

Construction of the dialysis and transplantation ontology, advantages, limits, and questions about Protégé OWL

Christine Golbreich, Sandrine Mercier

Laboratoire d'Informatique Médicale,
Faculté de Médecine, University Rennes 1
Av du Pr. Léon Bernard, 35043 Rennes France

Christine.Golbreich@univ-rennes1.fr sandrine.mercier@tremplin-utc.net

Abstract 16 May 2004

Abstract. The project under progress concerns the development of a Local as View mediator for querying heterogeneous sources of end stage organ failure and transplantation information. In that context, we have chosen to use Protégé OWL for building an OWL ontology of the dialysis and transplantation domain. This paper is a short overview about some points related to its construction. This concrete experience clearly enlightened some advantages of OWL DL against some more traditional frame based approach, for instance for dealing with multiple viewpoints. But it also showed some limits of OWL DL for the construction of biomedical ontologies, for instance for expressing the transfer of properties from parts to wholes. It also aroused several modeling questions, for instance about necessary vs necessary & sufficient conditions etc., and about the Protégé editor.

1 Benefits of OWL DL for dealing with multiple hierarchies

Multiple viewpoints is an old and recurrent problem in biomedicine, which often originates inconsistencies in biomedical terminologies or ontologies. Previous experiences conducted at LIM, like the terminological server of the French National Agency for Transplantation [9], or BioMeKe, a system using GeneOntology™ (GO) and the UMLS® to annotate genes, clearly highlighted several difficulties due to multiple viewpoints. For instance, GO top categories Molecular Function, Biological Process, and Cellular Component being structured according to different viewpoints, a term may be found both as sibling and child of another term e.g. ‘Metal ion transporter activity’ is a sibling of ‘Cation transporter activity’, while in another subtree it is a child of ‘Cation homeostasis’. The EfG terminology server has been built to integrate several existing terminologies, e.g. the French Thesaurus of Nephrology and ICD, being driven by a nosological viewpoint. In that server, the diseases are described according a frame-like view, and organized according different dimensions, e.g. “diseases classified by location”, “diseases classified by evolution”, “diseases classified by finding” etc. But being built pragmatically and manually, and implemented in a language without multiple inheritance management, the different subtrees exhibit inconsistencies redundancies, or misclassifications, mainly issued from the multiple hierarchies. Our present approach aiming at constructing an OWL formal ontology for dialysis and transplantation with Protégé OWL [7] is really different and more satisfying. Indeed, the different classes are first defined, *without boring about multiple hierarchies*. Next, based on the classes and properties logical definition and inclusion assertions, *the multiple hierarchies classification is automatically computed*, by a Description Logics reasoner. Our methodology of construction includes two main steps: a first “modeling” step based on a frame view, and a “formal representation” step based on a Description Logics logical view.

1.1 Classes, properties, hierarchies

At a first step, we adopted a rather classical frame-based modeling approach, describing a class as a frame, with a name, its superclass(es), a list of slots and restrictions specifying the range of their fillers. Thus, we used Protégé to explicitly define the names of the different classes, their properties, and hierarchies. The ontology includes a “main” hierarchy of classes whose root is the class “Patient”, which gathers several information about patient disease, treatment, location, etc. and secondary¹ disjoint hierarchies for specific domains like Disease, Treatment etc. The class Patient is related to the other domain classes by its properties hasDisease, hasLocation, hasTreatment, hasHealthCareUnit etc. Each secondary domain is then recursively defined in the same way, for instance the class Disease is related by its properties diseaseFinding, diseaseLocation, diseaseEtiology, diseaseEvolution, diseaseAssociatedLesion etc. to other subdomains Finding, AnatomicLocation, BiologicalProcess etc. Each class hierarchy is defined according multiple dimensions, corresponding to the relevant viewpoints of the domain, thanks to its different properties. For instance Disease is subdivided into several subclasses Acute vs. Chronic, Organic vs Functional, Primary vs Secondary, Infectious, Genetic, Metabolic Disease etc. distinguished according the values restrictions of its properties diseaseEvolution, diseaseAssociatedLesion, etc. Then, these subclasses are further refined by crossing several superclasses, for instance the class AcuteOrganicGlomerulopathy is defined as a subclass of AcuteDisease, OrganicDisease, and Glomerulonephroathy etc. This process quickly leads to a very big graph, which is quite impossible to be managed by hand (Figure 2).

