Teaching Good Biomedical Ontology Design

Martin Boeker [1]*, Daniel Schober [1], Djamila Raufie [1], Niels Grewe [2], Johannes Röhl [2], Ludger Jansen [2], and Stefan Schulz [1,3]

[1]Institute of Medical Biometry and Medical Informatics, University Medical Center Freiburg, Germany
[2]Institute of Philosophy, University of Rostock, Germany
[3]Institute for Medical Informatics, Statistics and Documentation, Medical University of Graz, Austria

ABSTRACT

Background: In order to improve ontology quality, tool- and language-related tutorials are not sufficient. Care must be taken to provide optimized curricula for teaching the representational language in the context of a semantically rich upper level ontology. The constraints provided by rigid top and upper level models assure that the ontologies built are not only logically consistent but also adequately represent the domain of discourse and align to explicitly outlined ontological principles. Finally such a curriculum must take into account the pre-existing skills and knowledge of the target audience.

Objective: To develop a well-structured curriculum aligned to the particular requirements of life science professionals, in order to enable them to create logically sound, domain adequate and predicable ontologies using the Web Ontology Language (OWL) in Protégé.

Methods: Content selection for the curriculum was based on the literature, pre-existing tutorials, and a guideline for good ontology development (i.e ontology design enhancing domain adequacy, sustainability and interoperability) that drew on the authors previous experiences with large ontology development projects. Learning objectives were formulated according to a needs assessment of the targeted learners, who were students trained in life sciences with basic knowledge and practical skills in computer science. As instructional format we choose an approach with a high amount of practical exercises. The curriculum was first implemented with 24 Students and 7 lecturers/ tutors over 5 full days. The curriculum was evaluated by gathering the participants feedback via a questionnaire.

Results: Curricular development produced 16 modules of approximately 2 hours each, which covered basic principles of Applied Ontology, description logic syntax and semantics, as well as best design practices outlined in ontology design patterns and variants of the BioTop upper ontology. An opinion survey based on questionnaires indicated that the participants took advantage from the teaching strategies applied, as they indicated good knowledge gain and acknowledged the relevance of the modules. The difficulty was rated slightly lower.

Conclusion: The development of teaching material for principled ontology design and best practices is of crucial importance in order to enhance the quality of biomedical ontologies. Here, we present a curriculum for a week long workshop, leveraging on current educational principles, focusing on interactive hands-on exercises, group interactions, and problem-oriented learning. Whereas evaluation clearly showed the success of this approach, in particular regarding student's satisfaction, the objective measurement of traceable effects on the quality of the generated ontology, although of much higher interest, has just started.

*To whom correspondence should be addressed: martin.boeker@uniklinik-freiburg.de

1 INTRODUCTION

Ontology engineering continues to be an area of major interest within the life sciences, as computer-interpretable domain representations are the only viable option for an efficient and intelligent exploitation of the vast amounts of high-throughput-generated data. Although numerous ontologies are publically available through ontology libraries and access portals, they remain of heterogeneous quality and ontological rigor (Smith *et al.*, 2004; Schulz *et al.*, 2009; Rector *et al.*, 2011; Boeker *et al.*, 2011). Coordination efforts and best practice providers, such as the OBO Foundry (Smith *et al.*, 2007) have recently emerged from the need to assure at least some basic quality with respect to ontological correctness and usability.

While these efforts mainly target already experienced ontologists, only few efforts have recently focused on teaching basic notions of ontology engineering to novices. This imbalance results in a growing number of practitioners being forced to apply particular design requirements and patterns as requested by the aforementioned policy providers, but without even knowing the most basic foundations in the semantics of the representation language used.

Another major obstacle in ontology design is that various scientific and engineering communities sustain different and sometimes even contradictory modeling objectives and paradigms. Although skilled logicians or computer scientists should normally understand the formal semantics of description logics, their approaches to certain modeling tasks are often based on other paradigms, like database technologies or object-oriented programming. Object-oriented programming uses, for example, inheritance principles that differ from the inheritance principle of description logics. While the inheritance principles of modern object oriented programming are based on the substitution principle (Liskov, 1987), inheritance in description logics is based on set theory (Baader *et al.*, 2007). Thus, programmers operating with classes in object-oriented programming as abstract types (signatures or interfaces) which can be instantiated and subtyped might have problems to apply description logics based classes as mere sets in a bottom-up approach. The inheritance/ subtyping concepts of modern programming languages allow the engineer for very powerful top-down driven approaches in the modeling of complex software architectures (with method overriding, method overloading and polymorphic types) that have no correspondences in DL. While a top-down modeling approach in class hierarchies is best-practice in object-oriented software paradigms, it will cause trouble when transferred to the design of an ontology that is based on description logics with a deviant formal semantics. This does, of course, not diminish the usefulness of the respective techniques and skills in their own field of application. We only want to point out that they do not comply with the rigid modeling requirements of description logics.

