Experiences in Modeling Clinical Examinations in Oral Medicine Using OWL*

Marie Gustafsson[1,2] and Göran Falkman[1]

[1] School of Humanities and Informatics, University of Skövde,
PO Box 408, SE–541 28 Skövde, Sweden
{marie.gustafsson, goran.falkman}@his.se
[2] Computer Science and Engineering, Chalmers University of Technology
SE-412 96 Göteborg, Sweden

Abstract. This article describes the modeling of clinical examinations in oral medicine using OWL. Based on experiences from our previous work and knowledge model, requirements for an ontology for examinations in oral medicine are identified. OWL can be used to address most, but not all, of the requirements. We found a lack of guidance for several design choices and for development of OWL ontologies at different levels of sophistication. However, using OWL gives us the ability to come back and refine the knowledge model after initial deployment.

1 Introduction

Basing clinical decisions on finding, evaluating, and using the latest research results is an essential premise of evidence-based medicine (EBM). A crucial part of the practice of EBM is the integration of the expertise of the individual clinicians with the best clinical evidence obtainable from external sources [1]. The MedView project [2] has aimed to provide IT-support for evidence-based oral medicine by equipping the clinicians with a set of software tools, assisting in the various processes of EBM, and by providing a formal knowledge model on which to base these tools. However, as this model is only used within MedView, it is difficult to reuse external knowledge sources and to share the data collected by MedView tools with others. There is also a need to expand the current model and to reexamine how to best conceptualize examination data in oral medicine.

We have investigated using OWL to model clinical examinations in oral medicine, taking the previous work of MedView as a starting point. The knowledge model of MedView includes the representation of examination templates describing the pattern from which the individual records are created, value lists, aggregates of values, and individual examination records. Within MedView, the separation of information models for patient records and terminological knowledge has been less distinct, which has affected this remodeling work.

* The work presented in this paper was supported by the Swedish Agency for Innovation Systems.

For both information models and medical terminologies, there has been extensive international work, to the extent that it is difficult for a small and previously uninitiated group to make an informed choice among the models and terminologies available. This, coupled with earlier decisions to investigate the use of OWL and the somewhat daunting task of integration the content of the local knowledge model with an external terminology, led us to not base our work on any of these larger initiatives. Also, in considering efforts to standardize nomenclature in dentistry, the clinicians we collaborate with found them lacking. In the dental domain, the America Dental Association provides both a Current Dental Terminology (CTD), which is updated regularly but limited to treatments and procedures, as well as a Systemized Nomenclature of Dentistry (SNODENT), which is an effort to create a comprehensive dental vocabulary. However, it has been found that it needs improvements in content, quality of coding, and quality of ontological structure [3].

Parts of the MedView applications have been adapted to handling the new OWL and RDF representations, which is also used in the SOMWeb (Swedish Oral Medicine Web) online community. This online community serves as support for the discussion of interesting and difficult cases in oral medicine among geographically dispersed clinics in Sweden [4]. In this article we describe the requirements for ontologies for oral medicine, the design and development of such ontologies, and our experiences in using OWL. While the perspective is that of remodeling, we believe that our experiences and requirements are relevant for using OWL and clinical modeling in general.

2 Requirements for an Ontology for Oral Medicine

In deciding to revise the knowledge model of MedView, we collected requirements for an ontology for oral medicine. These are based on 10 years of experience with the MedView system and interviews with domain experts and developers, and are described in [5]. To summarize:

- We need the possibility and ability to utilize external sources of knowledge.
- The relation between the conceptual models of fundamental clinical concepts in use, e.g., examination templates, lists of approved values, and groups of related terms, and their corresponding concrete entities must be examined.
- Relations and interactions between different entities of the ontology must be captured, e.g., that a certain answer to a specific question in a given examination template triggers another question.
- A stronger typing of elements is needed. We must be able to enforce that a certain term e.g., only has numeric values, or a certain enumerated domain.
- We need to be able to capture different kinds of meta-data, e.g., the creator and purpose of a specific examination template.
- The localization of data has to be addressed: How to provide different language-based versions of the defined concepts, definitions and terms?
- We need to be able to provide different views of the underlying data, to be utilized for e.g., information visualization and intelligent user interfaces; e.g., a patient or temporal oriented view.