¹ Main and secondary depends of the focus of the biomedical application, the main ontology might be the Anatomy hierarchy or Disease etc. for other applications

1.2 Primitive vs defined classes, necessary vs necessary & sufficient conditions

At a second step, we used the Protégé OWL plugin to specify inclusions or definitions, thanks to “terminological” axioms. Considering that the classes at the top of the hierarchies are general “abstract” classes (e.g. Disease, Etiology, DiagnosisMethod, Treatment, Lesion etc. cf. Figure 1), we have chosen to define them as “primitive” concepts by *class inclusion* axioms of the form $CN \subset \text{ClassExpression}$ where CN is a class name and ClassExpression is a complex expression complying to the OWL DL syntax, which can be interpreted as a necessary condition for an individual to be an instance of the subclass CN. For instance, the class Disease is defined as a primitive concept by inclusion axioms like $\text{Disease} \subset \forall \text{diseaseAssociatedLesion Lesion}$, $\text{Disease} \subset \forall \text{diseaseFinding Finding}$ etc. (Figure 2). More concrete classes that must be classifiable or about which queries may be posed, are specified as “defined” concepts by *class equivalence* axioms of the form $CN \equiv \text{ClassExpression}$ where CN is a class name and ClassExpression is a complex expression, which can be interpreted as a necessary and sufficient condition for an individual to be an instance of the class. For instance, specific types of diseases, infectious, organic diseases, glomerulonephropathies, etc. are defined as “defined” concepts by class definitions like $\text{InfectiousDisease} \equiv \text{Disease} \wedge \exists \text{diseaseCausedBy InfectiousAgent}$, $\text{OrganicDisease} \equiv \text{Disease} \wedge \exists \text{diseaseAssociatedLesion Lesion}$, $\text{Glomerulonephropathy} \equiv \text{Nephropathy} \wedge \exists \text{diseaseLocation KidneyGlomerulusLocation}$, $\text{AcuteOrganicGlomerulonephropathy} \equiv \text{AcuteDisease} \wedge \text{OrganicDisease} \wedge \text{Glomerulonephropathy}$, etc. We also used the OWL DL possibility of asserting additional necessary condition on defined classes by inclusion axioms. For instance, the defined concept $\text{DialysedPatient} \equiv \text{Patient} \wedge \exists \text{hasTreatment Dialysis}$ specifies a necessary&sufficient condition for an individual to belong to that class, but other inclusion axioms enable to assert additional constraints on a class, for instance $\text{DialysedPatient} \subset \exists \text{hasDialysisTreatmentMethod}$ expresses that a dialysed patient has at least some dialysis method (e.g. hemodialysis) (Figure 1).

1.3 Consistency checking and classification based on DL

We used Racer [6] to find out hidden dependencies, inconsistencies, and to compute the overall multiple hierarchies classification, from the class and properties logical definitions and inclusions. We incrementally fixed them and revised the ontology until it was proved to be globally consistent. Such reasoning services were really indispensable for the construction, and validation of the ontology. This experience clearly demonstrates the benefits of automatic reasoning techniques and tools based on DL for biomedical ontologies, which most often involve different viewpoints, are huge and continually evolving.

2 A provisory solution for dealing with properties propagation within OWL-DL

OWL DL supports powerful automatic reasoning, but it also arouses some difficulties. Needs to express the propagation of properties, in particular the transfer of properties from parts to wholes, or properties dependencies as exhibited for instance in the brain-cortex ontology [2, 4] is another well identified problem for biomedical ontologies. We were inevitably faced to this recurrent problem due to DL intrinsic expressiveness limitations for representing “deductive” knowledge, when representing the transplantation and dialysis ontology in OWL. Indeed for instance, we had to represent for any patient, any dialysed or transplanted patient etc., the patient disease ore initial disease, his geographical location, which for instance plays an important part for graft allocation. Thus, we needed a solution allowing inheritance of person geographical locations and disease anatomical locations: a patient that lives in a point situated in a city of a region is located in the region, similarly, a disease, a lesion etc. located in some point of a glomerulus is situated in a kidney. Rules are the right way to deal with this, but since SWRL [8] is not yet available, nor inferentially integrated to OWL [5], we were obliged to find some other provisory solution. We distinguished two concepts, the AnatomicalLocation class, representing the set of all possible spatial anatomical locations organized in a subsumption hierarchy according a spatial viewpoint, and the AnatomicalConcept class representing the set of all anatomical parts according an anatomical viewpoint. Since AnatomicalLocation (resp. Geographical Location) classes are defined as a set of spatial elements, it is licit to assert that for each spatial location x if $x \in A$ et $A \subset B$ then $x \in B$. Thus, making this distinction, it is “semantically” correct to organize the AnatomicalLocation, or GeographicalLocation in a subsumption hierarchy. In that way, diseases are inherited as expected, for instance since a GlomerularLocation is a KidneyLocation, a Glomerulonephropathy defined as having a GlomerularLocation is inferred to be a Nephropathy.