In the light of the rising demand for of formal representation of, e.g., biomedical knowledge (Ashburner *et al.*, 2000), such limited or even mislead ontology engineering skills can lead to wrong design decisions in biomedical ontology. Finally the philosophical foundations of ontology engineering, as concretized in the recent discipline of Applied Ontology (Munn and Smith, 2008) is still largely ignored by ontology engineers with a computer science background (Mizoguchi and Kozaki, 2009). Explicitly, we do not claim to know and teach the *one and only* best ontology, but we provide a robust ontological backbone on the basis of explicitly outlined design principles rooted in traceable/ decidable logics.

In order to arrive at a sustainable development of sound ontology artifacts, it is necessary that there are ontology developers with (1) a technical background in computer science that will (2) closely cooperate with experts for the domain to be represented, and have (3) a foundational grounding in the principles of Applied Ontology and the logical formalisms underlying ontology description languages. The basic idea of our curriculum is to teach such principles as a series of robust and well-calibrated ontological building blocks, targeting students in life sciences disciplines with some background in computer sciences. It should teach the essential skills to novices to develop sound OWL-DL ontologies based on approved methods in formal ontology.

The need for a curriculum on biomedical ontology

Some educational material on formal ontology engineering for non-computer scientists is publicly available, and widely used for self-training or in tutorials, esp. the Pizza Ontology tutorial (Protégé-Tutorial)[1] based on the Pizza ontology[2] and "Ontology Development 101"[3] or publicly available course material[4]. These tutorials are well structured and found wider acceptance, but suffer from some limitations. They are often written as quick introductions and user guides to the Protégé OWL editor and only partially address more complex practically occurring modeling problems, but rather focus on easy-to-understand examples. They do not teach how the ontologies created should be structured, and which tools lend themselves for sustainable design decisions.

As the recent debate on the scientific value of formal ontology in the life sciences has shown, the development of real life ontologies applying decidable logics is an interdisciplinary and highly complex process, whose inter-dependencies are not yet fully understood (Brochhausen *et al.*, 2011). Typically, four scientific communities with overlapping concerns and tasks will be involved in the development of an Ontology:

1. Experts from one or more domains who provide the insights in *what* should be represented (e.g. medicine or biology),
2. philosophers as experts of the *principles* of ontology (as a philosophical discipline),
3. logicians and mathematicians as experts both of the mathematical *formalisms* used for building ontology axioms and their computational properties; and
4. computer scientists who provide the computational framework for *development and deployment* (e.g. knowledge representation, natural language processing, software engineering).

Several challenges have to be met in the ontology engineering process, starting with the usage of editing and reasoning artifacts with all their idiosyncrasies and computational requirements, followed by putting into practice the collaborative editing of "single-file" or modularly structured ontologies.

The scientific community has made considerable progress in the understanding and development of large ontologies. Besides new efforts in continuing the generation of knowledge, we must understand how to educate students to become ontology engineering experts. It is not acceptable that just a small community is able to understand a topic of high interest, which, in contrast, is often considered rather esoteric by those who are supposed to use the artifacts created. In our view, well-instructed domain experts are needed to build good ontologies, because they are the only ones who know what has to be represented and how, and who will later engage in the dissemination and use of ontology-enabled tools. Domain experts with some computer science background should be educated appropriately to understand the basics of philosophical and logical foundations needed for ontology development and be trained in real-life ontology engineering.

2 METHODS

The development of a specific curriculum for building OWL-DL ontologies is part of a larger project in which the key elements for the design of qualitatively *good* ontologies are defined (GoodOD: Good Ontology Design). In this regard, the curriculum objectives and contents mirror major parts of the goals of the GoodOD project.

Terminological decisions

Initiation to ontology design faces the problem that every community involved has its own terminology. There are the vocabularies of philosophical ontology, set theory, DL, Protégé, knowledge management, and so on. This can easily lead to ambiguities and accordingly to confusion on the side of the learner. We decided to choose a rigid front end orientation and to use Protégé vocabulary wherever possible, as this is what modelers see when working with this editor.