3 Design and Development of the Ontologies

The design of the SOMWeb ontologies takes the MedView knowledge representation and content as a starting point, which includes (1) examination templates describing the pattern from which the individual records are created and which are used in constructing graphic input forms, (2) value lists from which values can be chosen when filling out these forms, (3) aggregates of values created and used when analyzing data from the examination records, and (4) individual examination records. We refer to the original, approach as the MedView representation, and the new OWL- and RDF-based representation as the SOMWeb representation.

3.1 Designing the Examination Template Ontologies

An examination template describes what should be included in an examination record. In MedView, examination templates were stored as XML documents, and there is a DTD to describe general features of such templates. In SOMWeb, each examination template is stored in a separate OWL file, and one OWL file defines common classes and properties. In MedView, an examination template is made up of several categories (subsections of the examination), each with a number of inputs (properties). An examination template OWL file contains definitions of the categories that can or need to be included in an examination constructed from that template. Examples of subclasses of `ExaminationCategory` (defined in the common file) are `PatientData` and `GeneralAnamnesis` (an anamnesis is a patient's account of a health history). In each examination template we also describe inputs (or properties) of the template. Each property also has properties pertaining to description and instruction. Figure 1 shows the structure of an example `ExaminationCategory`, `GeneralAnamnesis`, and how it relates to the common description and the value list ontologies.

To create the templates, a translator program was written which uses existing MedView Java classes for reading templates and value lists. Based on the proposed structure and design decisions described below, we create the appropriate OWL constructs using the Jena API.[3]

Design Choices

Domain and range For each input we need to define what `ExaminationCategory` they can belong to and what types of values they can take. This can be done using `rdfs:domain` and `rdfs:range` or using restrictions such as `owl:allValuesFrom` and `owl:someValuesFrom`. We initially expected that using domain and range would provide some sort of constraint checking, while they are rather used to infer additional information about individuals. Another consequence of using `domain` and `range` from RDFS is that they apply globally to the property, while `allValuesFrom` and `someValuesFrom` restrictions are local to the class for which they apply. This leads us to not use `domain` and `range`.

[3] `http://jena.sourceforge.net/`

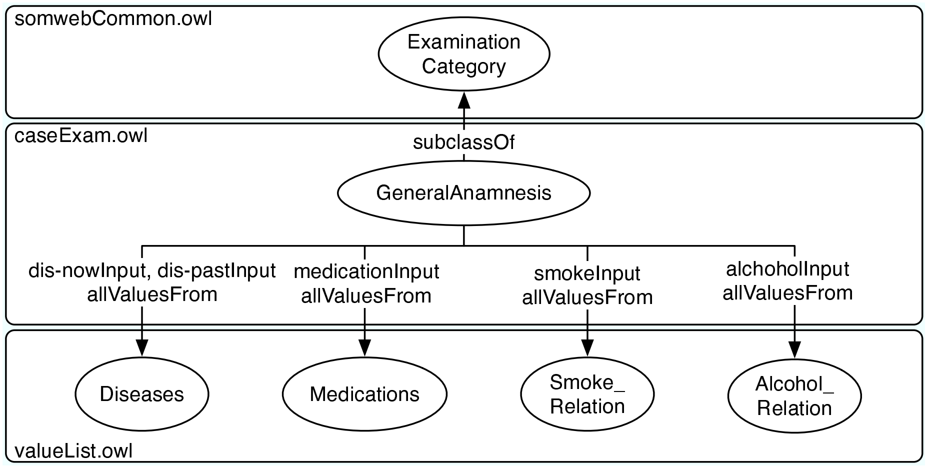

Fig. 1. Structure of an `ExaminationCategory` subclass, `GeneralAnamnesis`. Boxes indicate which OWL files the classes and properties are in. The properties used by `GeneralAnamnesis` are in the caseExam.owl file, which also contains restrictions, e.g., an instance of `GeneralAnamnesis` has only instances of the class `Diseases` as values of the property `dis-nowInput`.