3 Combining Frames and DL views

Although Protégé OWL Racer provides a crucial help, it still remains a long and difficult task to build an OWL ontology for a real application. Combining Frames and DL views is really a fine approach, but from our experience it may also entail some surprising unexpected results from inferences due this combination. In fact, after the two steps methodology described above, we were faced with many inconsistencies, which causes were not obvious to discover. Indeed, merging the frame and DL views may sometimes lead to some confusion. For instance, a class frame A defined with slots p_i with domain A and range B_i , is generally considered as a set of necessary conditions for an instance to belong to the class (and not necessary and sufficient conditions). Thus it is expected to be semantically equivalent to subclass axioms expressing the properties restrictions corresponding to the slots: $A \subset \bigcap (\forall p_i B_i)$. But in fact, a Class A with Properties at class p_i with range B_i , is semantically equivalent to $A \& \bigcap (\forall p_i B_i)$. In addition, it should also be noted that in a DL view, a class

C is often defined by a conjunction of the form $C \equiv \bigcap (\forall p_i B_i)$ as soon as its description is considered to be completed. Such a class definition C, combined with the definition of properties p_i having a defined domain C or undefined domain i.e. owl:thing, entails that any class is a subclass of C. Thus, the semantics of Properties, rdfs:range and domain axiom, combined with the usual DL semantics of allValuesFrom constructor, may provide quite surprising effects. Therefore, constrained by the OWL representation, we had to modify our initial representation driven by DL habits, in moving some necessary and sufficient, to necessary conditions, so as to avoid unwanted subsumption or equivalence inferences.

4 Conclusion

In conclusion, Protégé OWL provides a powerful help for the construction of formal ontologies with multiple hierarchies in biomedicine. However, (1) although solutions can be provided within OWL, SWRL rules interoperating with OWL, not only syntactically and semantically, but also *inferentially* are required for chaining ontologies properties, such as the transfer of properties from parts to wholes, and also for reasoning across domains, for Web ontologies data integration, for expressing complex queries upon the Web, for facilitating ontology engineering (acquisition, validation, maintenance), (2) since the ontology classification is mainly based on the classes definitions, on the properties involved in the classes, and the properties range and domain, it would be fine to provide Protégé OWL with “safeguards”, “warnings” or other helps to prevent unwanted inferences, and “assistance” to guide the use of equivalence axioms for defined classes (using necessary & sufficient condition) vs subclass axioms for primitive classes or complementary necessary conditions asserted on a defined class.

5 References

1. Bechhofer, S., van Harmelen, F., Hendler, J., Horrocks, I. McGuinness, L. D., Patel-Schneider, P. F., Stein, L., A.: OWL Web Ontology Language Reference. W3C Working Draft. 2003
2. Dameron O., Burgun A., Morandi X., Gibaud B. Modelling dependencies between relations to insure consistency of a cerebral cortex anatomy knowledge base. Proceedings of Medical Informatics in Europe (2003)
3. Dameron O., Gibaud B., Musen M. Using semantic dependencies for consistency management of an ontology of brain-cortex anatomy, KR-MED 2004.
4. Golbreich, C., Dameron O., Gibaud B., Burgun, A.: Web ontology language requirements w.r.t. expressiveness of taxonomy and axioms in medicine, International Semantic Web Conference (2003)
5. Golbreich, C., Imai, A., Combining SWRL rules and OWL ontologies with Protégé OWL Plugin, Jess, and Racer, 7th International Protégé Conference, Bethesda, 2004.
6. Haarslev V., Möller R. : Description of the RACER System and its Applications. Description Logics 2001
7. Holger K., The Protégé OWL Plugin, 7th International Protégé Conference, Bethesda, 2004.
8. Horrocks, I., Patel-Schneider, P., Harold, B., Tabet, S., Grosz, B., Dean, M.: SWRL: A Semantic Web Rule Language Combining OWL and RuleML. Version 0.5 Nov. 2003
9. Jacquelinet C, Burgun A, Delamarre D, Strang N, Djabbour S, Boutin B, Le Beux P. Developing the ontological foundations of a terminological system for end-stage diseases, organ failure, dialysis and transplantation. Int J Med Inf. 2003 Jul;70(2-3).

