FMA and HMTP Portal in OWL: Reconciling Ontology with Terminology in Life Sciences via Metamodeling

Christine Golbreich, Julien Grosjean, Stéfan J. Darmoni

FMA & HMTP Portal in OWL: Reconciling Ontology with Terminology in Life Sciences via Metamodeling

C. Golbreich[1], J. Grosjean[2], S.J. Darmoni[2]

[1] LIRMM CNRS 5506, Montpellier & University Versailles Saint-Quentin, France
[2] CISMEF & TIPS LITIS EA 4108, Rouen University Hospital, Rouen, France

cgolbrei@gmail.com, Julien.Grosjean@chu-rouen.fr, Stefan.Darmoni@chu-rouen.fr

Abstract. The aim is to highlight how it is possible to reconcile ontologies and terminologies in Life Sciences via metamodeling. The paper focuses on the Foundational Model of Anatomy (FMA). The first part describes the FMA ontology formalization in OWL 2. The second part presents the Health Multi-Terminologies Portal (HMTP) of French terminologies implemented in OWL. It explains how its FMA terminology was obtained from the FMA ontology by *reification*. The FMA ontology and terminology illustrate how ontologies and terminologies can be made compatible via metamodeling. Advantages and possible means to bind the two views even more closely are discussed in conclusion. To the best of our knowledge, no complete representation of the FMA ontology and terminology in OWL 2 existed so far.

Keywords: Ontology, Terminology, Controlled Vocabulary, Metamodeling, OWL, Portal, Browsing, Indexing; Life Sciences, Health, Anatomy.

1 Introduction

'Ontologies' and 'Terminologies' both refer to vocabularies, but their concern and purposes are different. The former deal with *knowledge* describing the *entities* (concepts) of a domain of interest, the later do not describe the domain but provides information about the *terms* (words) used in the domain (though sometimes the distinction might be blurred or the two notions may overlap, e.g. SNOMED-CT [13]).

"Ontologies are *formalized* vocabularies of terms, often covering a specific domain and shared by a community of users" [1]. They specify the *meanings* of terms by describing their relationships with other terms in the ontology. For example, the Foundational Model of Anatomy (FMA), "a reference ontology about human anatomy" [2] specifies the anatomical structures by their relationships with other FMA entities, indicating their regions, constituents, innervations, blood vessels, boundaries etc., e.g.; a Heart is composed of two regions - its left and right side -, has several constitutional parts – *Wall of Heart, Interatrial, Interventricular*, and *Atrioventricular septum, Mitral Valve*, etc. -, is innerved by the *Deep cardiac plexus, Right and Left coronary nerve plexus*, etc.

The number of biomedical ontologies is increasing, ranging from anatomy, genomics, experimental conditions, imaging, chemistry etc., e.g., Gene Ontology[1], SNOMED-CT.

While a formal ontology provides the meanings of terms as axioms and facts, a terminology is of completely different nature. Terminologies do not formalize the meanings of terms, but state what are the language terms used in a domain of interest and tell information (data) about them, such as the preferred term, synonyms, broader, narrower terms. They consist at minimum, of a controlled vocabulary and a system of identifiers[2]. Identifiers are the primary means of referring to the vocabulary units, e.g. CUI (Concept Unique Identifiers) in the UMLS [14], ID in SNOMED CT. They are associated with a 'label' in natural language intended for communication with humans. For example in MeSH[3], the descriptor (MeSH Heading) identified by the Unique ID D009203, has `Myocardial Infarction` as label,: `Myocardial Infarct` as synonym, is associated to other terms (See Also): `Heart Rupture`, `Post-Infarction`, and can be affiliated with various subheadings (MeSH Qualifiers), e.g. `diet therapy`, `diagnosis` etc. Terms are usually organized into hierarchies. But their hierarchical links, e.g. broader, narrower, do not have any formal semantics and should not be confused with a subclass relationship. They may for instance denote whole-part relations. These links are intensively used to search and browse terminologies.

The UMLS (Unified Medical Language System) includes more than 130 health terminologies, some in multiple languages, e.g., SNOMED Clinical Terms (SNOMED CT), MeSH (Medical Subject Headings), ICD 10 (International Statistical Classification of Diseases and Related Health Problems)[4], HL7 (Health Level Seven Vocabulary), HUGO (Gene Nomenclature Database), WHOART (WHO Adverse Drug Reaction Terminology), etc.

OWL ontologies and terminologies have different advantages. While ontologies provide knowledge useful to support ontology reasoning, design and maintenance, querying or mining data, semantic annotation, health terminologies are extensively used for resources indexing and retrieval, and for coding systems. For example, the classical utilization of SNOMED-CT is to use its catalogue and codes to index medical records. In short, ontologies and terminologies have different nature and purposes: an ontology is a domain model that brings a clear semantics and reasoning support, whereas a terminology is a metamodel of a domain ontology; and supports powerful indexing and search.

The goal is to reconcile these two complementary views and to combine their advantages. The paper presents a first step achieved in that direction for the Foundational Model of Anatomy (FMA). Section 2 first describes the FMA ontology formalization in OWL 2 and the FMA-OWLizer tool developed for it. Section 3 then presents the Health Multi-Terminologies Portal (HMTP) of French terminologies implemented in OWL and its FMA terminology obtained from the FMA ontology. All along, FMA examples illustrate how metamodeling allows reconciling terminologies

[1] http://www.geneontology.org/

[2] http://ec.europa.eu/information_society/activities/health/docs/publications/2009/2009semantic-health-report.pdf

[3] http://www.nlm.nih.gov/mesh/

[4] http://apps.who.int/classifications/apps/icd/icd10online/

and ontologies. The FMA terminology is a metamodel of the FMA ontology, which is represented by an OWL ontology (FMA-TERM) that stores facts about the terms used to denote the entities of the FMA ontology (FMA-OWL); Reversely, the FMA ontology metadata are represented by OWL 2 annotation of annotations. Advantages and possible means to bind the two even more closely are finally discussed.