Curricular development

The curricular development followed a widely adopted method from medical education (Kern *et al.*, 1998), which uses a six-step iterative development procedure: the problem identification and general needs assessment, the needs assessment of the targeted learners, the definition of goals and specific measurable objectives, the selection of educational strategies, the implementation of the curriculum, concluded by evaluation and feedback.

Based on a general requirements analysis as given in the background section, we identified the targeted learners of the curriculum as employees or students in the life sciences with a background in computer science, either as a minor subject or part of a bioinformatics curriculum. The needs of these learners with regard to ontology development were assessed.

Content selection

Prior to the curriculum development, a guideline had been developed, in which the authors elucidated the principles of good

[1] http://owl.cs.manchester.ac.uk/tutorials/protegeowltutorial/

[2] http://www.co-ode.org/ontologies/pizza/

[3] http://protege.stanford.edu/publications/ontology_development/ontology101-noy-mcguinness.html

[4] http://www.meteck.org/teaching/SA/MOWS10OntoEngCouse.html

biomedical ontology design using a decidable description logics in OWL format. The objective of the guideline is to provide practical guidance for novices and experts on how to use the abovementioned representation framework, and how to address ontology engineering projects using top-level ontologies and ontology design patterns. Based on this guideline and in view of the curriculum time constraints the most appropriate content for the learning objectives was selected.

The main step in curricular development is the formulation of goals for the complete curriculum and specific educational objectives. Based on the problem analysis, and the general and targeted learners needs assessment, learning objectives and goals for the curriculum were specified. Although literature-based, the final selection of educational objectives was led by personal experience in a series of life science ontology development projects (Boeker *et al.*, 2007; Beisswanger *et al.*, 2008; Schober *et al.*, 2010; Schulz *et al.*, 2011).

We decided to use the Protégé editor because of its free availability and its support for OWL and automatic reasoners. In view of our guidelines we decided, however, not to use all available features of Protégé. For example, we did not include individuals and data type properties, because they are of no relevance for the modeling of proper ontological facts and even apt to mislead a novice developer.

A small sample ontology (Zoo ontology) was developed as a running example throughout most lectures and exercises. We decided to use the BioTop domain top level ontology (Beisswanger *et al.*, 2008) in the modules concerned with top-level ontologies and created a reduced version of it (BioTopLite) to restrict the information load to the essential information needed for the solving of the tasks and learning objectives.

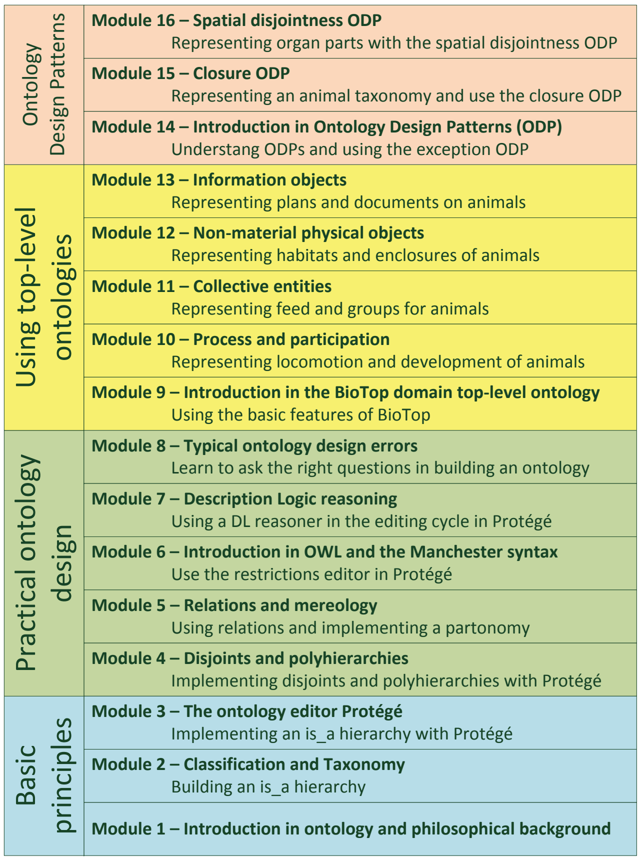

Fig. 1. Structure of the curriculum. Modules are arranged as placed in the curriculum in ascending order, each relying on the previous one.

Instructional format

As the most appropriate instructional format for the training of skills practical exercises were developed, so that the curriculum structure was designed to be based on 16 training modules (each lasting 2-3 hours) which consisted in a short introductory oral presentation of about 15 min followed by one or more practical tasks. Hands-on exercises were shaped for pair-wise execution, whereas paper-based practical exercises were adapted to groups of up to six students.

