

# Partitioning of BioPortal Ontologies: An Empirical Study

Alsayed Algergawy and Birgitta König-Ries

Heinz-Nixdorf Chair for Distributed Information Systems,  
Institute for Computer Science,  
Friedrich Schiller University of Jena, Jena, Germany  
{alsayed.algergawy, birgitta.koenig-riesg}@uni-jena.de

**Abstract** BioPortal is a leading repository of biomedical ontologies developed in different formats such as OWL and OBO. There is an increasing number of ontologies as well as an increasing number of concepts available via this platform. This has sparked a number of studies analyzing different aspects of BioPortal ontologies, such as their quality and reuse. With this paper, we add a new aspect to this body of work: The current version of BioPortal supports the access to whole ontologies; however, often, users are interested to obtain access to subsets of ontologies. This is particularly true for big ontologies. This requires partitioning of ontologies. In this paper, we therefore investigate how suitable BioPortal ontologies are for being partitioned with state of the art tools.

## 1 Introduction

Ontologies provide domain knowledge in machine readable formats. They are widely used in various applications, e.g., to drive data annotation, data integration, information retrieval, and in particular widely used in biological and biomedical research [12]. Therefore, a large number of ontologies have been developed and there is a growing necessity to keep them in a common repository to make them accessible and manageable. Examples of such repositories include BioPortal [17,25,32]<sup>1</sup>, OntoBee [20]<sup>2</sup> and AgroPortal [14]<sup>3</sup>.

Since its deployment, the National Center for Biomedical Ontologies (NCBO) BioPortal has evolved to become the prevalent repository of biomedical ontologies and terminologies [17,32]. In 2008, BioPortal had 72 ontologies with around 300,000 concepts, while in the current version<sup>4</sup> it contains 821 ontologies with 8,859,512 concepts. This shows that there is a tremendous increase in both the number of ontologies and the number of concepts. BioPortal provides a number of services, amongst others to obtain ontology information, such as

<sup>1</sup> <http://bioportal.bioontology.org/>

<sup>2</sup> <http://www.ontobee.org/>

<sup>3</sup> <http://agroportal.lirmm.fr/>

<sup>4</sup> visited on 18.11.2019

ontology metadata and individual ontology terms, to search within ontologies, and to visualize them.

Due to its popularity, different aspects of BioPortal ontologies have been investigated. For example, a number of approaches have been introduced to investigate the quality of BioPortal ontologies [5,18]. Another aspect that is intensively studied w.r.t. BioPortal ontologies is *ontology reuse* [8,15,19]. However to the best of our knowledge, there is only one study focusing on the decomposition and partitioning of BioPortal ontologies [31]. This study was conducted in 2011 when the number of ontologies was 250 of which 218 were OWL or OBO ontologies. The study was conducted on that set of ontologies. This study has been replicated in 2014 by another study utilizing the same methodology and software to extend the experiments on BioPortal ontologies at that time [13]. There is no recent or comprehensive study on this important aspect, though. With this paper, we aim to fill this gap with an empirical analysis for partitioning BioPortal ontologies.

Ontology modularization covers the problem of identifying a fragment or a set of fragments of an ontology. The process of identifying a fragment of an ontology given a user input (request) is called *ontology module extraction* [9,23,24], while the process that partitions the ontology into a set of fragments is called *ontology partitioning* [2,4,22]. Ontology modularization can be used to support a number of complex tasks, such as maintenance, reuse and knowledge selection [6], reasoning [22] and integration of existing ontologies. An ontology module is defined as a reusable component of a larger or more complex ontology [7,21], which is self-contained but bears a definite association to other ontology modules, including the original ontology. In this paper, we aim to study and investigate the partitionability of BioPortal ontologies. To this end, we adopt three different partitioning approaches belonging to two different categories: *PATO*, and *OAPT* as *structural-based* and *AD* as a *logic-based* partitioning approaches. *PATO* is a tool used to partition large ontologies into smaller modules based on the structure of the class hierarchy [26,27]. *AD* (*Atomic decomposition*) depends on the definition of logical dependence that allows the definition of clumps of highly interlaced axioms (called atoms) that are never split across two or more modules [31,13]. *OAPT* (Ontology Analysis and Partition Tool) aims to split an ontology into a set of modules exploiting a seeding-based clustering approach [1,2]. We applied these approaches on the BioPortal ontology repository and analyzed the partitioning results.

