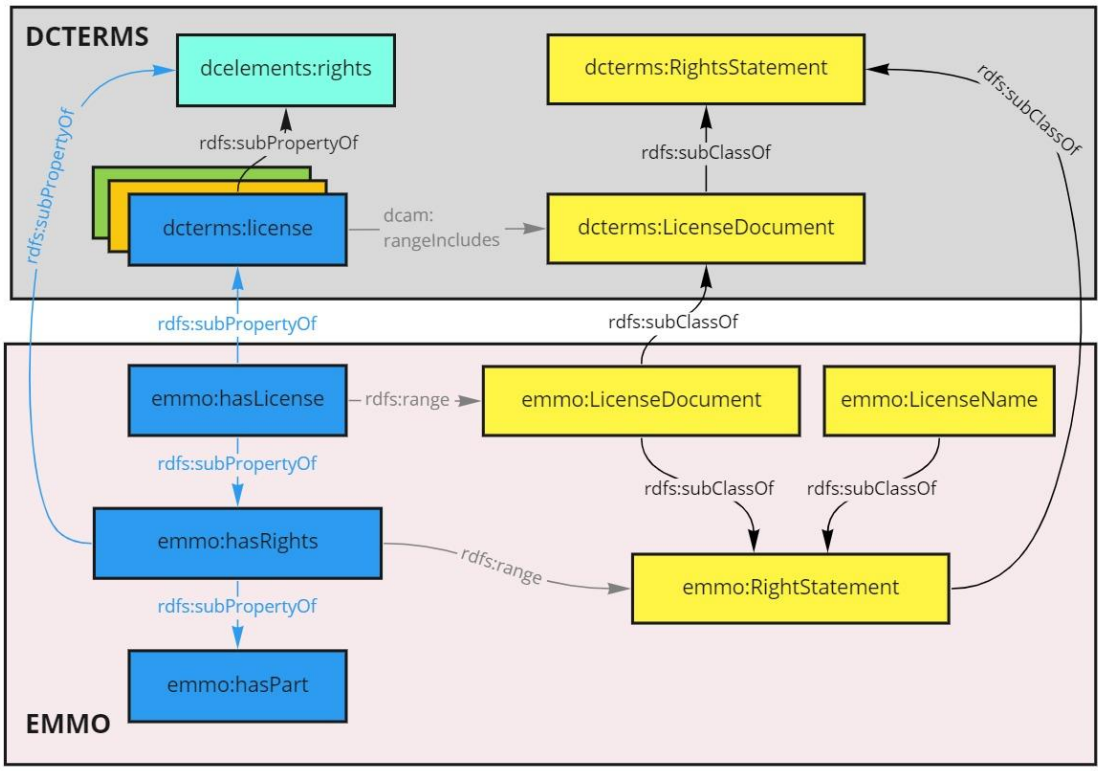

Figure 13 EMMO mapping of dcelements:rights and dcterms:license

the DCTERMS and DCELEMENTS properties that express rights statements and licensing under a mereological framework, able to syntactically place them within the data.

4.3.9 Source

4.3.9.1 Reference Schemas

The **dcterms:source** is a **rdf:Property** that is defined as: *"A related resource from which the described resource is derived."*[19]. In particular, the described resource may be derived from the related resource in whole or in part. It is a sub property of **dcterms:relation** that identifies a generically related resource. DCTERMS recommends using non-literal values, referring to physical, digital, or conceptual entity.

4.3.9.2 EMMO Mapping

The EMMO mapping enhances the expressivity by explicitly considering the two cases of use mentioned in the DCTERMS (but not implemented), when the documented data i) is part of the source, or when ii) is an elaboration of the information given by source. In the first case the **emmo:isDataSubSetOf** relates mereologically the data to the whole dataset to which it belongs. In the second case, the **emmo:isDerivedDataOf** relates the data to the whole dataset to which is derived, following a semiotic process that documents also the agent and the methodology used for the derivation.

