Migrating the FMA from Protégé to OWL

Christine Golbreich[1], Songmao Zhang[2], Olivier Bodenreider[3]

[1]LIM, University Rennes 1, France
E-mail: Christine.Golberich@univ-rennes1.fr

[2]Institute of Mathematics, Chinese Academy of Sciences, Beijing 100080, P.R.China
E-mail: smzhang@math.ac.cn

[3]U.S. National Library of Medicine, 8600 Rockville Pike, MS 43, Bethesda, Maryland 20894, USA
E-mail: olivier@nlm.nih.gov

Abstract. This paper focuses on the migration of the Foundational Model of Anatomy from its frame-based representation in Protégé to its logical representation in OWL. First, it considers specificities of the FMA in Protégé that were taken into account for the migration, and presents some conversion rules defined for migrating FMA from Protégé 2.1 to OWL DL. Then, the incremental approach currently adopted is outlined. Preliminary results are reported, exhibiting the benefits of this work both for the FMA and for description logic systems.

1. Introduction

The long term goal of this project is to provide a service assisting the conversion of frame-based ontologies to OWL, in order to take advantage of the higher expressiveness and powerful reasoning services of its underlying description logic (DL). Converting frame-based ontologies to OWL becomes an important issue corresponding to general needs for interoperability and resources sharing on the Semantic Web. This trend is already observed in biomedicine, where thesauri are currently being migrated to OWL (e.g. Gene Ontology, MeSH, NCI).

The frame-based ontology under study is the Foundational Model of Anatomy (FMA, version 1.1), which was converted from Protégé 2.1 to OWL DL. The FMA was selected mainly because anatomy plays a central role in medicine. The FMA claims to be [2] "a reference ontology in biomedical informatics for correlating different views of anatomy, aligning existing and emerging ontologies in bioinformatics...". Next, from a representational viewpoint, it is pertinent to evaluate the relevance of OWL DL and benefits of DL reasoning services such as consistency checking, automatic classification and instance recognition, for large biomedical ontology such as the FMA. Finally, its complexity and size make converting the FMA into OWL a challenge for editing environments (e.g., Protégé OWL) and reasoners (e.g., Racer). In fact, the sheer size of FMA brought to light some computational difficulties with the systems used. For this reason, we adopted an incremental approach to converting the FMA in order to reduce the performance issue and investigate the other issues step by step. While [6] suggests a solution based on OWL Full, our paper presents preliminary results obtained from a conversion of a large subset of the FMA into OWL DL, and its benefits.

2. Methods

At a first glance, it might be thought that a Protégé ontology could be converted into OWL by a simple export function mapping Protégé primitives to OWL constructs. In fact, DLs have a higher expressiveness and migrating a frame-based ontology to OWL requires not only syntactic "translation", but also semantic "enrichment". Indeed, property restrictions such as allValuesFrom and someValuesFrom cannot be directly derived from the original ontology, where they are not specified. Additionally, satisfiability checking and classification strongly rely on the classes *logical* definitions. A reasoner (e.g., Racer) can only automatically classify the "defined" classes[1] - i.e. classes with at least one necessary and sufficient condition. In frames, all slots with a range defined at a class are considered as a set of necessary conditions. Identifying *necessary and sufficient* conditions is a major "enrichment" of the ontology. Finally, the FMA in Protégé makes extensive use of metaclasses, which are not allowed in OWL DL. Each concept of the Anatomy Taxonomy is modeled both as a metaclass and as a class, instance of a metaclass. This was the "technical solution for enabling the selective inheritance of attributes" [2]. For example, `Heart` is defined (1) as a metaclass, subclass of `Organ+with+cavitated+organ+parts`, itself subclass of `Organ`, and (2) as its instance. At the meta level (1), `Heart` inherits all the slots, facets, characteristics of its superclassses, e.g. it inherits from `Organ` the slot `bounded+by` with multiple values allowed in the class `Surface+of+organ`, the slot `arterial+supply` etc. At the instance level (2) the class slots of `Heart` are assigned particular values e.g. `bounded+by` is filled with `Surface+of+heart`, `arterial+supply` with `Right+coronary+artery` and `Left+coronary+artery` etc. Simply ignoring metaclasses in the migration would not be satisfactory, because all the knowledge specified at the meta level would be lost. The adopted solution is, for each concept, to merge the two levels of representation in Protégé onto a single OWL class.