The remainder of the paper is organized as follows: background is presented in Section 2. Section 3 provides an overview of the proposed methodology. The experimental evaluation of the used partitioning approaches w.r.t. Bioportal ontologies is introduced in Section 4. Finally, Section 5 concludes the paper.

## 2 Background

An ontology,  $\mathcal{O}$ , is a set of axioms, each representing a statement about the domain. The building blocks of axioms are entities, such as concepts,

properties, individuals, and data types [11,29]. We describe an ontology as a 6-tuple, denoted as  $\mathcal{O} = \{C, P, H^C, H^P, A, I\}$ .  $C$  and  $P$  are two disjoint sets of classes (concepts) and properties, respectively.  $H^C = \{(C_1, C_2) \in C \times C \mid C_1 \text{ subsumes } C_2\}$  represents the hierarchy of class subsumption. Similarly,  $H^P$  is the hierarchy between properties.  $A$  is a set of axioms and  $I$  is a set of instances associated to the set concepts  $C$  and properties  $P$ . A signature  $\mathcal{S}$  of an ontology  $\mathcal{O}$  based on a description logic  $\mathcal{L}$  is the union of concepts, properties, and instances, i.e.  $\mathcal{S} = C \cup P \cup I$ .

A module  $\mathcal{M}$  of an ontology  $\mathcal{O}$  is a reusable part of  $\mathcal{O}$ , which is self-contained but bears a definite association to other ontology modules, including the original ontology [7]. Formally, we define an *ontology module*  $\mathcal{M}_i(\mathcal{O})$  following [9,10,16]:

**Definition 1.** *A module  $\mathcal{M}_i(\mathcal{O})$  is a module of the ontology  $\mathcal{O}$  w.r.t. a description logic  $\mathcal{L}$ , if for every axiom  $\alpha$  over  $\mathcal{L}$  with  $\mathcal{S}(\alpha) \subseteq \mathcal{S}$ , we have  $\mathcal{M}_i(\mathcal{O}) \models \alpha$  if  $\mathcal{O} \models \alpha$ .*

An ontology module can be represented as a 6-tuple  $\mathcal{M}_i(\mathcal{O}) = \{C_{\mathcal{M}_i}, P_{\mathcal{M}_i}, H_{\mathcal{M}_i}^C, H_{\mathcal{M}_i}^P, A_{\mathcal{M}_i}, I_{\mathcal{M}_i}\}$ , where  $C_{\mathcal{M}_i} \subseteq C$ ,  $P_{\mathcal{M}_i} \subseteq P$ , etc. This definition implies that any information that exists or can be entailed from the module  $\mathcal{M}_i(\mathcal{O})$  should also exist or could be entailed from the original ontology  $\mathcal{O}$ . This enables the reuse of ontology modules either as they are or by enlarging them by adding more axioms. Therefore, each module can be considered as an ontology by itself. To achieve this, each module should be *self-contained*, *consistent*, and *topic-centric* [4,7].

We define the ontology modularization process (partitioning) as follows: given an ontology  $\mathcal{O}$ , partition the ontology entities into a set of modules  $\mathcal{M}_1, \mathcal{M}_2, \dots, \mathcal{M}_k$  such that the cohesion of entities in each module is high (i.e. intra-module similarity is high), while the coupling between any pair of modules is low. To conduct the analysis of the partitioning process, a number of evaluation criteria been designed as a trade-off between the modularization quality and the modularization efficiency [3,26]. In this study, we consider the number of modules, the number of entities in each module, and the time needed to achieve partitioning.