[19] http://purl.org/dc/terms/source

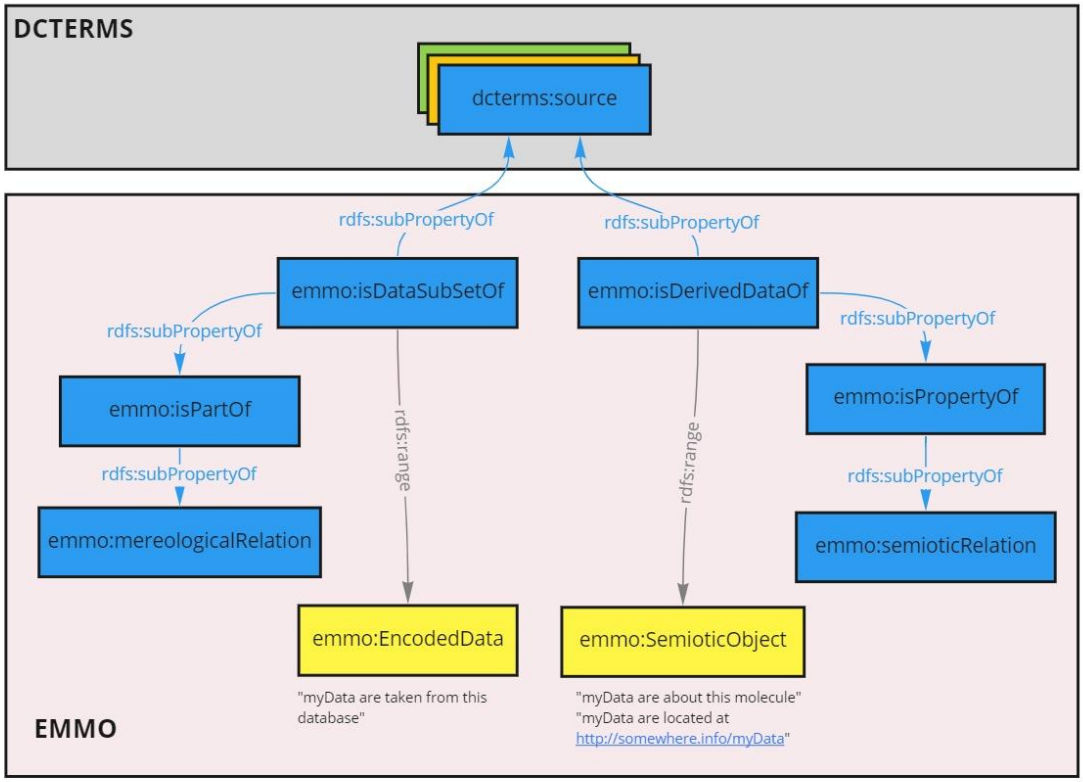

Figure 14 EMMO mapping of dcterms:source

4.3.10 URI

4.3.10.1 Reference Schemas

The need for a unique identifier provided by a URI datatype can be addressed **xsd:anyURI** which is an **rdfs:Datatype** defined as: *"Absolute or relative URIs and IRIs"*[20]. These definitions refer both to http://www.ietf.org/rfc/rfc3986.txt. DCTERMS provides the **dcterms:identifier** which is an **rdf:Property** with range **rdfs:Literal** defined as: *"An unambiguous reference to the resource within a given context."*[21]

4.3.10.2 EMMO Mapping

The **dcterms:identifier** can be used as superclass for the **emmo:hasURI** datatype property that has range **xsd:anyURI** and provides a unique identifier for ant resources. The sub properties **emmo:hasURN** can be used to refer to a specific name according to a particular namespace starting with *urn:* (e.g. urn:uuid:00d47850-ded2-44d8-9b3d-5719d46aeb02). The sub property **emmo:hasURL** can be used to specify a location on a network.