[1] except if a property has a domain (or range) that is a primitive class, which can coerce classes to be reclassified under the primitive class that is the domain or range of the property (§4).

2.1 Migration rules for FMA

The migration was achieved from the Protégé text files FMA.pont and FMA.pins. Two types of migration rules have been designed: "translation" and "enrichment" rules. Translation simply draws on the structural correspondence between Protégé and OWL constructs, e.g. inverse, symmetry. By contrast, enrichment rules interpret the underlying principles of FMA, so as to introduce logical features such as allValuesFrom and someValuesFrom property restrictions. Here are some of the migration rules we created (see [5] for details).

Class information. Classes and properties – stored at (meta)class level in Protégé – are converted to OWL classes and properties with specified domain (`rdfs:domain`) and range (`rdfs:range`). The following property characteristics are translated into OWL: inverse (`owl:inverseOf`), logical characteristics, i.e. transitive, symmetric (`owl:TransitiveProperty`, `owl:SymmetricProperty`), and also property cardinality and restrictions (`owl:FunctionalProperty`, `owl: hasValue`). In practice, the main rules are the following.

- **Top level slots**, specified in Protégé to save top-level slot information, are converted into `DatatypeProperty` or `ObjectProperty` with the relevant range and restrictions, according to their definition. For example, a top-level slot with type SYMBOL is converted (1) into a `DatatypeProperty` with `range #Boolean` (e.g., `has_boundary`) when its allowed values are TRUE FALSE, (2) into an `ObjectProperty` with an enumerated class (`oneOf`{allowed-values}) as range when its allowed values are different from TRUE FALSE and (3) into an `ObjectProperty` with the union of the allowed (meta)classes as `range` when it has allowed classes (or allowed-parents).

  Single-slots with cardinality 0 1 are converted to `FunctionalProperty`.

  Inverse-slots. If top level slot have "inverse-slot", it is converted to SymmetricProperty or inverseOf. If the inverse value is itself, it is SymmetricProperty with range assigned to its domain, else it is inverseOf. Thus, for example, the top level slot `has+boundary` is converted to a `DatatypeProperty` with `range #boolean`, with a `FunctionalProperty` restriction, the multislot `bounded+by` is converted to an `ObjectProperty` with `range #Physical_anatomical_entity`, and `inverse #bounds`.
- **Slots at class** enable to define the `domain` of an OWL property and to refine its value constraints: if p is slot of class A_1, then A_1 becomes the domain of p e.g. `#Physical_anatomical_entity` becomes the domain of `has_boundary`; if the same slot p occurs in class A_2, then the domain of p is the union of A_1 and A_2. Optimization of domain $c_1 \cup \ldots \cup c_n$ has been done: if c_i is descendant of another class according to two levels of is-a, then c_i is removed from the domain (reducing the domain of "arterial supply" from 4007 classes to 4).

  Allowed-parents, allowed–classes, value define the allowed values of properties at class. They are converted into necessary conditions expressing value constraint on the property for this class: if p is slot of class A specified with allowed-parents or allowed-classes (resp. with value), then p is converted by a necessary condition at class A expressing `owl:allValuesFrom` (resp. `owl:hasValue`) restrictions to the union class of all its allowed-parents or allowed-classes.

  Is-a is converted into subsumption axioms (subClassOf): A is-a B (if B is not USER nor :STANDARD-CLASS or :STANDARD-SLOT or RELATION) is converted to A `subClassOf` B (resp. is-a B_1 B_2 is converted to `subClassOf` $B_1 \cap$ `subClassOf` B_2), e.g. `is-a Anatomical+structure` is converted to `<rdfs:subClassOf rdf:resource="#Anatomical_structure"/> <rdfs:subClassOf>`.