### 3 Methodology

Partitioning BioPortal ontologies into suitable partitions (modules) is certainly valuable when it comes to processing, editing, and analyzing them or reusing their parts. To investigate the partitioning aspect of BioPortal ontologies, we propose a workflow that contains the following main steps: i) get all accessible ontologies using the BioPortal API<sup>5</sup>, ii) transform these ontologies into OWL or OBO formats using OWL API<sup>6</sup>, iii) partition these ontologies using one of the following partitioning algorithms, and iv) analyse the partitioning results. To

<sup>5</sup> <http://data.bioontology.org/documentation>

<sup>6</sup> <http://owlcs.github.io/owlapi/>

keep this paper self-contained, we give a brief summary of each the partitioning algorithms used.

### 3.1 PATO

PATO is an ontology partitioning tool making use of the following steps to partition an ontology [27,28]: i) *dependency graph creation*: a graph structure is created to represent dependencies between ontology entities, where nodes of the graph are the values of "rdf:label" or "rdf:ID". ii) *graph partitioning*: a set of nodes of given minimal and maximal size for which the strength of the connection between the nodes inside the set is higher than the strength of any connection to nodes outside the set is used to determine sets of ontology elements that should be in one module, and iii) a distributed ontology is created based on the graph partitioning.

### 3.2 OAPT

The ontology analysis and partitioning tool (OAPT) [2] aims to partition ontologies into a set of modules based on exploiting the seeding-based clustering algorithm [1]. The algorithm has the following steps: i) *ranking the ontology concepts*: A first step is to quantify the importance of each concept within the concept graph (ontology) to select which concepts could be used later as *important concepts*. Some of these important concepts are then elected to be cluster heads, the seed of the partition. ii) *determine cluster heads*: the next step is to select which concepts represent cluster heads. In this context, we have to deal with two arising questions: how many cluster heads should we select? and which cluster heads? iii) *partitioning*: the seed-based algorithm initiates one partition for each cluster head. Then, it places direct children in the corresponding partition and finally, for the remaining (non-partitioned) concepts, a membership function to assign remaining nodes to their fitting partition is developed. and v) *generate module*: the following step is to generate a module for each partition preserving the required intra-relationships between concepts in the same partition as well as inter-links between concepts from different partitions.

### 3.3 AD

The atomic decomposition (AD) is a compact representation of modular structure of the ontology [13,31]. AD of an ontology  $\mathcal{O}$  is a pair consisting of a set of atoms and a directed dependency relation over these atoms [30], where an atom is a maximal set of axioms which are tightly bound to each other. For computing Atomic Decompositions we used the off-the-shelf implementation provided by Del Vescovo and Palmisano [30]. The implementation is available via Maven Central (maven.org) with an artifactId of `owlapitools-atomicdecomposition`. The current implementation of the AD approach supports extracting three types of syntactic-locality-based modules: the *bottom* module, the *top* module, and the *star* module.

## 4 Evaluation

In this section, we first describe the setup of our evaluation and then discuss evaluation results.

### 4.1 Setup

We carried out a set of experiments using a 3.4GHz Intel (R) Core i7 processor with 16GB RAM running Windows 7. We make use of the available implementation of the partitioning tools: *AD* can be accessed through this link<sup>7</sup>, *PATO*, and *OAPT* from this link<sup>8</sup>. We run this set of experiments using the BioPortal ontology repository version that contains 792 ontologies<sup>9</sup>, of which 710 are accessible and can be downloaded using the BioPortal API. 657 ontologies are represented or can be converted to OWL or OBO formats.

### 4.2 Results

We ran the *PATO*, *OAPT*, and *AD* partitioning tools to partition these 657 ontologies according to the respective partitioning algorithm implemented in the tool. In the following, we present the results of *PATO* and *OAPT* together as they are both classified as *structural-based* partitioning approaches, while the results of *AD* are presented for using different strategies, *bottom (Bot)*, *top*, and *star*, as it is classified as *logic-based* partitioning approach.

**Number of modules.** We started our analysis by considering how many ontologies can be partitioned and how many partitions are generated for each ontology. Results are summarized in Fig. 1, where results of *OAPT* and *PATO* are shown in Fig. 1a and Fig. 1b, respectively. These figures show that *OAPT* and *PATO* can partition 97% (635 out of 657) and 84% of the ontologies from the repository, respectively. However, the two partitioning tools generate different numbers of partitions according to their respective procedure.