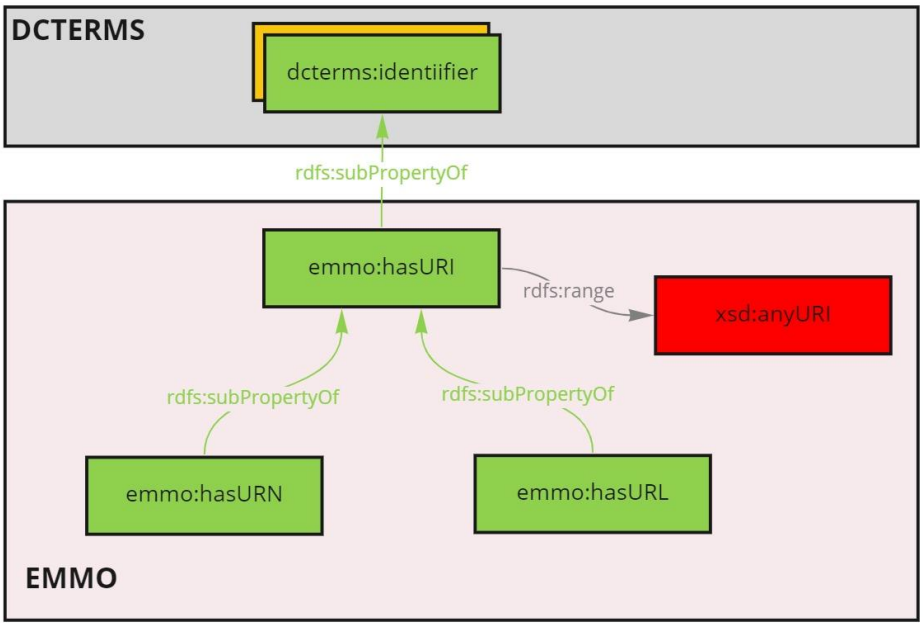

Figure 15 EMMO mapping of dcterms:identifier

[20] http://www.w3.org/2001/XMLSchema#anyURI
[21] http://purl.org/dc/terms/identifier

4.3.11 Homepage

4.3.11.1 Reference Schemas

The **foaf:homepage** is an **owl:ObjectProperty** defined as: *"The homepage property relates something to a homepage about it"*[22]. The domain is **owl:Thing** and range **foaf:Document**, both **rdfs:Class**. The **foaf:Document** concept is defined as: *"The Document class represents those things which are, broadly conceived, 'documents'"*, without distinguishing between electronic, physical, copies or abstraction[23].

4.3.11.2 EMMO Mapping

The **foaf:homepage** is mapped to the EMMO through the **emmo:hasHomepage** object property that semantically enhance the concept through the semiotic perspective. This enables to document the process of assign structured data (i.e. the **foaf:Document**) to physical entities. The definition of the document term provided by FOAF is tautological (i.e. a document is a "document") but seems to encompass the overall range of agent-generated data, so that we decided to restrict it with the **emmo:EncodedData** class, as range of **emmo:hasHomepage** to provide a better defined concept.