2 FMA Ontology in OWL 2

The FMA ontology is intended to model *canonical* human anatomy that is, "the ideal or prototypical anatomy to which each individual and its parts should conform" [2] [3]. It contains more than 85,000 classes, 140 relationships connecting the classes and over 120,000 terms. The FMA describes anatomical entities, most of which are anatomical structures composed of many parts interconnected in complex ways. It is a very large and perhaps one of the most complex ontology in the biomedical sciences.

OWL 2 is the W3C standard for ontologies on the Semantic Web. OWL 2 provides several advantages for Life Sciences ontologies: interoperability, semantics, reasoning services. (1) *Interoperability* is important for shared use across different biological and medical domain. Once converted to OWL 2, ontologies become easier to be connected or combined with other ontologies. (2) *Semantics* (meaning) of terms is formally specified thanks to the underlying description logics. (3) Another major practical benefit is that it allows to exploit the multitude of existing OWL tools, in particular powerful *reasoners.* For example, the FMA in OWL 2 makes it *interoperable* with other life sciences ontologies, provides a *formalized* vocabulary of anatomy with a precise meaning of terms, supports reasoning, which is crucial for its design, maintenance, and quality insurance. OWL 2 higher *expressiveness,* in particular its metamodeling abilities, is also of major interest for conciliating ontologies and terminologies, as shown next.

The FMA ontology is implemented in Protégé frames[5] and stored in a MySQL database backend. Transforming it into OWL 2 is not a simple translation. It requires to specify the meaning of terms in logics and to express anatomical knowledge, which was not explicit, by logical statements (axioms). This raises several issues. First, different types of information are embedded in Protégé FMA. Apart from the domain knowledge concerning the anatomical entities, the FMA includes meta-level knowledge. The problem is that interpreting both knowledge in the same model might lead to undesired consequences because of their interactions. Two solutions are proposed: an OWL 1 DL ontology *without* metaclasses and an OWL 2 ontology *with* metaclasses (§2.1). A second challenge is to provide formal definitions and axioms that are *semantically* correct from an *anatomical* viewpoint. The idea is to use lexical patterns for it (§2.2). Thirdly, given the large size of the FMA, it is essential to *automatically* generate the OWL axioms. A highly flexible tool (§2.4) enables to create not only a single ontology but several 'FMA-OWL' customized variants (§2.3).

[5] the frame-based system developed by Stanford Center for Biomedical Informatics Research

2.1 Metamodeling

Protégé Metaclasses. In FMA Protégé frames each anatomical entity is modeled both as a class and a metaclass[6]. At the domain level, classes describe the anatomical entities. At the meta-level, metaclasses serve several purposes. They associate metadata to the anatomical entities, for example they attach to the class Heart its author 'JOSE MEJINO, MD', preferred-terms 'Heart' in English, 'Cor' in Latin, Non-English equivalent 'coeur' in French, its definition, synonyms, FMAID, etc. Metaclasses are also used to define 'templates' for some given types of entities. For example, the metaclass *Organ With Cavitated Organ Parts*, is intended to specify the common template of all the organ types (species) that have cavitated organ parts. Metaclasses are organized into a subclass hierarchy. The metaclass *Heart*, is a subclass of *Organ with cavitated organ parts*, itself subclass of *Organ*, of which it inherits the slots, facets, etc., e.g.; *bounded by* with range *Surface of organ*, *arterial supply* with range *Artery, Arteriole, Arterial plexus* etc. On the opposite, at the class level, the own slots, e.g.; *part of, bounded by, arterial supply*, are assigned particular values. Thus, the structure of an anatomical entity, e.g.; a canonical *Heart,* can be specified as being an *Organ With Cavitated Organ Part*, having a *Right atrium*, *Left atrium, Right ventricle, Left ventricule* as parts, being *bounded by Surface of heart,* having *Right coronary artery* and *Left coronary artery*, etc., as *arterial supply*.

There exist earlier conversions of the FMA to OWL [6] [7] but they are not satisfying: directly translating metaclasses into OWL 1 leads to OWL Full, simply removing them makes the knowledge encoded at metaclasses lost. We propose two other solutions: an OWL 1 (2) DL ontology without metaclasses still capturing their knowledge, an OWL 2 ontology with metaclasses.

OWL 1 Ontology Without Metaclasses: to get an OWL 1 DL ontology, requires the deletion of the FMA higher order structure. The problem however is to capture the information embedded at metaclasses. Before OWL 2 existed, the solution was to replace metaclass instantiations by subclass axioms and to convert metaclasses into ordinary OWL classes [4]. This did not introduce significant change, because "*all concepts in the Anatomy Taxonomy are subclass of a superclass and also an instance of a metaclass*". As metaclasses specify a given "template" of classes and classes specify the structure of their instances, property restrictions at metaclasses are interpreted as ako closure axiom and approximated by universal restrictions, while restrictions at classes are translated into existential restriction.

The *OWL 2 Ontology With Metaclasses* relies on OWL 2 metamodeling new features [10], punning and enhanced annotations, respectively used for templates and for metadata. While OWL 1 DL required a strict separation between the names of classes and individuals, OWL 2 relaxes this separation. Thanks to *punning*, it is now possible to use the same term to refer to different types of entities, namely to a class and an individual, while retaining decidability [1]. Thus, the name `Heart` can be used both for the metaclass `Heart` and for the class `Heart`, instance of `Organ with cavitated organ parts`. Removing instantiation links is no more mandatory and using metaclasses that reflect more accurately the intent of FMA

[6] In FMA frames, each anatomical entity is modeled both as a metaclass and as a class. "… *for enabling the selective inheritance of attributes*" [3].

templates is possible. *Enhanced annotations* are used for the metadata attached to the FMA entities. While OWL 1 allowed extralogical annotations, such as a label or a comment, OWL 2 additionally allows for annotations of axioms and of annotations themselves. In FMA frames, properties such as preferred name, synonyms, non-English equivalents, etc. are modeled by slots assigned with individuals of the Concept name class as values. As they concern metadata and not data the domain of anatomy, using OWL 2 annotations of annotation is more appropriate: the domain and meta-level data are no more confused and do not interact. Besides, a huge number of individuals are thus removed. For example, the class `Heart` (1) is annotated by the label "Coeur"@fr (4), the labeling itself being annotated (2) by its creator JOSE MEJINO MD (2), date (3), FMAID "217079" (4), publisher, etc.

```
(1) Declaration(Class(:Heart))
(2) AnnotationAssertion(Annotation(dc:creator "JOSE MEJINO MD"^^xsd:string)
(3) Annotation(dc:date "Thu May 12 142434 GMT-0800 2005"^^xsd:date)
(4) Annotation(:FMAID "217079"^^xsd:string)... rdfs:label :Heart "Coeur"@fr)
```