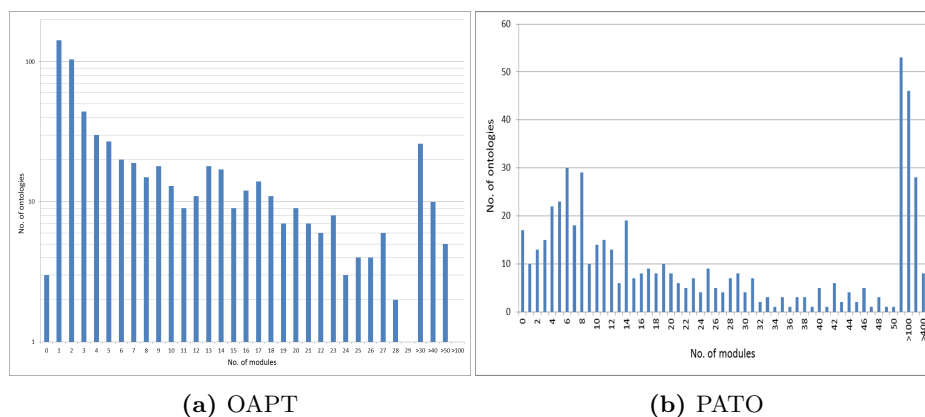
For *OAPT*, Fig. 1a shows that 142 ontologies can be represented as one-module ontologies. The two main reasons behind that are i) 101 ontologies have less than 50 concepts. Here, partitioning seems unnecessary. We report also that 123 ontologies have less than 100 concepts. That means that at least 19 ontologies with more than 100 concepts, are represented as one-module ontologies. ii) Investigating this set of 19 ontologies, we found that 15 out of them have less than 200 concepts. For these, it could also be acceptable to be represented as one-module ontology. We reviewed the remaining four ontologies (*BFLC*, *LUNGMAP-HUMAN*, *MSV* and *TM-SIGNS-AND-SYMPTS*) and we found some issues of them. For example, the *MSV* (Metagenome Sample

<sup>7</sup> <https://web.stanford.edu/~horridge/publications/2014/iswc/atomic-decomposition/data/>

<sup>8</sup> <https://github.com/fusion-jena/OAPT>

<sup>9</sup> at the time of evaluation execution

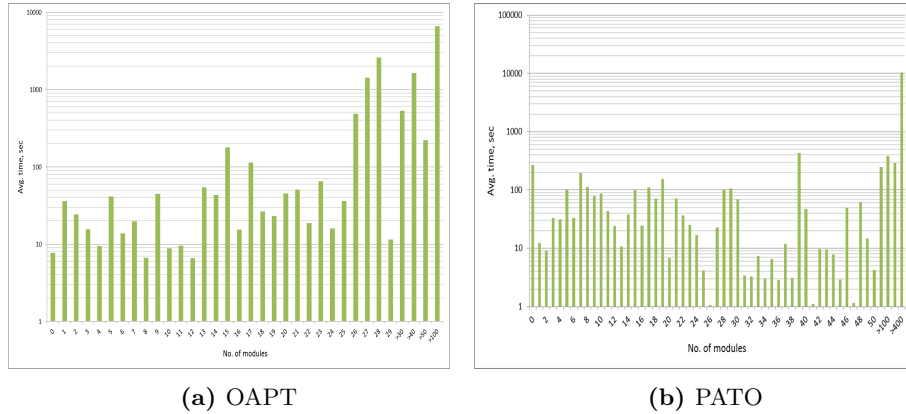
Vocabulary) ontology is in its beta version since 2017. It has 648 concepts with only five is-a relations. Fig. 1a also shows that half of the accessible ontologies (347 ontologies) in the BioPortal repository can be partitioned in up to only five partitions, while 590 ontologies can be partitioned in up to 30 modules. The remaining set of ontologies representing most larger size ontologies are partitioned into more than 30 partitions requiring more computational resources. For example, the *GO-PLUS* ontology containing 80,999 is partitioned into 50 modules. The figure also illustrates that three different ontologies (*SEQ*, *SMASH*, and *ENM*) generate zero modules, where the sequence ontology (*SEQ*) has only one concept producing a problem during the extraction of the concept, while the current JENA API fails to parse and read the other two ontologies.