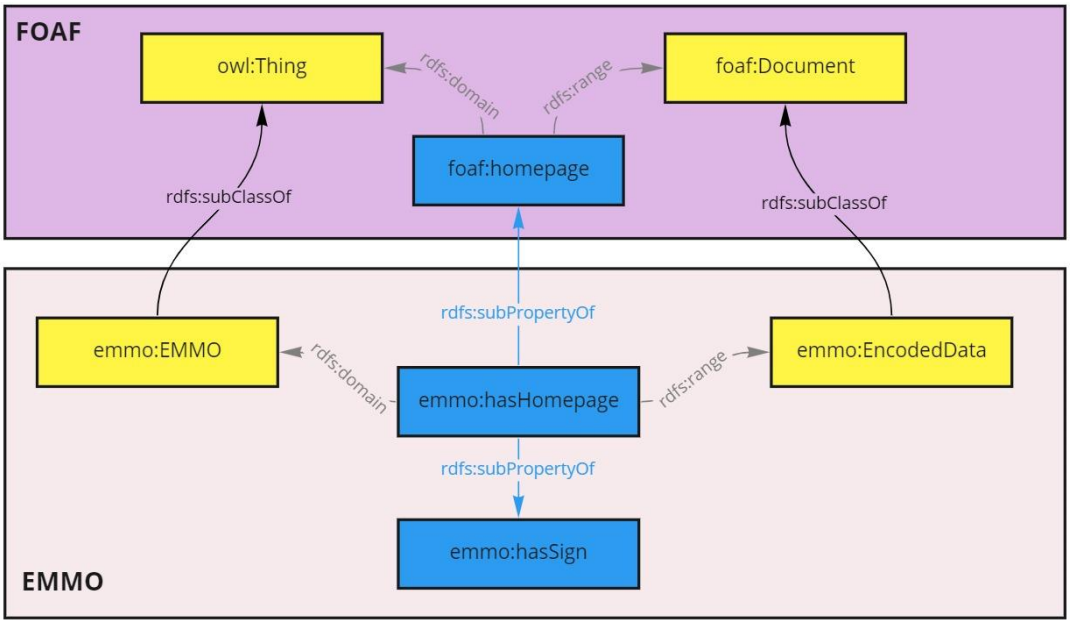

Figure 16 EMMO mapping of foaf:homepage and foaf:Document

[22] http://xmlns.com/foaf/spec/#term_homepage
[23] http://xmlns.com/foaf/spec/#term_Document

4.3.12 Description

4.3.12.1 Reference Schemas

The **dcterms:description** is a **rdf:Property** with range **rdfs:Literal**, defined as *"An account of the resource"*[24], while in DCAT it is defined as: "*A free-text account of the item*"[25]. A description may include but is not limited to an abstract, a table of contents, a graphical representation, or a free-text account of the resource.

4.3.12.2 EMMO Mapping

Since the terms refers to free text, it is reasonable to assign the status of annotation property when such term is brought into an OWL 2 DL environment. The EMMO possesses several annotations that deals with human-oriented descriptions such as:

- **emmo:definition**, for statements expressed formally within a logical system
- **emmo:elucidation**, for explanations to connect the terms to their real-world counterpart
- **emmo:comment**, for generic considerations about the concept
- **emmo:example**, to show example of usage of the term
- **emmo:etymology**, to provide an etymological analysis of a label aimed to better identify the concept behind a word and its historical evolution.

All these EMMO annotations (that are also **rdfs:comment** sub properties) can be considered legitimate sub properties of **dcterms:description**.

[24] http://purl.org/dc/terms/description
[25] https://www.w3.org/TR/vocab-dcat-3/#Property:resource_description