Representing order An examination template has an implicit order. A general anamnesis normally comes before a diagnosis. Whether or not this order is intrinsic to the examination, or is part of the *presentation* of the examination remains an open question. However, OWL and RDF, being graphs, have no order. We use `rdf:list` to represent the order of categories and inputs. However, using `rdf:list` brings us into OWL Full. As we currently don't reason over the order of `ExaminationCategories`, it is less of a problem. However, one can think of cases where we will need to reason over the order (e.g., in dependency relations between different categories). The W3C Working Group Note on N-ary relations [6] has suggestions for representing lists as N-ary relations, which could be used to avoid `rdf:list`. See also recent work on representing lists in OWL [7].

MedView's Question Inputs MedView has an input category called 'question'. This is used, e.g., for specifying how long a patient has had a certain problem (value + time unit), or smoking habits (value + tobacco category + time unit). The MedView representation of this was unsatisfactory, as there was for example no relation between the time units. For dealing with 'questions' we use the W3C OWL Best Practices Proposal for Representing N-ary Relations [6].

3.2 Designing the Value List Ontology

The MedView value lists, stored in text files, contain term names and the possible values the terms can take. In SOMWeb, all terms are represented as OWL classes, with their values as instances. The term Allergy becomes the class `Allergy`, and the values of the Allergy term are instances of this class. Since the structure of the terms in MedView is flat, no subclasses are created. For each term and value, we can add metadata about the creator. Some of the values in MedView contain extra information in the value name, e.g., the diagnosis values can have an ICD code concatenated to them.In SOMWeb, such extra information is separated from the name, and provided in a `somweb:icdCode` property.

In moving to OWL, we initially attempted to categorize and clean up the instances using Protégé, so that e.g., `Allergy` would have a subclass `PollenAllergy`, of which `BirchPollenAllergy` would be an instance. However, this approach was time-consuming, and should be carried out by the clinicians (or from reuse). Instead, the MedView value lists are converted programatically, though this means that any irregularities of the lists are included in the SOMWeb version.

Design Choices

Using Instances We chose to use instances, rather than classes, for values, which can be problematic for several reasons. Consider, e.g., a property `hasAllergy` pointing to the instance of the class `Allergy` representing a dog allergy. If we use this instance for two different people, are we somehow saying that they have the same allergy against dogs, in the sense that the chemical reactions in their bodies resulting from an exposure to dogs are identical? A solution could be to have each kind of allergy as a subclass of the class `Allergy`, and then creating instances of these for each patient's allergy. Such an approach would be similar to the referent tracking system, suggested by Ceusters and Smith [8].

Naming In deciding what URI to use for the values, the most straightforward is perhaps to create one based on the current term value, which is in Swedish. If the name of the Swedish value contains symbols not allowed in URIs, such as Swedish letters, the URI is made by replacing these (International Resource Identifiers were not considered, though we may switch in the future). The `rdfs:label` is used for the original name. Initially we have Swedish labels for all values; labels for English can be added later. Should we at some later time want to switch to URIs using English, we can state that the old URI using Swedish refers to the same thing as the new one using English.

Before deciding to base the URIs on the Swedish values, we considered using English names or an identifier scheme not containing the value name. Using English names was decided against, as many of the terms we need are not in a common dictionary, and a manual translation would be time-consuming. Having URIs with an identifier scheme not based on the value name and having the Swedish name only in the label, was also considered. However, with this approach the RDF-file describing the examination is less readable and contains less information as you need to consult the description of the value to find its Swedish name. Related to the question of URIs and naming are Life Science

Identifiers (LSID),[4] designed to be location-independent, stable, and resolvable identifiers for entities on the Semantic Web for the life sciences [9]. These were found to be beyond the scope of this work.

3.3 Representing Value Aggregates and Examinations

In MedView, value aggregates are created and used when analyzing collected data. For example, one may want to group allergies into different categories to investigate relations between these categories and certain mucous membrane changes in the mouth. In MedView, aggregates were seen as an abstraction of values, which are in themselves values. In the SOMWeb representation, this is mainly done by subclassing the values in the value list ontology, and making the appropriate individual values instances of this subclass.