Instance information. Values of properties – specified at instance level in Protégé – are converted by OWL annotation or property existential restrictions for the class. In practice, the rules are the following:

- **Non structural slots**. In Protégé slots such as preferred name, synonyms, UWDAID, definition, `author` etc., are defined at metaclasses [3]) for preventing their values to be propagated to their instances or subclasses. We used annotations on classes instead, which are allowed in OWL-DL. For example, UWDAID is an annotation (`<owl:AnnotationProperty rdf:ID="UWDAID">`) whose value for Heart is 7088.
- **Structural slots**. Another main use of metaclass is for "structural" slots, such as `part+of`, `custom+partonomy`, `bounded+by`, etc. It enables to specify each class for "canonical" anatomy thanks to the particular values assigned to its metaclass own slots, which are thus not propagated. For example, a "canonical" `Heart` is specified as having as custom partonomy exactly a `Right+atrium`, a `Left+atrium`, a `Right+ventricle`, a `Left+ventricule`, as being `bounded+by` exactly a `Surface+of+heart`. Structural own slots are converted by a necessary (or necessary and sufficient) condition at class A expressing, `owl:someValuesFrom` restrictions for p values to the union class of all the classes assigned to p. For example (`bounded+by Surface+of+heart)` is converted to a `someValuesFrom` restriction on property `#bounded_by`, which means that any instance of heart is necessarily bounded at least by one `#Surface_of_heart`. This preliminary representation of structural own slots is not complete and shall be improved soon (§4).
- **Instantiation** is converted by subsumption axioms: [A] of C (if C is not A itself nor is-a B) is converted into A subClassOf C (e.g., Heart is subClassOf of Organ_with_cavitated_organ_parts)

2.2 Incremental approach

About 40,000 FMA concepts and their slot values were extracted for conversion, i.e. 57% of the original 70,000 FMA concepts. Applied to this subset, the migration process resulted in 117,000 frames, including 40,000 classes and 77,000 conditions (necessary, and necessary and sufficient) on these classes. There are 155 properties and 85 individuals. It took about 15 minutes to load the FMA.owl file in Protégé OWL. Launched from Protégé-OWL, Racer classification failed. Running Racer classifier directly from Rice, problems with spatial resources occurred. Since Racer could not handle the whole FMA.owl file (in fact restricted to 2/3 of the whole FMA), as suggested by the Racer authors, we decided to test smaller versions so as to reduce the size and time problems and investigate eventual errors, adding more features incrementally. First a FMA OWL version without any properties was checked to test if the taxonomy could pass Racer. Then, we added equivalent class definition using only one property to test if defined classes could pass Racer, and successively introduced step by step necessary conditions for defined classes, annotation properties, datatype properties and attributed slots.

3. Preliminary results

The first test was successfully passed. To test classification with "defined" classes, the property selected was "constitutional part", resulting in 570 "defined" classes. This choice was partly motivated by a size issue: constitutional part is well populated in FMA, compared for instance to "custom partonomy" or "regional part", thus is computationally more significant. Obviously, such a definition is not "semantically" satisfactory since anatomical structures cannot be defined solely in terms of their constitutional parts (the same parts may belong to different structures), but it may be next refined. The classification of this FMA.owl file took about 25 minutes. About 300 classes were moved within the inferred hierarchy. Based on this definition, some classes were identified to be equivalent, e.g., `Wall_of_biatrial_part_of_heart` and `Wall_of_biventricular_part_of_heart`, since composed of the same constitutional parts. No inconsistencies were found. However, when datatype properties have been added, several inconsistencies were identified by Racer:

Inconsistencies of classes from boolean datatype properties. A class that inherits from its superclasses both *true* and *false* values for the same property is unsatisfiable. For example, "Zone of cell" is unsatisfiable because it is specified as having no mass, and on the other hand, is subsumed by "Material physical anatomical entity" from which it inherits mass. 116 classes were identified as unsatisfiable by Racer because of such inconsistencies.