2.2 Formalization in OWL

Another main challenge is to enrich the FMA with formal definitions and axioms having a sound anatomical meaning. The formalization process has two main steps. The first step formalizes the FMA frames syntax in OWL, the second step the FMA semantics. While the first transformation closely mirrors the FMA native model, the latter pushes the logical formalization further, adding new definitions and axioms to express semantics which was not stated in frames. Partly for historical reasons (in 2005 OWL 2 did not exist), the first step transforms FMA from frames to an OWL 1 DL ontology (*FMA-OWL v1*), the second step brings it to OWL 2 (*FMA-OWL 2*).

Formalizing Frames Syntax in OWL 1. The transformation of the frames syntax in OWL relies on 2005 rules [4]. In short, Protégé classes and slots are converted into OWL classes and properties, with the specified domain and range. Slot characteristics (inverse, symmetric, functional) are translated using corresponding OWL constructs. Values of own slots of classes are converted either into OWL values of annotation properties or into existential property restrictions. As said above, property restrictions defined at metaclasses or classes are respectively transformed into universal or existential property restrictions and metaclass instantiation is replaced by a subclass relation. However, as the earlier program did not scale up and was not robust, it has been much revised. The new implemented mapping of the syntax can now handle the *entire* FMA and overcome the changes of FMA successive updates.

Formalizing FMA Semantics in OWL 2. The second step pushes the logical formalization forwards. The semantics of the FMA ontology is enriched in several ways: (a) classes definitions are automatically generated from patterns; (b) meanwhile numerous related axioms are automatically created or moved (c) New properties characteristics are added; As described §2.1 (d) OWL annotations of annotation are used for metadata (e) OWL 2 metaclasses are created, but can be omitted on demand.

(a) Class definitions. An important shortcoming of the 2005 ontology was its class definitions. Class expressions were built from one uniform property, e.g.; *constitutional part*. However, all anatomical entities cannot be uniformly defined from the same properties [4]. New formalization rules are now defined that provide safe definitions. The key idea is to exploit *lexical* patterns of the FMA vocabulary and implicit properties omitted in such names (joined to the inference power of OWL). For example, it is very likely that the pattern `Left_A` (e.g., `Left_Hand`) denotes all A (`Hands`) that have left *laterality*, that `Left_superior_cervical_ganglion` means all the left and superior cervical_ganglion, `Region_of_cytoplasm` all the regional parts of cytoplasm etc. As the new rules create different forms of definition depending on each pattern, the patterns are basically unambiguous and moreover, their meaning was checked with FMA authors[7], all class definitions and axioms introduced in this manner are fully reliable. At the moment, two types of patterns are supported: (i) Pattern `P_A` denoting symmetrical siblings with an opposite anatomical_coordinate, e.g., `Left_A/Right_A`, `Anterior_A/Posterior_A`, `Inferior_A/Superior_A` etc., or an opposite gender, e.g.; `MaleA/FemaleA` and (ii) Pattern `A_of_B` denoting parts of entity, e.g., `Lobe_of_Lung`. Classes are incrementally defined as follows.

• Pattern P_A. At first, the `Anatomical_coordinate` subclasses are defined. `Primary_Anatomical_coordinate` are specified via property value restrictions, for example, axiom (1) states that *`Left`* denotes all objects with left laterality. `Binary_Anatomical_coordinate` are defined as an intersection of `Primary_Anatomical_coordinate` classes. For example axiom (2) states that `Left_superior` refers to all objects having a left and superior anatomical_coordinate. Entities of pattern `P_A`, where `P` is a `Primary_Anatomical_coordinate` subclass, are then provided definitions. For example, axiom (3) states that `Left_Hand` (resp. `Right_Hand`) denotes all hands having left laterality.

```
(1) EquivalentClasses(:Left ObjectHasValue(:laterality :individual_Left))
(2) EquivalentClasses(:Left_superior ObjectIntersectionOf(:Superior :Left))
(3) EquivalentClasses(:Left_Hand ObjectIntersectionOf(:Hand :Left))
(4) EquivalentClasses(:Lobe_of_Lung
       ObjectIntersectionOf(:Anatomical_Lobe
       ObjectSomeValuesFrom(:regional_part_of :Lung)))
(5) EquivalentClasses(:Region_of_cytoplasm
       ObjectIntersectionOf(:Region_of_cell_component
       ObjectSomeValuesFrom(:regional_part_of :Cytoplasm)))
(6) SubClassOf(:Hand ObjectExactCardinality(1 :laterality
       ObjectOneOf(:individual_right :individual_left)))
```

• Pattern A_of_B. In most cases a name A_of_B is a contraction formed from A and B, that omits some property *p* relating the entities A and B. The idea for providing semantics to entities `A_of_B` is to build a class expression from that relation. The missing property is recovered from the list of property restrictions attached to the class. For example, axiom (4) expresses that `Lobe_of_Lung` refers to all

[7] In a very few cases, ambiguity was solved with .

anatomical lobe that are a *regional_part_of* some *lung*. A particular process is defined for A_of_B where A is Region, Zone, Segment, Subdivision. From FMA authors, all 'region' classes of the FMA denote *regional parts,* further distinguished on the types of boundary used to define the region, for example *Organ segment* is an organ region with one or more anchored fiat boundaries, *Organ zone* is an organ region with one or more floating fiat boundaries. At the moment, the p handled are only the part_of properties and subproperties (e.g.; regional_part_of) but this will next be extended to other relationships.