Each individual examination is stored as a separate RDF file. We had initially thought that we would be able to validate this against the corresponding OWL examination template. However, given the open world assumption and the no unique names assumption, validation is not automatically available without making assumptions. That is, we cannot take a RDF description of an examination instance and check that it fulfills all the requirements of the corresponding OWL examination template description without, e.g., using a tool such as Eyeball,[5] which makes closed world assumptions for checking RDF models for problems.

4 Results in Relation to the Requirements

Utilization of External Sources We found it difficult to locate relevant ontologies to reuse. Part of this was lack of appropriate services for finding ontologies of use,[6] and partly due to a lack of ontologies fitting our needs. One problem may have been naming; there could exist ontologies that fit our needs, where important concepts are named differently. Further, in the case of the SOMWeb community, the users want to have a high level of control over what kind of data they collect. How can this need be balanced with the possibilities of interoperability which may be gained from reuse? The problems of reuse are widely known (e.g., [10]), and while our problems were partly from not finding relevant ontologies in OWL to reuse, it might not be alleviated by more ontologies being published. As noted by [11], with more available ontologies, more effort is required to evaluate ontologies, and there are few available objective measures to determine ontology quality.

Relations Between Conceptual Models of Clinical Concepts The remodeling work has value in itself, as it meant that many of the structures of the knowledge model had to be thought through further. Indeed, one reason for developing an ontology is to elaborate on a common conceptual model [12].

[4] `http://lsid.sourceforge.net/`
[5] `http://jena.sourceforge.net/Eyeball/`
[6] Though Swoogle (`http://swoogle.umbc.edu/`) is a good resource for this.

Capturing Interactions Between Different Parts of the Ontology This requirement, where e.g., one examination question triggers another, has not been studied within this work. It would probably be appropriate to use the Semantic Web Rule Language (SWRL)[7] in conjunction with OWL, if one wants to stay within the sphere of Semantic Web technologies.

Stronger Typing of Elements Here we get the benefit of working within a larger framework, where structures for this is already in place. However, the consequences of the open world assumption on validation means that we do not get validation in the way that we had expected.

Capturing Examination Template Meta-data Using OWL provides good possibilities for capturing meta-data regarding creators and purpose of different examination templates.

Localization of Data It is possible to provide different language-based versions of the parts of the examination template and the values that can be used, by utilizing the `xml:lang` of `rdfs:label`. However, such translations have not yet been provided.

Differentiating between different views of the underlying data While it has not been rigorously addressed in this work, it seems feasible that the RDF graph is a good starting point, as we can 'pull' the graph from different nodes, making them starting points, thereby providing different perspectives.

Representing Data Ranges Though not listed as a requirement, the possibility to define data ranges would have been useful for e.g., representing values on a scale from 1 to 10. There is currently no support for this in OWL, which was not something that we had anticipated when we considered using OWL. However, in the proposed OWL 1.1, support for value ranges is included.

5 Our Experiences in Using OWL

In working with OWL, possibilities for knowledge sharing are increased and we get the option of using general tools developed by others. In using a Semantic Web technology, we also get the prospect of being part of a distributed knowledge-base. However, using OWL has not been without issues. The remodeling of the MedView knowledge model has taken place over the course of approximately two years (fall of 2004 to fall of 2006). At the beginning of this process, OWL was a fairly recent recommendation. Though ontology research has been an active research field since the 90's and the precursors of OWL provided initial expertise, we still experienced a lack of guidance in many design choices.

As the MedView knowledge model includes both terminological knowledge and models for patient records, our remodeling work asks if OWL can handle both. The use of OWL to represent the terminological knowledge was less of an issue than using OWL to represent templates for patient records. Initially, we had expected to be able to specify a schema, which would be used to validate examination records, in addition to being able to give more elaborate definitions of what to include in an examination. As we found out, this kind of validation is not

[7] `http://www.w3.org/Submission/SWRL/`

available without making several assumptions, such as a closed world assumption. This could be made clearer to newcomers to OWL. Further, we see a need to be able to provide schema functionality within the Semantic Web framework. New restrictions, such as `makeSenseForTypeKnownToBe`, `canOnlyBeKnownType`, and `mustBeAtLeastOneKnown`, as suggested by [13], could be a step in this direction.