Inconsistencies of classes from domain and range. `rdfs:range` (resp. `domain`) restrictions are global. Thus if p has class A' as domain and B' as `range`, and A has a property p with range B, then B must be a subclass of B' and A must be a subclass of A'. Conflicting definitions of global and local ranges or domains lead to inconsistencies. For example, "Surface of wrist" is unsatisfiable because the range of "2D part" is "Non-material physical anatomical entity", which is disjoint from its "2D part" `owl:someValuesFrom` restriction to class "Anatomic snuff box" which is a subclass of "Material physical anatomical entity".

Inconsistencies between metaclass and class definitions. For example, "Compartment subdivision" is defined as a subclass of "Anatomical cluster", which is a subclass of "Material physical anatomical entity". On the other hand, "Compartment subdivision" is an instance of Anatomical_space, which is a subclass of "Non-material physical anatomical. The two definitions are inconsistent because these two classes are disjoint.

4. Discussion and perspectives

As far as we know, the NCI Thesaurus was one of the largest file in Protégé OWL so far. It contains "only" 53,000 frames, including 34,000 classes and 9,000 conditions. But it is much smaller and less complex than the FMA OWL. Moreover, there are no defined class, hasValue or allValuesFrom restrictions, nor class union, specified, whereas all these features occur in the FMA OWL file. Thus the FMA in OWL offers a real challenge for description logics systems. This conversion showed that, with the current state of the art of DL inference technology, it might generate inference problems that are hard to solve in terms of time and space. Indeed, the main problem was a computational issue. But finally converting a large part of FMA from Protégé into OWL DL proved possible and demonstrated the applicability and usefulness of OWL DL techniques for very large ontologies. Racer revealed inconsistencies in the original FMA, which might have been missed otherwise. Some classes were also reclassified. These benefits prefigure the potential improvements that might result from "OWLlizing" the FMA.

The migration presented here is not complete yet and will still be improved. At this step, the objective was to stick as much as possible to the FMA representation in Protégé, in order to evaluate the original properties of the FMA. We suggest adding disjointness axioms between primitive classes (siblings), closure axioms, and enriching the FMA representation with "semantically" correct class definition(s) (equivalent class expression combining several properties e.g. parts and boundaries):

– **Property existential restrictions and closure restrictions**. We defined existential restrictions (`owl:someValueFrom`) instead of metaclass structural own slots values. The two reasons behind this choice are: On the one hand (1) the assumption (from [2] and [3]) that in Protégé FMA, if a class A has a slot p filled with values V_1, V_2 …V_i… V_n (e.g., constitutional part), it means that for every individual of A, p has exactly one value of each class V_i. On the other hand, we were faced to the expressiveness limitation of OWL DL, which does not support qualified cardinality restrictions. However, existential restrictions do not offer equivalent flexibility. For example "has part someValuesFrom V_1 and has part someValueFrom V_2, etc" is not equivalent to "has part exactly one V_1 and exactly one V_2, etc.". First, "someValuesFrom" does not prevent from having another part V_3, which is not V_1 or V_2, nor to have several parts of the same V_i. A classical solution to the first point, also point, also called "closure axioms" [4], is to introduce universal property restrictions by suited `owl:allValuesFrom` axioms. But this is not satisfactory either for two reasons: i) Computing the closure is not obvious. If the property is transitive, e.g. part of, it is necessary to recursively compute its transitive closure and also the union of all the parts of all the subclasses, and to add a value restriction on the property stating that the only possible values must belong to their union. ii) Closure axioms do not prevent from having several individuals of the same class V_i (e.g. two parts right frontal lobe for a right hemisphere!). Although adding closure axioms might be a partial improvement, an OWL extension with qualified cardinality restrictions would be more desirable.

– **Disjointness axioms**. At that time, the inconsistencies reported (§3) are mainly based on opposite values of a given boolean datatype property or on the disjointness of classes due to it. But the same holds for disjoint classes in general. Ideally, a classification satisfies the so-called "jointly exhaustive and pairwise disjoint" rule. If the FMA complies with this rule, however it is not explicit in the FMA Protégé. Explicit disjointness axioms should be asserted between relevant siblings (for primitive classes). Introducing such disjointness axioms will most probably lead to identifying more inconsistencies. For example, OWL domains and ranges are global axioms used in reasoning. If a property p has domain A and p is used for B, it will be inferred that class B must be a subclass of A. This can force classes A and B to be reclassified, and if A and B are disjoint, the reasoner will identify an inconsistency (the same reasoning holds for range combined with disjointness). Situations similar to §3 can result in reclassification or in the identification of inconsistencies from domain and range object properties or from metaclass and class definition, when A and B are disjoint.