**Figure 1:** No. of ontologies vs no. of modules

For *PATO*, as shown in Fig. 1b, the tool generates a large number of partitions with a small number of entities based on a defined parameter. The tool can generate 1-module category for 10 different ontologies, eight of them also appear in the same category when generated by *OAPT*. The remaining two ontologies are *SEQ* (which appears in the 0-module category by *OAPT*) and *HORD*. In total, *PATO* generated 0 modules for 17 different ontologies. This is because the tool fails to build the dependency graph for this set of ontologies. Fig. 1b also shows that among 554 ontologies 140 ontologies are partitioned into more than 50 partitions.

**Partitioning time** One important aspect that should be considered during the analysis of partitioning results is to study the partitioning performance. In this analysis we measured the time needed to execute the reading and partitioning of each ontology. We sum the execution time for the set of ontologies within the same category. Results are reported in Fig. 2. The figure summarizes the average execution time (*avg. time*) to partition an ontology within the category.



**Figure 2:** No. of ontologies vs average partitioning time

For example, as shown in Fig. 2a, the 10-module category needs an average time of 8.6 seconds to partition an ontology within the category for *OAPT*, while *PATO* needs 86.5 seconds to partition an ontology within the same category, as shown in Fig. 2b. Fig. 2a demonstrates also that the 100-module category needs an average time of 110 minutes (approximately two hours) to achieve the partitioning of an ontology. One more interesting findings that can be extracted from the figure is that the execution time depends not only on the number of modules (partitions) but also on other internal characteristics of the ontology. For example, the *FAST-EVENT* ontology has only four concepts and it needs 1286 and 103 seconds to do the partitioning using *OAPT* and *PATO*, respectively. We investigate this ontology and we found that it has 15,700 individuals. Similarly, the *CU-VO* has 11 concepts and 7320 individuals. However, it needs 920 seconds for partitioning using *OAPT*, while *PATO* fails to partition it.

**AD results.** Since results of the atomic decomposition of the BioPortal ontologies has been introduced in an earlier study [31,13], in this section we introduce the new results w.r.t. the current repository. These results are summarized in Table 1. The table shows that the AD approach using different strategies (*bottom (Bot)*, *Top*, and *Star*) was applied to the ontology repository with 657 ontologies. The different strategies can generate correct atoms (partitions) for 435, 410, and 442 ontologies using the *bottom (Bot)*, *Top*, and *Star* strategy, respectively. This represents 67% of the whole repository. Even though this category of partitioning approaches generate atoms very fast, it fails to cope with a large number of BioPortal ontologies. The table also shows that the AD approach decomposes a number of ontologies with 0 atoms, where the *Bot* and *Star* strategies produce 0 atoms for seven (the same set) ontologies, while the *Top* strategy produces 0 atoms for 31 ontologies. One important finding here is that the ontology *SMASH* is partitioned by all tools into 0 partitions (atoms).

criteria	Bot	Top	Star
No. of ontologies	511	510	509
Exception	27	27	18
0-atom	7	31	7
no-result	42	42	42
with-result	435	410	442

**Table 1:** AD results

## 5 Conclusion

BioPortal is an important resource for biomedical ontologies. It is thus worthwhile to investigate the portal and the ontologies contained therein. While existing work addresses a number of aspects, an analysis of partitionability of BioPortal ontologies was still missing. In this paper, we describe first steps to fill this gap. We introduced an empirical study and analysis of the applicability of partitioning to BioPortal ontologies. We investigated success of partitioning as well as partitioning performance for three existing partitioning tools. The study showed that overall, for many - but not for all - ontologies in BioPortal partitioning works. Failure to partition seems to be at least partially due to characteristics of the ontologies rather than of the tools. The study also shows, however, that different algorithms result in very different partitions. In future work, we plan to study the semantic content and usage of individual modules.