Curriculum evaluation was performed with a questionnaire consisting of 48 closed, and 12 open questions, where the first ones assessed how students judged the educational principles, the difficulty of the content, and the relevance of each module using a five-point Likert scale. Additionally opinions and attitudes with regard to problems and advantages of the course were collected.

Implementation

The curriculum was implemented in Summer 2011 as an elective summer school at Freiburg University, Germany. 24 students from Austria, Germany and Slovenia participated who either studied biology as major subject and computer science/ bio-informatics as minor subject or studied computer science/ mathematics as major subject and biology as minor subject. The participants agreed in an informed consent to the scientifically analysis of results of the summer school and to their participation in a subsequent educational study. Each participant received an expense allowance of € 500 after completing the summer school and the study. It was clearly communicated that the payment of the expense allowance would depend on their uninterrupted participation, but not on results in any of the associated tests and questionnaires.

The condensed main curriculum took place in the first five days of the summer school. In the remaining three days a quantitative study on ontology development was performed.

3 RESULTS

Curriculum structure and contents

Figure 1 shows the structure of the curriculum and the main contents of the modules. Sixteen modules with a length between two and three hours followed each other with increasing complexity. These could be grouped in four sections without sharp borders: *Basic principles*, *Practical ontology design*, *Using top-level ontologies*, and *Using ontology design patterns*.

Especially in the first phase of the curriculum, many exercises were conducted in group work without a computer, the aim being to demonstrate that major parts of the development process consist of cognitive decision steps, which are independent of technical skills or software programs.

Module structure

Most modules were organized in a similar structure, consisting of an introductory short presentation (at most 20 min), followed by a training phase in which the students worked on practical tasks alone or in pairs. In a plenary session results, suggested solutions and problems were presented and discussed, and a take-home message was formulated. All material and all exercises were laid out in a brief document, which focused on most important details without disturbing the students' creativity. Module 10 *Process and participation* is presented here as an example.

The module starts with a presentation which introduces *processes* as an important ontological category, relating it to the corresponding BioTop classes and relations: Processes (*Process*) are things which have parts in time following each other in a sequence, so that all process parts which are preceded by the next part in sequence are existentially related to the latter (**preceded-by**). Processes are only fully instantiated when they are finished. They have at least one participant (expressed by the relation pair **hasParticipant**, **participatesIn**), and they have a duration (**hasDuration**). The location (expressed by the relation pair **hasLocus**, **locusOf**) of a process has to be differentiated from its participant. The process participation can be further distinguished, which is expressed by the subrelations **hasAgent**, **hasPatient**, and **hasOutcome**. A plan (*Plan*) to do something can only be realized (**hasRealization**) by processes.

In the four exercises of this module, practical skills and implementation issues had been theoretically introduced beforehand, and were then issued as practical Protégé editing task. To provide students with a framework of basic classes and relations, they were requested to use a custom-tailored BioTopLite version, that included basic constraints during practical ontology building tasks, and limiting potential variants to facilitate the subsequent comparison of the resulting OWL artifacts.

In the first exercise of this module students had to provide a taxonomy of animal locomotion: *Swimming*, *Flying*, *Running*, *Riding* and *Digging*. These processes should be represented with at least one participant and should be correctly localized in a suitable environment, e.g. (*PortionOfAir*, *PortionOfLand*, *PortionOfWater*). The definition of locomotive processes should then be used to define *FlyingAnimal* and the other classes described along their locomotion modality. In other exercises of this module the sequence of processual parts in a developmental process had to be represented with the **precededBy** relation or different roles had to be assigned to participants in a hunting action (*Hunting*, *Hunter*, *Prey*), so that individuals that are members of the same class can have different roles.

Curriculum evaluation

The quantitative evaluation of students on the curriculum was performed on a 5-point Likert scale (1 = very good; 5 = very poor).

As shown in table 1, students evaluated the overall didactic principles of the curriculum modules and their relevance positively. The ratings for difficulty were slightly lower.

As anticipated, the qualitative answers of the students showed a large variety of partly contradictory opinions and feelings on the curriculum. However, they purport a clear view on the following aspects of the curriculum:

	Mininum (module)	Maximum (module)	Mean (SD) n=24
Didactics	1.5 Taxonomy	2.7 Immaterial object	2.1 (0.84)
Difficulty	1.8 Taxonomy	3.3 Immaterial object	2.5 (0.92)
Relevance	1.5 Process, Closure	2.4 Design errors	1.8 (0.96)

Table 1. Results of the curriculum evaluation with questionnaire on a 5-point Likert scale (1 = very good; 5 = very poor). The range as minimum and maximum of average module ratings is given with the corresponding module names. The mean and standard deviation includes the ratings of all 24 students on the complete set of 16 modules.