Acknowledgments

The authors want to thank Christian Jacquelinet and Cecile Couchaud from The Etablissement Français des Greffes.

6 Examples from the ontology

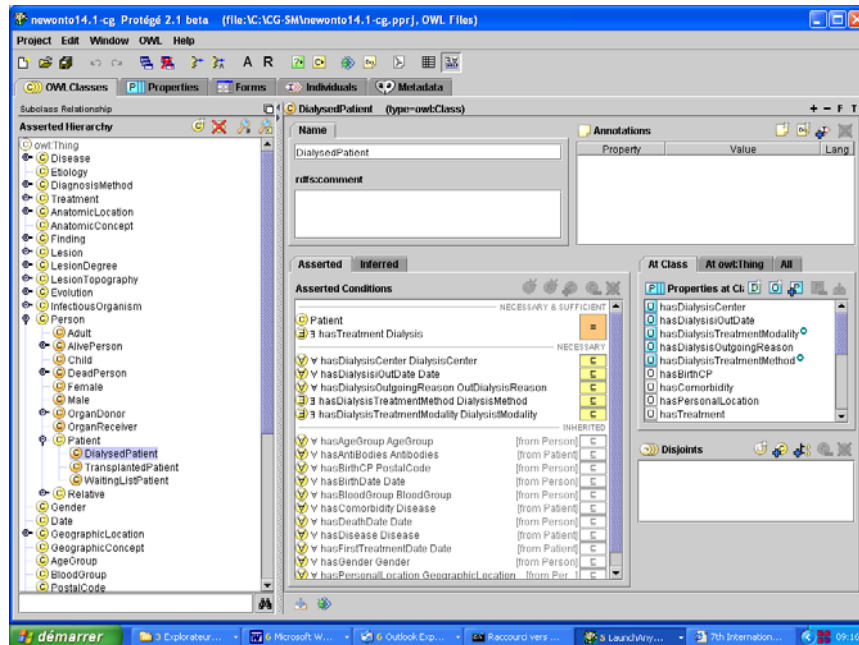
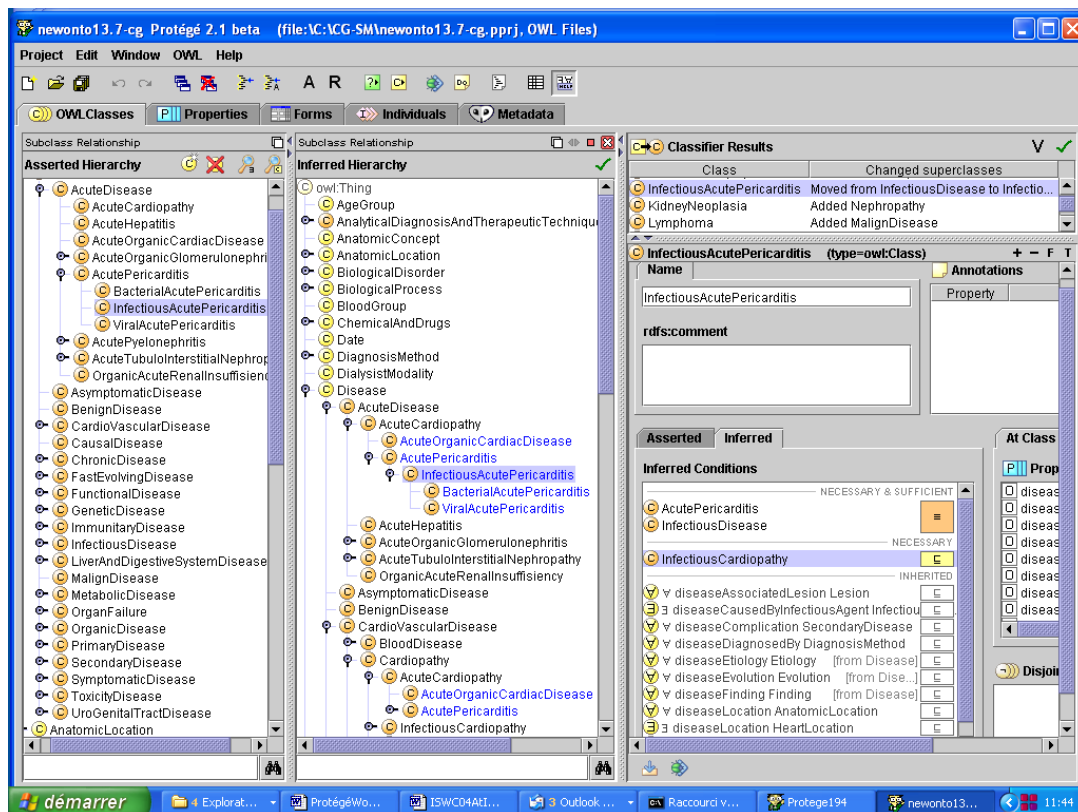