## 6 Acknowledgments

This work has been mostly funded by the *Deutsche Forschungsgemeinschaft (DFG)* as part of the CRC 1076 AQUADIVA.

## References

1. A. Algergawy, S. Babalou, M. J. Kargar, and S. H. Davarpanah. Seecont: A new seeding-based clustering approach for ontology matching. In *19th International Conference on Advances in Databases and Information Systems, ADBIS*, pages 245–258, 2015.
2. A. Algergawy, S. Babalou, F. Klan, and B. König-Ries. OAPT: A tool for ontology analysis and partitioning. In *Proceedings of the 19th International Conference on Extending Database Technology, EDBT*, pages 644–647, 2016.
3. A. Algergawy, S. Babalou, and B. König-Ries. A new metric to evaluate ontology modularization. In *2nd International Workshop on Summarizing and Presenting Entities and Ontologies Co-located with the 13th Extended Semantic Web Conferenc*, 2016.
4. F. Amato, A. D. Santo, V. Moscato, F. Persia, A. Picariello, and S.R.Poccia. Partitioning of ontologies driven by a structure-based approach. In *2015 IEEE International Conference on Semantic Computing*, pages 320–323, 2015.
5. M. Amith, Z. He, J. Bian, J. A. Lossio-Ventura, and C. Tao. Assessing the practice of biomedical ontology evaluation: Gaps and opportunities. *Journal of Biomedical Informatics*, 80:1–13, 2018.



6. M. d'Aquin, A. Schlicht, H. Stuckenschmidt, and M. Sabou. Ontology modularization for knowledge selection: Experiments and evaluations. In *18th International Conference on Database and Expert Systems Applications, DEXA*, pages 874–883, 2007.
7. P. Doran, V. A. M. Tamma, and L. Iannone. Ontology module extraction for ontology reuse: an ontology engineering perspective. In *Proceedings of the Sixteenth ACM Conference on Information and Knowledge Management, CIKM*, pages 61–70, 2007.
8. M. Fernández-López, M. Poveda-Villalón, M. C. Suárez-Figueroa, and A. Gómez-Pérez. Why are ontologies not reused across the same domain? *J. Web Semant.*, 57, 2019.
9. B. C. Grau, I. Horrocks, Y. Kazakov, and U. Sattler. Just the right amount: extracting modules from ontologies. In *Proceedings of the 16th International Conference on World Wide Web, WWW*, pages 717–726, 2007.
10. B. C. Grau, I. Horrocks, Y. Kazakov, and U. Sattler. Modular reuse of ontologies: Theory and practice. *J. Artif. Intell. Res. (JAIR)*, 31:273–318, 2008.
11. N. Guarino, D. Oberle, and S. Staab. What is an ontology? In *Handbook on Ontologies*, pages 1–17. 2009.
12. R. Hoehndorf, P. N. Schofield, and G. V. Gkoutos. The role of ontologies in biological and biomedical research: a functional perspective. *Briefings in Bioinformatics*, 16(6):1069–1080, 2015.
13. M. Horridge, J. Mortensen, B. Parsia, U. Sattler, and M. A. Musen. A study on the atomic decomposition of ontologies. In *The Semantic Web - ISWC 2014 - 13th International Semantic Web Conference, Riva del Garda, Italy, October 19-23, 2014. Proceedings, Part II*, pages 65–80, 2014.
14. C. Jonquet, A. Toulet, E. Arnaud, S. Aubin, E. D. Y. Kaboré, V. Emonet, J. Graybeal, M. Laporte, M. A. Musen, V. Pesce, and P. Larmande. Agroportal: A vocabulary and ontology repository for agronomy. *Computers and Electronics in Agriculture*, 144:126–143, 2018.
15. M. R. Kamdar, T. Tudorache, and M. A. Musen. A systematic analysis of term reuse and term overlap across biomedical ontologies. *Semantic Web*, 8(6):853–871, 2017.
16. B. Konev, C. Lutz, D. Walther, and F. Wolter. Model-theoretic inseparability and modularity of description logic ontologies. *Artif. Intell.*, 203:66–103, 2013.
17. M. A. Musen, N. F. Noy, N. H. Shah, P. L. Whetzel, C. G. Chute, M.-A. Story, B. Smith, and the NCBO team. The national center for biomedical ontology. *Journal of the American Medical Informatics Association*, 19(12):190–195, 2012.
18. C. Ochs, Z. He, L. Zheng, J. Geller, Y. Perl, G. Hripcsak, and M. A. Musen. Utilizing a structural meta-ontology for family-based quality assurance of the bioportal ontologies. *Journal of Biomedical Informatics*, 61:63–76, 2016.
19. C. Ochs, Y. Perl, J. Geller, S. Arabandi, T. Tudorache, and M. A. Musen. An empirical analysis of ontology reuse in bioportal. *Journal of Biomedical Informatics*, 71:165–177, 2017.
20. E. Ong, Z. Xiang, B. Zhao, Y. Liu, Y. Lin, J. Zheng, C. Mungall, M. Courtot, A. Ruttenberg, and Y. He. Ontobee: A linked ontology data server to support ontology term dereferencing, linkage, query and integration. *Nucleic Acids Research*, 45(Database-Issue):D347–D352, 2017.
21. J. Pathak, T. M. Johnson, and C. G. Chute. Survey of modular ontology techniques and their applications in the biomedical domain. *Integrated Computer-Aided Engineering*, 16(3):225–242, 2009.