– **Equivalent class definition**. Four options can be considered for specifying the "defined" classes. 1) Each concept has a single class definition, expressing the intersection of all the property restrictions asserted in Protégé for that concept by its own slots and attributed relations values. 2) Each "defined" concept has a set of several equivalent class definitions (necessary & sufficient conditions). "Defined" concepts would then be specified by several *class equivalence* axioms of the form CN ≡ Expression_1 ≡ …Expression_i…≡ Expression_n, where CN is the concept name and Expression_i are complex expressions (OWL class description), interpreted as a necessary and sufficient conditions for an individual to be an instance of the class CN. 3) Each concept has one preferred definition, the other conditions being simply necessary. 4) No "defined" classes are a priori selected: since the FMA is a "shared reference ontology", only primitive classes are provided (i.e. all conditions are necessary), and the most usual class definitions are proposed as optional. The responsibility of refining this OWL DL ontology by additional axioms selected from the predefined expressions (as class equivalent definition) is left to the users, as required by their applications

Other changes may also be introduced in the future, without giving up the FMA underlying ontological and modeling principles, for example about attributed relations or new classes defined in agreement with the FMA authors.

In conclusion, although not fully completed yet, converting the whole FMA into OWL DL proved possible, with most features of the original FMA being preserved. After narrowing progressively time and space issues and some optimizations, Racer could successfully be used. The present migration demonstrates the benefits of DL reasoning services, and prefigures additional improvements for the FMA. Enriching the representation with "semantically" correct class definition(s) that would allow identifying uniquely an anatomical entity is a promising perspective. A main strength of our approach is its flexibility: different conversion rules can be selected and different class definitions provided, depending on the application. A compromise for large sharable domain ontologies such as the FMA, might be to represent them in OWL DL, but with only primitive classes, and to refine them by additional axioms specifying more precise ontologies customized to each application.

5. Acknowledgments

This work was done when C. Golbreich and S. Zhang were visiting scholars at the Lister Hill National Center for Biomedical Communications, National Library of Medicine, National Institutes Department of Health and Human Services. We also thank C. Rosse and the FMA team for their support, V. Haarslev and R. Möller for their advice and help on Racer and DLs.

6. References

1. OWL Web Ontology Language 1.0 Reference. Mike Dean, Dan Connolly, Frank van Harmelen, James Hendler, Ian Horrocks, Deborah L. McGuinness, Peter F. Patel-Schneider, and Lynn Andrea Stein. W3C Working Draft 12 November 2002 http://www.w3.org/TR/owl-ref/. OWL Web Ontology Overview :W3C Working Draft 4 March 2003
2. Rosse C, Mejino JL, Jr. A reference ontology for biomedical informatics: the Foundational Model of Anatomy. *J Biomedical Informatics* 2003; 36(6):478-500.
3. Noy N, Musen MA, Mejino JL and Rosse C. Pushing the Envelope: Challenges in a Frame-Based Representation of Human Anatomy. *Data and Knowledge Engineering Journal*; 48(3):335-359. 2002.
4. Rector A, Drummond N, Horridge M, Rogers J, Knublauch H, Stevens R, Wang H, Wroe C. OWL Pizzas: Practical Experience in Teaching OWL-DL: Common Errors and Common Patterns. *European Conference on Knowledge Acquisition (EKAW-2004)*. 63-81. 2004.
5. Golbreich C., Zhang S., Bodenreider O. From Frame Based Ontologies to OWL: the FMA migration (ISWC 2005 submitted)
6. Dameron O, Rubin DL and Musen AM. Challenges in Converting Frame-Based Ontology into OWL: the Foundational Model of Anatomy Case-Study. Proc. AMIA Annual Symposium 2005 (submitted).