Figure 1 Primitive (yellow) vs Defined classes (orange), Necessary & Sufficient vs Necessary conditions



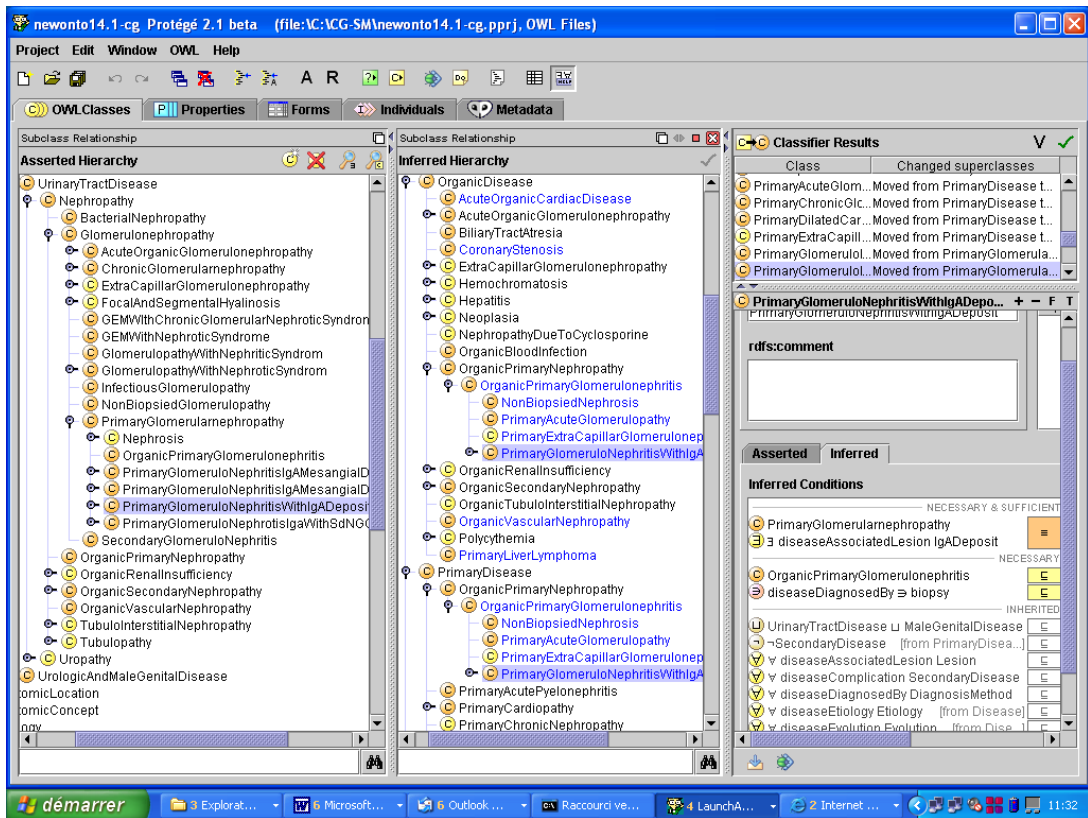


Figure 2 Multiple hierarchies automatic classification