22. S. Priya, Y. Guo, M. Spear, and J. Heffin. Partitioning OWL knowledge bases for parallel reasoning. In *2014 IEEE International Conference on Semantic Computing*, pages 108–115, 2014.
23. A. A. Romero, M. Kaminski, B. C. Grau, and I. Horrocks. Ontology module extraction via datalog reasoning. In *29th AAAI Conference on Artificial Intelligence*, pages 1410–1416, 2015.
24. A. A. Romero, M. Kaminski, B. C. Grau, and I. Horrocks. Module extraction in expressive ontology languages via datalog reasoning. *J. Artif. Intell. Res. (JAIR)*, 55:499–564, 2016.
25. M. Salvadores, P. R. Alexander, M. A. Musen, and N. F. Noy. Biportal as a dataset of linked biomedical ontologies and terminologies in RDF. *Semantic Web*, 4(3):277–284, 2013.
26. A. Schlicht and H. Stuckenschmidt. Towards structural criteria for ontology modularization. In *1st International Workshop on Modular Ontologies, WoMO'06, co-located with the International Semantic Web Conference, ISWC'06*, 2006.
27. A. Schlicht and H. Stuckenschmidt. A flexible partitioning tool for large ontologies. In *International Conference on Web Intelligence, WI*, pages 482–488, 2008.
28. H. Stuckenschmidt and M. C. A. Klein. Structure-based partitioning of large concept hierarchies. In *Third International Semantic Web Conference, ISWC 2004*, pages 289–303, 2004.
29. R. Studer, V. R. Benjamins, and D. Fensel. Knowledge engineering: Principles and methods. *Data Knowl. Eng.*, 25(1-2):161–197, 1998.
30. C. D. Vescovo. *The Modular Structure of an Ontology: Atomic Decomposition and its applications*. PhD thesis, The University of Manchester, 2013.
31. C. D. Vescovo, D. Gessler, P. Klinov, B. Parsia, U. Sattler, T. Schneider, and A. Winget. Decomposition and modular structure of biportal ontologies. In *The Semantic Web - ISWC 2011 - 10th International Semantic Web Conference, Bonn, Germany, October 23-27, 2011, Proceedings, Part I*, pages 130–145, 2011.
32. P. L. Whetzel, N. F. Noy, N. H. Shah, P. R. Alexander, C. Nyulas, T. Tudorache, and M. A. Musen. Biportal: enhanced functionality via new web services from the national center for biomedical ontology to access and use ontologies in software applications. *Nucleic Acids Research*, 39(Web-Server-Issue):541–545, 2011.