(b) Axioms. The lexical patterns are not only used for class definitions, but also for handling - creating/removing/moving - axioms:

• Disjointness and subclass axioms. While the sibling symetrization process provides semantics to classes of pattern P_A, it operates other tasks at the same time: 1° it adds relevant subclass axioms. 2° it detects and repairs errors or omission in the native FMA (for details see Algorithm 1). For example, while the meaning of Left_Hand is formalized by the equivalent class axiom (3) meanwhile, several subclassOf axioms are added: for example axiom (6) asserts that each hand necessary has exactly one left or right laterality and the axiom DisjointClasses(*Left Right*), states that nothing can be both left and right. In fact, for each modality, only one single disjointness axiom is created to state that nothing can have two opposite modalities. Hence, all Left_A and Right_A, e.g.; left and right hands are inferred to be exclusive, and much less axioms are used. The algorithms implemented for each pattern are quickly sketched below.

Algorithm 1. The process for symmetrical siblings first parses all names of classes to get the terms matching a specific prefix P_ where P is a subclass of Primary_Anatomical_coordinate (e.g. Left). For each class P_A, (e.g. Left_A/Right_A), if A exists and A (or Anatomical_A) is a direct superclass of P_A, then several axioms are created respectively for P_A, its sibling and its father, according to the following rules: (1.1) each time A has a child P_A, A should have the pair as children, unless exceptions; (1.2) each time A has two symmetrical children, e.g.; Left_A and Right_A, and A has an existential restriction on a *part* property or subproperty, the two siblings should have symmetrical restrictions (modulo symmetry); (1.3) if a (symmetrical) restriction is present in two symmetrical siblings but not in their direct superclass, the relevant abstracted restriction is added to it. For example, as Left_Hand and Right_Hand have restrictions ObjectSomeValuesFrom(:*constitutional_part: Investing+fascia+of+left+hand*) (resp. *Investing+fascia+of+right+hand*), the missing axiom subclassOf(*Hand* ObjectSomeValuesFrom(:*constitutional_part*:*Investing+fascia+of+ +hand*)) is created; (1.4) as explained above, for each P_A, two axioms are created: a class axiom EquivalentClasses(:*P_A* ObjectIntersectionOf(:P :A)) and a subclassOf axiom like (6) for example, which asserts that each A necessary has exactly one left or right laterality SubClassOf*(:A* ObjectExactCardinality*(1 :laterality* ObjectOneOf*(:individual_Left :individual_Right)))*.

*Algorithm 2***.** Similarly, the process first parses all names of classes to get the terms that match the pattern A_of_B. The EquivalentClasses(:*A_of_B* ObjectIntersectionOf(*:A* ObjectSomeValuesFrom*(:p_of :B)*)) axiom is created in any of the following cases: (2.1) if the direct superclass of `A_of_B` is `A` or `Anatomical_A` and `A` has a restriction on a *part_of* property or subproperty `p_of`: `SubClassOf(:A ObjectSomeValuesFrom(:p_of :B'))` with `B'` direct superclass of B (e.g. `Ganglion_of_cranial_nerve`). (2.1b) if `B'` is not a direct superclass of `B` (it may be a distant ancestor) but `A` or `Anatomical_A` exists and `B'` has a restriction for the *inverse* p of p_of: `SubClassOf(:B' ObjectSomeValuesFrom(:p : A_of_B'))`; (2.2) if the direct superclass of `A_of_B` is `A` or `Anatomical_A` and `A_of_B` has a restriction `SubClassOf(:A_of_B ObjectSomeValuesFrom(:p_of : B))` (e.g. `Tendon_of_biceps_femoris`). For example, as the direct superclass of `Lobe_of_Lung` is `Anatomical_Lobe` and `Lobe_of_Lung` is a subclass of `regional_part_of some Anatomical_Lobe`, an axiom `EquivalentClasses(:Lobe_of_Lung ObjectIntersectionOf(:Anatomical_Lobe ObjectSomeValuesFrom(:regional_part_of: Lung)))` is created. (2.3) A specific process handles classes A_of_B where A is *Region_of, Zone_of, Segment_of, Subdivision_of* (1273 classes). It defines `A_of_B` as `regional_part` of B, like axiom (5) for `Region_of_cytoplasm`.

• Completing or compacting axioms. In canonical anatomy, if an entity A has some part B, then reversely B should also have some part A (which is not logically equivalent). 669 missing subclassOf axioms expressing such 'symmetrical' restrictions are created. On the other hand, based on inference, several axioms are removed: if all the subclasses of A have a same existential restriction, it is removed from the subclasses and moved up to A.

(c) Properties characteristics. OWL 2 allows new characteristics of object properties. According to FMA authors, *part, regional_part, constitutional_part, systemic_part, member* and their inverse are asserted to be transitive, irreflexive, asymmetric, *continuous_with* and *connected_to* are symmetric, and *continuous_with* is reflexive.

2.3 FMA-OWL in OWL 2

Table 1: Metrics of FMA-OWL ontologies

File	Size	Classes	Class axioms	Expressivity
FMA-OWL 1 from FMA 2005				
#1 without N&S	41,6	41648	236208	$\mathcal{ALCOIF}(\mathcal{D})$
#2. with N&S	40,8	41648	230690	$\mathcal{ALCOIF}(\mathcal{D})$
FMA-OWL 2 from FMA 3.0 2008				
#3. without MTC.	256	85005	263389	$\mathcal{SROIQ}(\mathcal{D})$
#4. with MTC.	314	85005	261331	$\mathcal{SROIQ}(\mathcal{D})$

Complete representations of the entire FMA are now available in OWL 2. An OWL 2 ontology[8] without metaclasses ($\mathcal{FMA}$-$\mathcal{OWL}$2_noMTC Table 1 #3) has been generated from FMA 3.0. It includes all FMA classes and properties (except *homonym_of* and *homonym_for,* discarded in agreement with FMA's authors), the new class definitions