- The curriculum started too slowly regarding the amount and the difficulty of the module content. In the mid-curriculum the pace was evaluated to be just right, but accelerated too steeply at the end, where the more difficult issues had been addressed in little time and more transfer tasks were demanded from the participants.
- Even after eight days of hand-on work with Protégé , many students expressed lack of confidence in applying ome of its functionalities appropriately.
- Many of the students were highly motivated by the curriculum and expressed interest allocating further time to ontology engineering.
- Some students wanted to be trained more thoroughly in formal and logical backgrounds. Many students complained of "philosophical deviations" during presentations and wanted to have a more straight forward teaching focusing on clearer statements on what had to be done practically.
- In the opinion of a few students the practical usage of ontologies in different scenarios should have been shown (and even practiced). For them the benefit, meaning and technical structure of an "ontology driven semantic framework" remained unclear and should have been incorporated in the curriculum.

4 DISCUSSION

This work is based on the assumption that better educational programs for the training of life sciences ontology engineering are necessary to improve the quality of the ontological artifacts produced in this domain. Consequently, we developed a curriculum which covered the most important aspects of ontologies in the biomedical domain according to the literature, based on prior work on a ontology design guideline and our practical experience as ontology developers. Moreover, the curriculum was specifically targeted to learners with a biomedical background, additionally trained in basic and practical computer science. This profile characterizes in our view the most important stakeholders in biomedical ontology building, maintenance, deployment, implementation, and use. The curriculum has consequently been developed to serve as a hands-on guide to real-life ontology design problems, and supports the future ontology engineer throughout the ontology development life cycle.

Although students were generally very satisfied with the curriculum (as shown in the evaluation), they also signaled some major and minor curriculum improvements. If their assessment on time allocation (in the beginning too long, in the end too short) is correlated with the curriculum structure (see Fig. 1) this could mean that the introductory modules, especially on taxonomy, should be condensed. On the other hand, the modules on complex real life scenarios, beginning in the middle of the curriculum, should have been allocated more time. Some students also wanted to gain deeper insight in formal and logical aspects, which were deliberately presented superficially, as we did not consider them important for a practical approach under the given time constraints. In the future this could be addressed by providing additional readings for those interested in these topics. As a consequence of the survey, we will modify the curriculum by integrating short tutorial sections about the principle and basic subject matter into modules with more practical impact. In addition to shifting the focus more towards practical design problems, we consider developing a new module which addresses the logical and formal foundations of ontology and description logic.

The modularized approach of the curriculum gives the educator freedom for customization, aligned to either specific requirements of certain user groups or to curriculum length. It is relatively easy to focus the modules on certain topics or to compress them for students with more pre-existing skills and knowledge.

There are two major sources for further development and improvement of the curriculum. Students evaluations as described above and typical mistakes and errors made in the exercises during the summer school. The latter were documented in detail and are currently subject of a systematic analysis. Future work will also include the design of online modules on the basis of the current curriculum, which enables the design of flexible courses in a blended learning approach, i.e. as a combination of electronic with face-to-face teaching.

5 CONCLUSION

The dissemination of good ontological practice in the life sciences is not only a matter of research, but also of the availability of professional trainers who impart the knowledge in this highly interdisciplinary area. On this basis, a curriculum of 16 modules was developed to train students in the biomedical domain with a background in computer sciences how to practically develop and build ontologies in OWL using the Protégé editor. The implementation of this curriculum in a summer school setting including 24 students clearly showed that it is adequate to convey and train the complex knowledge and skills of ontology engineering in a week long course.

This curriculum represents an outline, how to successfully train students and researchers in the life sciences with the abstract matters of ontology engineering, enabling them to formally represent their domain knowledge with maximum profit.

ACKNOWLEDGEMENTS

This work was supported by the Deutsche Forschungsgemeinschaft (DFG) grant JA 1904/2-1, SCHU 2515/1-1 within the project GoodOD (Good Ontology Design), http://www.iph.uni-rostock.de/Good-Ontology-Design.902.0.html.