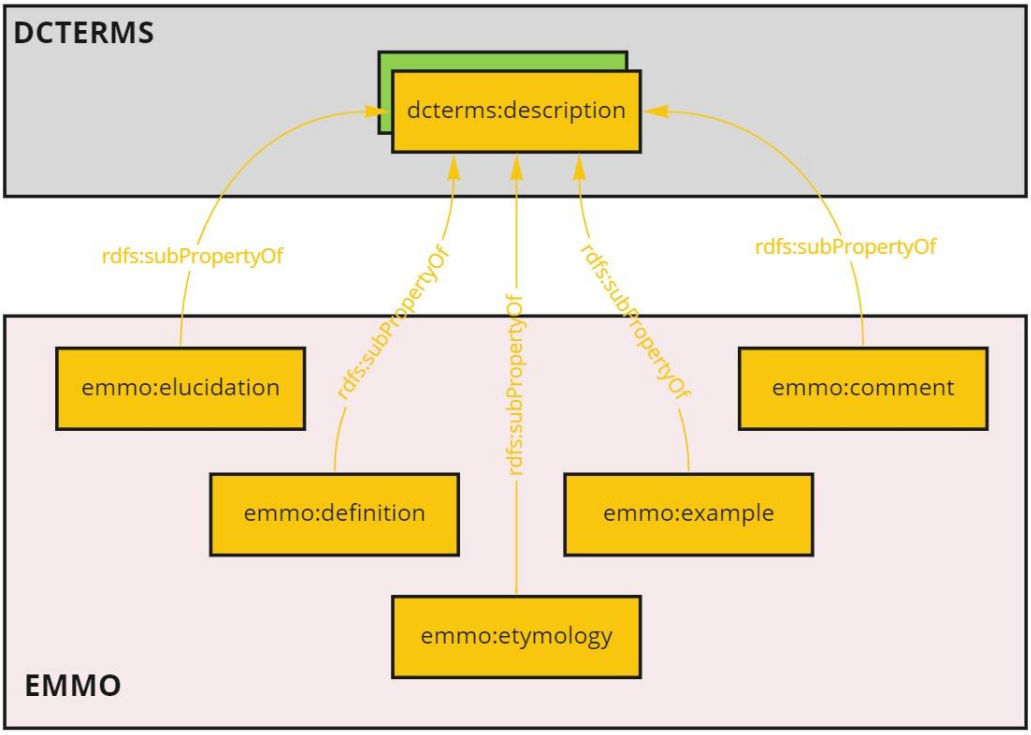

Figure 17 EMMO mapping of dcterms:description

5. Future Developments

The work presented here enables the consumption, or conversion of a general dataset catalogue in DCAT into EMMO. Other standards exist, notably the International Data Space Information Model (IDS-IM)[26,27] which has additional more specific metadata elements pertaining especially to the various forms of data sharing, clearing house and contracts (see Figure 18). The IDS information model fits well with the native DOME information model and will be integrated and mapped in the same manner as DCAT.

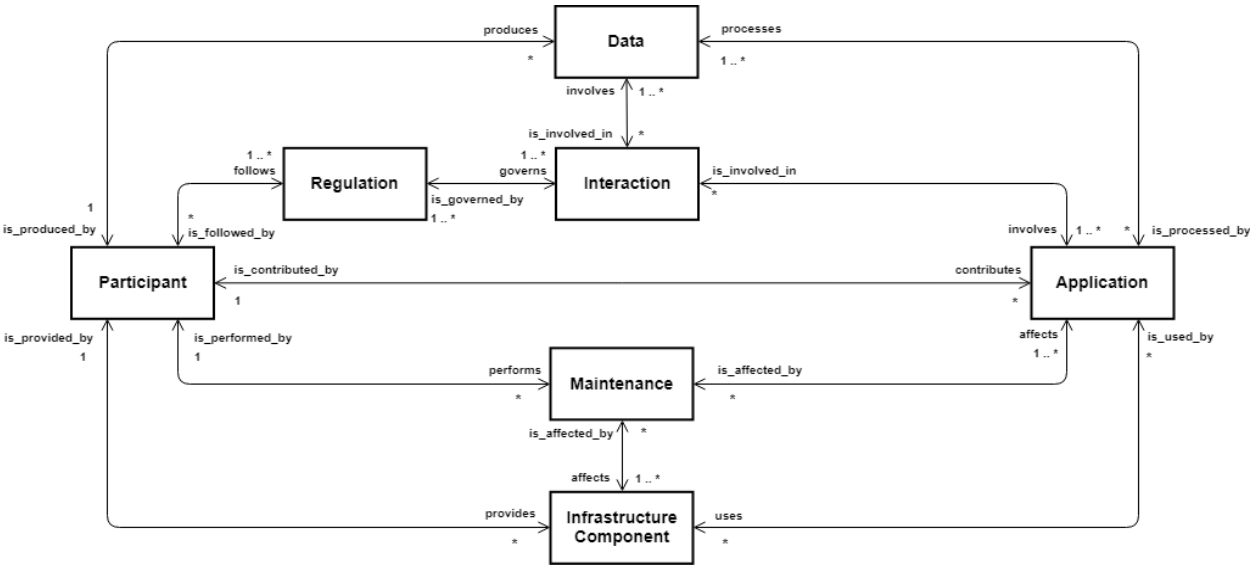

Figure 18 An example of the top level information model of the IDS showing more intricate relations between erntities than e.g., DCAT. From[28]

Once such an integration is achieved, EMMO and DOME 4.0 will enable a new generation of tools and algorithms that can efficiently cater for reasoning-powered automated, seamless, mapping between any IDS resources and any DCAT resource (see Figure 19). In this sense the DOME 4.0 and consequently EMMO mapping enables true cross domain interoperability.