[8] http://gforge-lirmm.lirmm.fr/gf/download/docmanfileversion/211/743/FMA_owl2_noMTC_100417.zip

and axioms, retains transitivity but voluntarily ignores irreflexivity and asymetry. This ontology offers 15084 new definitions of classes, 16113 disjointness axioms; 85467 initial axioms are removed and replaced by one single axiom (next inherited), 15 subproperties axioms and 228263 annotations. 7664 class definitions are obtained from the pattern `A_of_B`, while 7333 from the pattern `Left_A/Right_A`. Another OWL 2 ontology *with* metaclasses is also available (*FMA-OWL*2_noMTC Table 1 #4)[9]. Which ontology should be chosen as the 'standard' FMA-OWL depends on future improvements of the native FMA, mainly of its templates. Thanks to the FMA-OWLizer tool (§2.4), several other versions of various size and complexity can also be generated to fit specific applications needs. For example, *FMA-OWL v1* (Table 1) are OWL 1 smaller ontologies (41 Mb) issued from FMA 2005, of which left/right leaves are cut, without (#1) or with (#2) equivalent classes (N&S) built from the *constitutional_part* property. Ontologies obtained from the recent FMA 3.1 update have also been generated. But, *all* these FMA ontologies are still unstable and exhibit many errors that should be fixed. As the FMA is being incrementally developed and repaired, FMA-OWLizer is a highly helpful tool for generating FMA-OWL updates.

The FMA-OWL ontologies are perhaps the largest and most complex OWL ontologies available, and reasoning with FMA-OWL proved to be a real challenge. No reasoner could classify them so far. Recently HermiT [8], having a special blocking strategy for FMA like ontologies with lots of unsatisfiable classes, could process them in a reasonable time. *FMA-OWL* 2 (Table 1 #3 - 2010-03-11) has 65,753 unsatisfiable classes out of 85,005. The time for classification, including loading and preprocessing was 58m 12s 929ms (by Birte Glimm). *FMA-OWL* v1 with constitutional-part for N&S (#1) had 33,433 unsatisfiable classes out of 41,648, and the time for classification was 33m 46s 55ms.

2.4 FMA-OWLizer

FMA-OWLizer is a friendly and easy to use tool that automatically generates an FMA ontology in OWL. It can process *all* existing public FMA versions, FMA 2005 version, FMA3.0 (2008), April 2010 FMA 3.1 update. It is highly flexible, allowing to provide a customized ontology adapted to the users' needs for their application, whatever OWL developers, FMA authors, or application designers. The main parameters have to be selected via a friendly *graphical user interface* (http://www.lirmm.fr/tatoo/IMG/pdf/FMA-OWLizer.pdf), while other ones should be configured via configuration files, like for example, the file 'classes_to_delete.txt' stating the classes to be removed. FMA-OWLizer includes many options. It is possible to select the source file as input, to have metaclasses or not, to choose the properties to be included, to customize the class and property axioms in various ways: to supply particular class definitions by designating the properties, e.g.; *constitutional_part, bounded_by* etc of the equivalent classes axioms, to include/remove all the subclass axioms (e.g. for performance tests), to configure properties characteristics. For example, to get an OWL 2 DL ontology that reasoners can process, it is recommended to select 'ignore irreflexive and asymetric'.

[9] http://gforge-lirmm.lirmm.fr/gf/download/docmanfileversion/212/744/FMA_owl2_MTC_100421.zip

Otherwise, as the properties *part* and their inverse are transitive, asymmetric and irreflexive, the ontology would violate the OWL 2 restriction that only simple roles can be used in asymetric and irreflexive object property axioms. It is also possible to choose the concrete syntax to store the ontology (RDF/XML, OWL/XML, Functional Syntax), to select French or English for the GUI. FMA-OWLizer is a local Java program designed and developed specifically for the FMA. All processes are performed via the OWL API 3.0, and benefit of its functionalities. The GUI is achieved with the Swing/AWT Java graphics libraries and is multilingual support (bundle files), thanks to the CISMeF Utils platform.

3 HMTP Portal & FMA Terminology in OWL

Health terminologies are intensively used for resources indexing and retrieval, or data annotation. The UMLS includes more than 130 terminologies. Far behind English, French is the second most used language in health terminologies. The Health Multi-Terminology Portal (HMTP) developed by the Rouen University Hospital aims at providing a portal to share and *access* the terminologies available in French. Its goal is to offer a framework that enables French terminologies to be semantically interoperable and to interact more effectively.

3.1 HMTP Portal

HMTP Terminologies. HMTP (http://pts.chu-rouen.fr) integrates health terminologies that deal with various aspects of health. It presently includes twenty two health terminologies and classifications, among which: SNOMED International terminology for clinical term [12], MeSH thesaurus (version 2010) for indexing and searching medical information in Pubmed, achieved by the National Library of Medicine, ICD10 (International Classification of Diseases, 10th revision), MedDRA terminology for adverse effects (http://www.meddramsso.com/), Orphanet for rare diseases including genes and symptoms (www.orpha.net), several terminologies developed by the World Health Organization (WHO): WHO-ART, WHO-ICPS, WHO-ATC, WHO-ICF etc. Most of them are bilingual (English & French). Additional terminologies/ontologies are currently being integrated, in particular SNOMED CT and LOINC (Logical Observation Identifiers Names and Codes).

HMTP Functionality includes the ability to (a) search, (b) browse terminologies, (c) visualize their hierarchy and (d) search resources indexed with their terms:

(a) HMTP allows searching for terms or words among the preferred terms of concepts, synonyms, definitions or codes in all terminologies. It is possible to make a search in French or English (if the terminology data include the translation). There are three search modes. The default mode is a truncated search that searches for a full expression from a truncated entry. For example, searching for ‘Infarction’, HMTP provides 15 answers in MeSH: angina unstable, Anterior Wall Myocardial Infarction, cerebral infarction, heart rupture, post-infarction, infarction, infarction anterior cerebral artery etc. The second mode is a *stemming* search that uses roots of words, removing usual prefixes and suffixes, e.g., searching for ‘cérébelleuses’ without stemming provides 8 answers, while with stemming 43, as they include terms like ‘cerebelleux’. The last mode is an *exact* search that looks for the exact expression entered, e.g.; ‘Myocardial infarction’.

(b) It is possible to browse all terminologies. In particular, the 'Relations' link offers a critical means for a user to navigate within and between terminologies, and thus to discover new information about relations between concepts. Intra-terminology relations state associations between concepts within a given ontology, while Inter-terminology relations state associations between concepts of different terminologies. The later rely on mappings that are manually inserted by professionals or obtained either from the UMLS or by specific techniques of natural language.