REFERENCES

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., Harris, M. A., Hill, D. P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J. C., Richardson, J. E., Ringwald, M., Rubin, G. M., and Sherlock, G. (2000). Gene ontology: tool for the unification of biology. *Nature genetics*, **25**(1), 25–29. PMID: 10802651 PMCID: 3037419.

Baader, F., Calvanese, D., McGuiness, D., Nardi, D., and Patel-Schneider, P., editors (2007). *The description logic handbook : theory, implementation, and applications*. Cambridge University Press, Cambridge, New York, 2nd ed. edition.

Beisswanger, E., Schulz, S., Stenzhorn, H., and Hahn, U. (2008). Biotop: An upper domain ontology for the life sciences. a description of its current structure, contents and interfaces to obo ontologies. *Applied Ontology*, **3**(4), 205–212.

Boeker, M., Stenzhorn, H., Kumpf, K., Bijlenga, P., Schulz, S., and Hanser, S. (2007). The @neurist ontology of intracranial aneurysms: Providing terminological services for an integrated it infrastructure. In J. M. Teich, J. Suermondt, and G. Hripcsak, editors, *American Medical Informatics Association 2007 Proceedings. Biomedical and Health Informatics: From Foundations to Applications to Policy*, pages 56–60.

Boeker, M., Tudose, I., Hastings, J., Schober, D., and Schulz, S. (2011). Unintended consequences of existential quantifications in biomedical ontologies. *BMC Bioinformatics*, **12**(1), 456.

Brochhausen, M., Burgun, A., Ceusters, W., Hasman, A., Leong, T. Y., Musen, M., Oliveira, J. L., Peleg, M., Rector, A., and Schulz, S. (2011). Discussion of "biomedical ontologies: toward scientific debate". *Methods of Information in Medicine*, **50**(3), 217–236.

Kern, D. E., Thomas, P. A., Howard, D. M., and Bass, E. B. (1998). *Curriculum Development for Medical Education: A Six-Step Approach*. The John Hopkins University Press, Baltimore and Maryland.

Liskov, B. (1987). Keynote address - data abstraction and hierarchy. *SIGPLAN Not.*, **23**(5), 1734.

Mizoguchi, R. and Kozaki (2009). Ontology engineering environments. In S. Staab and R. Studer, editors, *Handbook on Ontologies. International Handbook on Information Systems*, pages 315–336. Springer-Verlag, s.l.

Munn, K. and Smith, B. (2008). *Applied ontology: An introduction*, volume 9 of *Metaphysical research*. Ontos-Verl, Frankfurt.

Rector, A. L., Brandt, S., and Schneider, T. (2011). Getting the foot out of the pelvis: modeling problems affecting use of SNOMED CT hierarchies in practical applications. *Journal of the American Medical Informatics Association*, **18**(4), 432–440.

Schober, D., Boeker, M., Bullenkamp, J., Huszka, C., Depraetere, K., Teodoro, D., Nadah, N., Choquet, R., Daniel, C., and Schulz, S. (2010). The debugit core ontology: semantic integration of antibiotics resistance patterns. In C. Safran, S. Reti, and H. F. Marin, editors, *MEDINFO 2010 - Proceedings of the 13th World Congress on Medical Informatics*, volume 160 of *Studies in Health Technology and Informatics*, pages 1060–1064, Amsterdam. IOS Press.

Schulz, S., Suntisrivaraporn, B., Baader, F., and Boeker, M. (2009). Snomed reaching its adolescence: ontologists' and logicians' health check. *International journal of medical informatics*, **78 Suppl 1**, S86–94.

Schulz, S., Spackman, K., James, A., Cocos, C., and Boeker, M. (2011). Scalable representations of diseases in biomedical ontologies. *Journal of Biomedical Semantics*, **2**(Suppl 2), S6.

Smith, B., Köhler, J., and Kumar, A. (2004). On the application of formal principles to life science data: a case study in the gene ontology. In E. Rahm, editor, *Data Integration in the Life Sciences*, volume 2994, pages 79–94. Springer Berlin Heidelberg, Berlin, Heidelberg.

Smith, B., Ashburner, M., Rosse, C., Bard, J., Bug, W., Ceusters, W., Goldberg, L. J., Eilbeck, K., Ireland, A., Mungall, C. J., Leontis, N., Rocca-Serra, P., Ruttenberg, A., Sansone, S., Scheuermann, R. H., Shah, N., Whetzel, P. L., and Lewis, S. (2007). The OBO foundry: coordinated evolution of ontologies to support biomedical data integration. *Nat Biotech*, **25**(11), 1251–1255.