Many of the problems we experienced relate to whether or not there is one, correct way to use OWL, and how important this is to follow. When creating an ontology it is hard to decide when the ontology is good enough to use. There is a compulsion to create the 'perfect' ontology. Combined with lack of support for some design choices, it becomes difficult to determine when the ontology is ready to be deployed. Knublauch et al. [14] see a major benefit in OWL's breadth, in that it offers a "migration route from entry level, hand-crafted taxonomies of terms, to well defined, normalized ontologies capable of supporting reasoning." We concur with this, and there is a great need for support for such a route, where the developer is given guidelines for what is the appropriate way of representing these 'entry-level' ontologies, so that they are coherent and so that the path to more well-defined ontologies is clear or even possible. We also found it uncertain how much DL expertise is needed for successful ontology development in OWL, which is also stated in an assessment of RDF and OWL modeling [15].

However, with OWL, much of the more practical guidance is found in mailing lists, wikis, and W3C working group notes. While these are public, their existence, validity, and use might not be obvious to the novice developer. In experience reports of OWL, e.g., [14, 16], many of the issues that we have faced come up, such as the open world assumption, the no unique names assumption, validation, value ranges, ontology reuse, imports, using instances or classes, OWL's XML syntax, use of domain and range, OWL's sublanguages, and problems for developers new to OWL.

6 Discussion

The development of the SOMWeb ontologies was based on the knowledge model of MedView. Taking a previous representation as a starting point for ontology creation simplifies the development, as much knowledge acquisition has already been done, and consideration has been given to the structuring of this knowledge. At the same time, this means that the ontology is developed within certain constraints, and cannot adhere entirely to the new representation paradigm.

MedView aims to provide the end users with means to themselves formulate what to include in examination templates and value lists. While the value lists are part of an ambition to harmonize the values used in examination records, the users have been able to add values as they see fit. This control can be a motivation for the clinician by easing the filling out of the examination forms, but as a result, the value lists contain duplicates (different spellings and different names for the same thing), and values mistakenly entered. The clinicians need tools to carry out 'clean up' and structuring of the lists. This tool has to be presented in a such way that users see a value in structuring, as it might not seem a natural part of their work tasks. Another way to get more structured value lists would be to reuse external resources. For example, SOMWeb could reuse an externally developed allergens ontology or a standardized nomenclature, such as those mentioned in Sec. 1. In SOMWeb, ICD codes are sometimes added as properties of instances of the `Diagnosis` class. Preferably, ICD as a whole could have been reused, but it is not available in OWL. Our approach has the advantage of only providing the subset of values that our users are interested in. Reusing classifications presents difficulties in that some medical entities will seem insufficiently detailed to some specialists, and too complex to use for others. Further, it can conflict with requirements of end-user control.

7 Conclusions and Future Work

This report describes how the W3C recommendations OWL and RDF can be used for representing clinical knowledge in oral medicine. From the limitations of MedView's original knowledge model, requirements for a new model were identified. Of these requirements, OWL and RDF have proven useful for capturing meta-data, different language versions, and providing a stronger typing of elements. While using these recommendations give better opportunities for knowledge reuse in theory, it was found that in practice it was difficult to find ontologies to reuse, which were at an appropriate level of conceptualization and were available in OWL. Using OWL has given us access to a potentially beneficial array of externally developed tools, the ability to come back and refine the knowledge model after initial deployment, as well as the possibility of, post development and use, aligning the developed ontology with that of potential collaborators.

There has yet to be a rigorous evaluation of the developed ontologies, either from a technical perspective or from a user perspective. However, they have gone through iterations of evaluation and adjustments from use in the SOMWeb community. In continuing and expanding on the remodeling of MedView's knowledge model, we need to provide tools for the clinicians to be able to add detail to and refine the ontologies described above.