(c) When a term belongs to a hierarchy, the Hierarchy Tab is displayed, which enables to visualize the term position in the (multi-)hierarchy

(d) Additional Tabs allow to search for a term in Pubmed, Doc'CISMeF, or InfoRoute (French Infobutton [11]).

User profiles enable to give a restricted access to HMTP among specific terminologies. HMTP relies on dedicated Web Service that send a SKOS answer to the server. HMTP is a Graphic User Interface that displays the data resulting from the selection or search operated by the Web Services.

3.2 HMTP representation

HMTP key-stone is the ‘Unifying Model of Vocabulary’ (UMV2). Its principles are inspired by RDF and UML *reification*.

Unifying Model of Vocabulary. UMV2 is a common model for all terms, whatever their terminology. It can be viewed as a meta-model or an upper ontology designed to support broad semantic interoperability between a large number of terminologies accessible under it. The basic concept of UMV2 is `Descriptor`, which is quite similar to skos:Concept. `Descriptor` has several attributes, e.g.; `label` refers to the preferred term used to name a descriptor in natural language, `synonym` refers to its synonyms in different languages, `notation` to its identifier. Another key concept is `Association`, used to model a relation between two descriptors as a class. It is quite similar to UML association classes. The concept `Group`, close to UML aggregation, is used to model a set of descriptors, for instance a terminology, e.g.; ICD10 or a group of terminologies, e.g.; SMQ, *Standardised MedDRA Queries*). In turn, the components and sub-components of ICD10, *Chapter, Category*, etc. are modeled as groups. For example, Chapter XI that list all the “Diseases of the digestive system”, further divided into the categories from K00 to K93 (K00-K93), are all modeled as groups. UMV2 includes other general classes such as BT-NT for a hierarchical association, or RT-RT for “See Also” associations. UMV2 offers general properties that can be used to relate the HMTP descriptors. For instance, the properties `BT` and `NT` of a `BT-NT` asssociation are used to assert that the two descriptors respectively referred by `BT` and `NT` have a hierarchical link. `BT` states the more general term of an association (‘broader’) and `NT` the more specific (‘narrower’). These notions are similar to skos:broader and skos:narrower. The property `RT` states that two descriptors have *some* relationship. UMV2 also provides ‘mapping’ properties imported from SKOS, used to align HMTP descriptors: `skos:closeMatch`, `skos:exactMatch`, `skos:broadMatch`, `skos:narrowMatch` and `skos:relatedMatch`.

Each terminology *T* of HMTP is built as an extension of UMV2, called *UMV1 T*, e.g.; UMV1 FMA, UMV1 ICD10. Each particular extension includes a specific specialization of `Descriptor` and specific specializations of `Association`, e.g.; UMV1 FMA defines the descriptor `FMAEntity`, the associations `FMAinnervation FMAdrainageveineux` (venous drainage), while UMV1 ICD10, the `ICD10categorie` descriptor, the `ICD10Exclusion` association.

OWL representation. HMTP terminologies are implemented as OWL ontologies. The UMV2 concepts `Descriptor, Association, Group` and their UMV1 specializations are unary relations represented by OWL classes, e.g.; `FMAdelimitation` (Table 2 #1). `Descriptor` attributes, `label`, `synonym`, etc., are represented by OWL Data Properties. UMV2 general properties, such as `BT`, `NT`, `RT`, `skos:exactMatch` and the specialized properties of each UMV1 T ontology are binary relations represented by Object Properties (#9 #12). The terms of each terminology are represented by individuals of the relevant descriptor subclass. For example, the terms of the FMA terminology, `Heart, Lung, Surface of Heart, Right atrium,` etc. are individuals of the descriptor `FMAEntite`, e.g.; `FMA_7088` (#13) for `Heart` (#15). A relation between two terms is represented by an OWL individual of the concerned `Association` joined with two property assertions. For example, the individual `FMA_ bounded_by~FMA_7088-FMA_7167` of the `FMAdelimitation` subclass of `Association` (#16) asserts that the term `Heart` and `Surface of heart` are related via a relation, which source is `Heart` (#17) and target `Surface of Heart`(#18). The individual `FMA_BTNT~55209` of the `BT-NT` class (#19) asserts that the terms `Heart` and `Organ_With_Cavitated_Organ_Parts` have a hierarchical relation, the broader term (#20) being `Organ_With_Cavitated_Organ_Parts (FMA_55673)` and the narrower (#21) being `Heart (FMA_7088)`.

3.3 FMA Terminology in OWL

The FMA-OWL ontology has been translated into a terminology (FMA-TERM) compliant with the UMV2 metamodel, by a *reification* process. FMA-TERM is integrated into HMTP portal: http://cispro.chu-rouen.fr/pts_site/index.html?lang=en. To consult it, click on “Connection”; login=fmauser, password=fmapass.

Table 2 FMA-TERM OWL ontology: Heart example in functional syntax

```
---------------- Classes ---------------
1.  Declaration(Class(:FMAdelimitation))
2.  AnnotationAssertion(rdfs:label :FMAdelimitation "Délimitation"@fr)
3.  AnnotationAssertion(rdfs:label :FMAdelimitation "Bounded by /
    bounds"@en)
```

```
4.  SubClassOf(:FMAdelimitation publishing:Association>)
5.  SubClassOf(:FMAdelimitation ObjectAllValuesFrom(:FMAdelimite
    :FMAentity))
6.  SubClassOf(:FMAdelimitation ObjectAllValuesFrom(:FMAestDelimitePar
    :FMAentity))
7.  SubClassOf(:FMAdelimitation ObjectMinCardinality(1 :FMAdelimite))
8.  SubClassOf(:FMAdelimitation ObjectMinCardinality(1
    :FMAestDelimitePar))
----------------- Object Properties ----------------
9.  Declaration(ObjectProperty(:FMAdelimite))
10. ObjectPropertyDomain(:FMAdelimite :FMAentity)
11. ObjectPropertyRange(:FMAdelimite :FMAentity)
12. Declaration(ObjectProperty(:FMAestDelimitePar)), etc.
----------------- Individuals ----------------
13. ClassAssertion(:FMAEntity :FMA_7088))
14. AnnotationAssertion(rdfs:label :FMA_7088 "Coeur"@fr)
15. AnnotationAssertion(rdfs:label :FMA_7088 "Heart"@en)
16. ClassAssertion(:FMAdelimitation :FMA_bounded_by~FMA_7088-FMA_7167)
17. ObjectPropertyAssertion(:FMAdelimite :FMA_bounded_by~FMA_7088-FMA_7167
    :FMA_7088)
18. ObjectPropertyAssertion(:FMAestDelimitePar :FMA_bounded_by~FMA_7088-
    FMA_7167 :FMA_7167)
19. ClassAssertion(publishing:BT-NT :FMA_BTNT~55209)
20. ObjectPropertyAssertion(publishing:NT :FMA_7088)
21. ObjectPropertyAssertion(publishing:BT :FMA_55673)
```