[26] https://github.com/International-Data-Spaces-Association/InformationModel

[27] https://www.fraunhofer.de/content/dam/zv/en/fields-of-research/industrial-data-space/IDS-Reference-Architecture-Model.pdf

[28] https://international-data-spaces-association.github.io/InformationModel/docs/index.html#

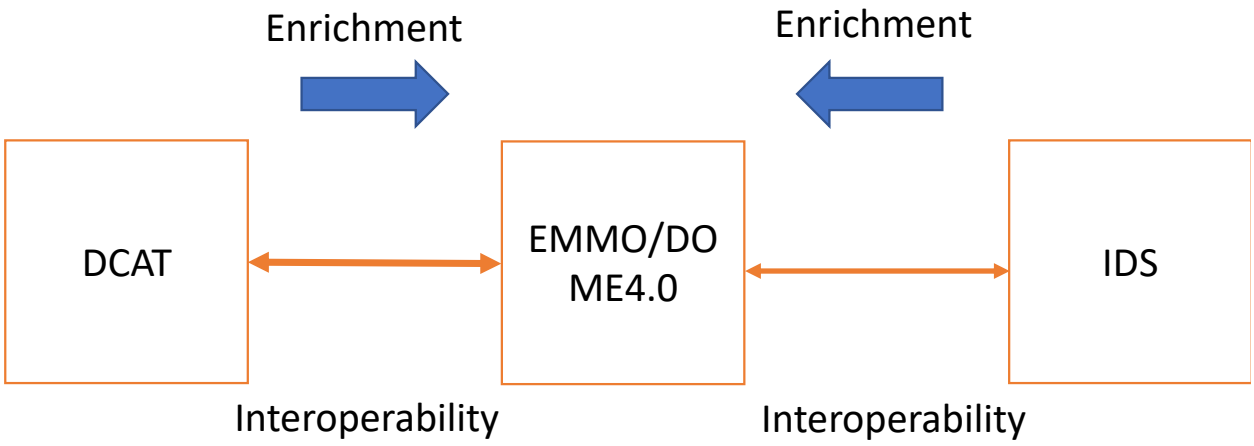

Figure 19 EMMO/DOME 4.0 enables seamless mapping between third party standards. The semantic enrichment enables seamless semantic interoperable exchange.

6. Syntactic Description

The EMMO combined use of perspectives can provide a full **syntactic description of data sets** that can be used to build a **semantic mapping of data sets entries** following standard (e.g. XML, JSON) or custom data formats. This approach is based on meretopology and has been already tested in CIF crystallography EMMO module[29].

An example of such syntactic/semantic mapping is shown in Figure 20, where two ASCII files can be decomposed into meretopological substructures. An ASCII CSV file can be decomposed e.g. in sequence of columns, where a column is a sequence of rows, the first one a header and the others actual data to be semantically interpreted according to an EMMO type.[30]

Such approach will be implemented for relevant data formats and applied to specific semantic mapping in the future DOME4.0 WP3 activities, according to the foundations defined in this document.