In using OWL we found that, while there is much useful material available, there is still a lack of support for design decisions and best practice guidelines are still under development. There is a lack of recommendations for what is necessary in developing ontologies with different levels of sophistication and guiding materials at an intermediary level, when the developer has moved beyond the introductory materials but is not yet prepared for the more advanced nuances of OWL. Further, methodologies for ontology development should cater to different levels of ontology sophistication, so that the developer knows what is necessary for reaching an intended result.

References

1. Sackett, D.L., Rosenberg, W.M.C., Gray, J.A.M., Haynes, R.B., Richardson, W.S.: Evidence based medicine: What it is and what it isn't. Brit. Med. J. **312**(7023) (1996) 71–72
2. Jontell, M., Mattsson, U., Torgersson, O.: MedView: An instrument for clinical research and education in oral medicine. Oral Surg. Oral Med. Oral Pathol. Oral Radiol. Endod. **99** (2005) 55–63
3. Goldberg, L.J., Ceusters, W., Eisnerc, J., Smith, B.: The significance of SNODENT. In: Proc. Medical Informatics Europe (MIE-05). (2005) 737–742
4. Gustafsson, M., Lindahl, F., Falkman, G., Torgersson, O.: Enabling an online community for sharing oral medicine cases using Semantic Web technologies. In: Proc. Int. Semantic Web Conf. (ISWC-06). (2006) 820–832
5. Gustafsson, M., Falkman, G.: Representing clinical knowledge in oral medicine using ontologies. In: Proc. Medical Informatics Europe (MIE-05). (2005) 743–748
6. Noy, N., Rector, A.: Defining n-ary relations on the Semantic Web. W3C Working Group Note (2006) Available at: `http://www.w3.org/TR/swbp-n-aryRelations/`.
7. Drummond, N., Rector, A., Stevens, R., Moulton, G., Horridge, M., Wang, H., Seidenberg, J.: Putting OWL in order: Patterns for sequences in OWL. In: Proc. 2nd Workshop on OWL: Experiences and Directions. (2006) Available at: `http://owl-workshop.man.ac.uk/acceptedLong/submission_12.pdf`.
8. Ceusters, W., Smith, B.: Strategies for referent tracking in electronic health records. J. Biomedical Informatics **39** (2006) 362–378
9. Clark, T., Martin, S., Liefield, T.: Globally distributed object identification for biological knowledgebases. Briefings in Bioinformatics **5** (2004) 59–70
10. Paslaru Bontas, E., Mochol, M., Tolksdorf, R.: Case studies on ontology reuse. In: Proc. 5th Int. Conf. on Knowledge Management (I-Know-05). (2005)
11. Noy, N.: Order from chaos. Queue **3**(8) (2005) 42–49
12. Noy, N., McGuinness, D.: Ontology development 101: A guide to creating your first ontology. Stanford Knowledge Systems Laboratory Technical Report (2001) Available at: `http://ksl.stanford.edu/people/dlm/papers/ontology-tutorial-noy-mcguinness.pdf`.
13. Ruttenberg, A., Rees, J., Zucker, J.: What BioPAX communicates and how to extend OWL to help it. In: Proc. 2nd Workshop on OWL: Experiences and Directions. (2006) Available at: `http://owl-workshop.man.ac.uk/acceptedLong/submission_26.pdf`.
14. Knublauch, H., Horridge, M., Musen, M., Rector, A., Stevens, R., Drummond, N., Lord, P., Noy, N., Seidenberg, J., Wang, H.: The Protégé OWL experience. In: Proc. OWL: Experiences and Directions Workshop 2005. (2005) Available at: `http://www.mindswap.org/OWLWorkshop/sub14.pdf`.
15. Reynolds, D., Thompson, C., Mukerji, J., Coleman, D.: An assessment of RDF/OWL modelling. Technical report, Hewlett Packard Labs (2005) Available at: `http://www.hpl.hp.com/techreports/2005/HPL-2005-189.html`.
16. Ruttenberg, A., Rees, J., Luciano, J.: Experience using OWL DL for the exchange of biological pathway information. In: Proc. OWL: Experiences and Directions Workshop 2005. (2005) Available at: `http://www.mindswap.org/OWLWorkshop/sub37.pdf`.