The transformation of the OWL 2 ontology FMA-OWL into the terminology FMA-TERM is relatively straightforward. FMA-OWL *classes* are mapped to OWL *individuals* representing terms in FMA-TERM, *object properties* are transformed into `Association` classes, `SubClassOf(`*`fma:A fma:B`*`)` axioms into `BT-NT` individual assertions, `SubClassOf`*`(fma:A`*` ObjectSomeValuesFrom`*`(fma:R fma:B))`* axiom into an individual and two object property assertions, FMA-OWL *preferred-name* annotations into labels, other *annotations* into datatypes. The IRI of the individuals are obtained from the FMAID of the FMA-OWL class. More precisely, each FMA-OWL *class*, e.g.; `Heart`, is mapped to an individual of the `Descriptor` subclass `FMAEntity` (#13 Table `2`), which IRI (`http://www.chu-rouen.fr/smts#FMA_7088`) is created from its FMAID (7088). *Object properties R*, e.g; `bounded_by`, `venous_drainage`, `innervation` etc., are transformed into `Association` subclasses (named `FMA`*`R`*) and two object properties with `SubClassOf`*`(A`*` ObjectAllValuesFrom`*`(R B))`* and cardinality restrictions axioms specified in UMV1 FMA. For example, `bounded_by` is mapped to `FMAdelimitation (#1) sub`class of `Association (#4)` and two object properties `delimite and estDelimitéPar (#9-#12)` are created with value restrictions `(#5-6-7-8)`. The `SubClassOf(fma:Organ_With_Cavitated_Organ_Parts fma:Heart)` axiom is transformed into the individual `FMA_BTNT~55209 (#19)` and the property assertions (#20-21), while `SubClassOf(`*`fma:A`*` ObjectSomeValuesFrom(`*`fma:R fma:B`*`))` into a class assertion `(#16)` and two object property assertions `(#17-18)`.

FMA-TERM is a lightweight $\mathcal{ALN}$(D) ontology with 39 classes, 41 object properties, 19 data properties, 136 subclass axioms and 218419 individuals. It includes 81042 FMA descriptors with 81020 unique English terms, 52040 unique English synonyms, 4436 unique French terms and 139 French synonyms. Twenty two FMA relationships were selected to be integrated. 52447 relations connect 39787 distinct classes and there are 80575 hierarchical BT-NT relations. Using HermiT for reasoning, exhibited a modeling error in FMA-TERM implementation. All Association classes are re-classified as FMA entities. These undesired inferences result of wrong domain assertions of object properties. Instead of `FMAEntity` (#10) the domain should be an `Association` subclass (`FMA_7088–FMA_7167` is not an `FMAEntity` but an individual of `FMAdelimitation`). The transformation of FMA-OWL into FMA-TERM is achieved by a Java J2EE parser with the SAX API; the resulting file is created thanks to Jena API.

4 Reconciling ontologies and terminologies via metamodeling

OWL ontologies and terminologies have respective advantages. Ontologies support powerful automated reasoning, while Terminologies support powerful searching and browsing services. Using HermiT for reasoning with the FMA-OWL ontology exhibited a huge number of inconsistencies due to modeling errors in the original FMA design (§2.3) and a modeling error in FMA-TERM ontology (§3.3). As already pointed out numerous inconsistencies are issued from conflicts between FMA metaclasses and classes or domain/range [4], for example conflicts between material vs immaterial, or left vs right laterality (e.g.; Internal pyramidal layer of *left* cerebral cortex is specified as a regional part of Internal pyramidal layer of *right* occipital lobe, etc.). Logical semantics, is crucial for detecting and repairing ontologies before using them in real world applications. OWL 2 support is vital for ontology engineering in Life Sciences. On the other side, HMTP is a tremendous resource for biomedical documents retrieval: it allows searching an expression among terms or synonyms of many terminologies, with truncated or stemming methods, navigating through terminologies and to visualize relations and hierarchies with a friendly user interface. FMA-TERM may play a central role in education as well. HMTP does not only provide a syntactic integration of various terminologies into a single framework. It offers semantic interoperability thanks to its UMV2 unifying vocabulary and inter-terminologies mappings. This is of great interest in practice. For example, the term 'acute myocardial infarction' of SNOMED alignment with 'myocardial infarction' of MeSH enables to automatically get an ICD10 encoding, useful for a socio-economical use, from clinical information initially coded with SNOMED.