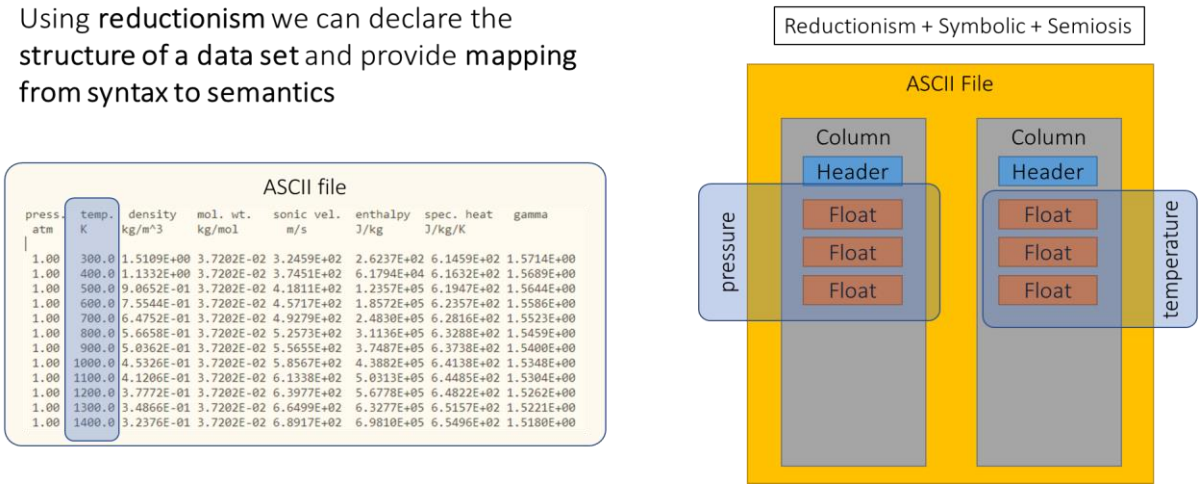

Figure 20 Syntactic description and semantic mapping of data sets.

[29] https://github.com/emmo-repo/CIF-ontology

[30] The syntactic representation of formats can be done referring to standards, such as CSVW https://www.w3.org/ns/csvw for CSV.

7. Networking Actions

The development of the EMMO ontology has been orchestrated by UNIBO/SINTEF/UCL involving several H2020 projects, to create a common ontological framework that would ensure compatibility between semantically documented resources.

The list of the H2020 projects whose resources that the EMMO will make compatible with DOME4.0 are:

- H2020-DT-NMBP-09-2018 SimDOME, *Digital Ontology-based Modelling Environment for Simulation of materials* (4 years, 4.6M€), Grant agreement ID: 814492, An industry-ready modelling framework for materials simulation
- H2020-NMBP-TO-IND-2019 OntoTrans, *Ontology driven Open Translation Environment* (4 years, 5.5M€), Grant agreement ID: 862136, Ontology-based system for more competitive manufacturing processes
- H2020-NMBP-TO-IND-2020 OntoCommons, *Ontology-driven data documentation for Industry Commons* (3 years, 4.2M€), Grant agreement ID: 958371, Standardising data documentation through ontologies
- H2020-NMBP-TO-IND-2020 OpenModel, *Integrated Open Access Materials Modelling Innovation Platform for Europe* (4 years, 5.2M€), Grant agreement ID: 953167, Engineering the future of materials modelling

In particular, UNIBO will extend the proposed RDF-DEV mapping to the OntoCommons TRO, aiming for the reuse of such approach with other top level ontology approaches (e.g. BFO, DOLCE).

8. Acknowledgement

The author(s) would like to thank the partners in the project for their valuable comments on previous drafts and for performing the review.

Project partners:

#	Type	Partner	Partner full name
1	SME	CMCL	Computational Modelling Cambridge Limited
2	Research	FHG	Fraunhofer Gesellschaft zur Förderung der Angewandten Forschung E.V.
3	Research	INTRA	Intrasoft International SA
4	University	UNIBO	Alma Mater Studiorum – Universita di Bologna
5	University	EPFL	Ecole Polytechnique Federale de Lausanne
6	Research	UKRI	United Kingdom Research and Innovation
7	Large Industry	SISW	Siemens Industry Software NV
8	Large Industry	BOSCH	Robert Bosch GmbH
9	SME	UNR	Uniresearch B.V.
10	Research	SINTEF	SINTEF AS
11	SME	CNT	Cambridge Nanomaterials Technology LTD
12	University	UCL	University College London

This document is part of a project that has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 953163. It is the property of the DOME 4.0 consortium and do not necessarily reflect the views of the European Commission.