A first step has been achieved to reconcile these two complementary views via metamodeling: a terminology is viewed as a metamodel of a domain ontology that states facts about its terms; it is stored in a separate OWL ontology (e.g.; FMA-TERM) compliant with the UMV2 metamodel, obtained by 'reification'. Reversely, the domain ontology (e.g.; FMA-OWL) metadata are represented by OWL 2 annotation of annotations. A clear separation between the two ontologies as presently, makes sense semantically and prevents from undesired interactions, but may also exhibit some shortcomings. Binding them more closely would allow to a) enrich both ontologies, b) maintain them consistent c) answer complex queries involving both domain and terminology knowledge, which is a crucial needs in Life Sciences, for example, to retrieve all the constituents, regions, vessels, innervations of Heart, all its relationships, or the organs that are `Organ_with_Cavitated_Organ_Parts`, using an OWL reasoner or SPARQL. A first idea might be to merge the two ontologies via punning, i.e. to use the same name, e.g.; `fma:Heart`, to refer not only to the class (1) and the metaclass (2), but also to the term, modeled as an individual (3). But this is not satisfying: `fma:Heart` would then denote an individual of two classes of completely different nature, `Organ_with_Cavitated_Organ_Parts` metaclass (2) and `FMAEntity` class (3), it might lead to unwanted inferences and inconstencies, and `FMAEntity` is clearly not a class of the Anatomy domain.

```
(1)Declaration(Class(fma:Heart))
(2) ClassAssertion(fma:Organ_with_Cavitated_Organ_Parts fma:Heart)
(3)ClassAssertion(fma:FMAEntity fma:Heart )
```

Using OWL 2 annotations is another possibility. Facts of a terminology correspond to annotations of the domain ontology. At the moment, the correspondence is only exploited one way, from the domain ontology to the terminology. Exploiting it also the other way, would enrich the domain ontology with the content of the terminology. For instance, the individuals of the Concept-name class transformed into OWL 2 annotations in FMA-OWL have exactly the same nature as the facts of FMA-TERM (§2.1), but these (meta)data are incomplete. Adding data from HMTP, such as UMLS synonyms, inter-terminologies mappings, etc., via OWL 2 annotations of annotations, would enrich FMA-OWL. The ontology would then include the FMA class and metaclass `fma:Heart`, and also useful metadata attached to them. Automatizing the transformation in both directions requires automated tools, which development is under way. Several perspectives may be considered. In the future, it might be worth to investigate whether a connection with the metaviews approach [15] would be useful. HMTP is influenced by UML, SKOS, and MONDECA tools (http://www.mondeca.com/). Migrating it to a more standardized view might facilitate its integration and interoperability with Bioportal (http://bioportal.bioontology.org/). The Unifying Model of Vocabulary UMV2 and UMV1 submodels might be defined directly in OWL 2, groups might be modeled via punning, axioms should use qualified cardinality restrictions, etc. Reversely, a Protégé plugin might be helpful to access HMTP services, particularly useful for French terminologies.

5 Conclusion

The paper has presented the FMA Ontology formalization in OWL 2, the French Health Multi-Terminologies Portal and its FMA Terminology obtained by reification. A first step has been achieved to reconcile the FMA ontology and terminology via metamodeling. The approach proposed should still be improved and generalized to other Life Sciences ontologies. Representing ontology and terminology knowledge in OWL 2 is very useful in practice. It enables to provide at the same time a clear semantics and powerful reasoning, to have a rich and reliable terminology for automated resources indexing and retrieval. Binding ontologies and terminologies even more closely by automated processes implementing their correspondence would provide an extremely valuable resource to the community. Both would be enriched and gain in content. It would facilitate answering complex queries involving the domain and metamodel and maintaining the two views consistent.

References

1. OWL 2 Web Ontology Language Document Overview W3C Recommendation 27 October 2009 http://www.w3.org/TR/owl2-overview/
2. C. Rosse, J.L. Mejino Jr, The Foundational Model of Anatomy Ontology Cornelius Rosse and José L. V. Mejino , In Burger A et al. (editors): Anatomy Ontologies for Bioinformatics: Principles and Practice, Springer,.59-118 ISBN 978-1-84628-884-5. 2008, New York.
3. C. Rosse, J.L. Mejino Jr, A reference ontology for biomedical informatics: the Foundational Model of Anatomy, J. Biomed. Inform. 36 (6) (2003) 478–500.
4. C. Golbreich, S. Zhang, O. Bodenreider, The Foundational Model of Anatomy in OWL: experience and perspectives, Journal of Web Semantics, Web Semantics: Science, Services and Agents on the World Wide Web, 4 (3) (2006) 181–195.
5. O. Dameron, D.L. Rubin, M.A. Musen, Challenges in converting frame based ontology into OWL: the Foundational Model of Anatomy case-study, in: AMIA Annual Symposium, Washington DC, 181-185, 2005.
6. O. Dameron and Julie Chabalier. Automatic generation of consistency constraints for an OWL representation of the FMA 10th International Protégé Conference, 2007.
7. Natalya F. Noy, Daniel L. Rubin, Translating the Foundational Model of Anatomy into OWL, Journal of Web Semantics, Web Semantics: Science, Services and Agents on the World Wide Web, 6 (2008) 133–136.
8. Birte Glimm, Ian Horrocks, and Boris Motik, Optimized Description Logic Reasoning via Core Blocking (submitted).
9. W3C OWL Working Group, OWL 2 Web Ontology Language Document Overview W3C Recommendation 27 October 2009, http://www.w3.org/TR/owl2-overview/.
10. C. Golbreich and Evan K. Wallace, OWL 2 Web Ontology Language New Features and Rationale W3C Recommendation 27 October 2009, http://www.w3.org/TR/owl2-new-features/
11. Darmoni, SJ; Pereira, S; Névéol, A; Massari, P; Dahamna, B; Letord, C; Kedelhué, G; Piot, J; Derville, A & Thirion, B. French Infobutton: an academic and business perspective. AMIA Symp., Pages 920, IOS Press, 2008.
12. Lussier YA, Rothwell DJ, Côté RA. The SNOMED model: a knowledge source for the controlled terminology of the computerized patient record. Met Inf Med. 1998; 37(2): 161-4.
13. Stearns MQ, Price C, Spackman KA, Wang AY. SNOMED clinical terms: overview of the development process and project status. Proc AMIA Symp. 2001: 662-6.
14. Lindberg DA, Humphreys BL, McCray AT. The Unified Medical Language System Methods Inf Med. 1993; 32(4): 281-91.
15. Motik, B. ; Cuenca-Grau, B., Horrocks I. (2007). Making metalogical information in ontologies logical using metaviews. In ISWC 2007, Busan, Korea.

Acknowledgments. We are grateful to C. Rosse and O. Mejino for their advice and support and to Oxford team for their specialized classification algorithm, special thanks to B. Glimm.