eXtensible ontology development (XOD) using web-based Ontoanimal tools

Edison Ong, Yongqun He
University of Michigan Medical School
Ann Arbor, MI, USA

Jie Zheng
University of Pennsylvania
Philadelphia, PA, USA

***Abstract*—The eXtensible ontology development (XOD) strategy proposes four principles to support interoperable and robust ontology development. These principles include ontology term reuse, semantic alignment, design pattern usage, and community extensibility. In this software demo, we show how Ontoanimal tools (e.g., Ontofox, Ontodog, Ontorat, and Ontokiwi) can be used to support the implementation of these XOD principles. The development of the Cell Line Ontology (CLO) is used for the demonstration.**

***Keywords*—eXtensible ontology development; XOD; Ontofox; Ontodog; Ontorat; Ontokiwi; Cell Line Ontology; CLO**

I. Introduction

Hundreds of ontologies have been developed to support biomedical research. Given the increasingly large number of biomedical ontologies developed, it is important to ensure that newly developed ontologies are interoperable with existing ontologies. However, this is a very challenging problem.

To address the ontology interoperability issue, the "eXtensible Ontology Development" (XOD) strategy is a newly proposed strategy for ontology development [1]. This strategy includes four principles. Ontology term reuse (XOD1) emphasizes the reuse of terms from existing reliable ontologies instead of reinventing the wheel. Ontology semantic alignment (XOD2) focuses on the alignment of imported terms from external ontologies and newly added terms with the same semantic framework. Ontology design patterns (ODP) usage (XOD3) emphasizes new term generation and existing term editing based on ODPs. Community extensibility (XOD4) states that the ontology should be co-developed and applied to more use cases by more people in a broader community. These XOD principles significantly support ontology interoperability and robust ontology development [1].

The collection of Ontoanimal tools is a suite of web-based tools that are primarily developed to support ontology development, visualization, query and analysis. Among the Ontoanimal tools are Ontofox [2], Ontodog [3], Ontorat [4], and Ontokiwi/Ontobedia [5]. These web tools are widely used in the biomedical ontology community, especially for those who have no or limited software programming skills.

Fig. 1 illustrates the pipeline for applyingthe XOD principles and Ontoanimal tools to support extensible ontology development. In this software demonstration, we will introduce the XOD principles and Ontoanimal tools, and desmonstrate how these can be used to develop the Cell Line Ontology (CLO), a community-based ontology in the field of cell lines and cell line cells [6].

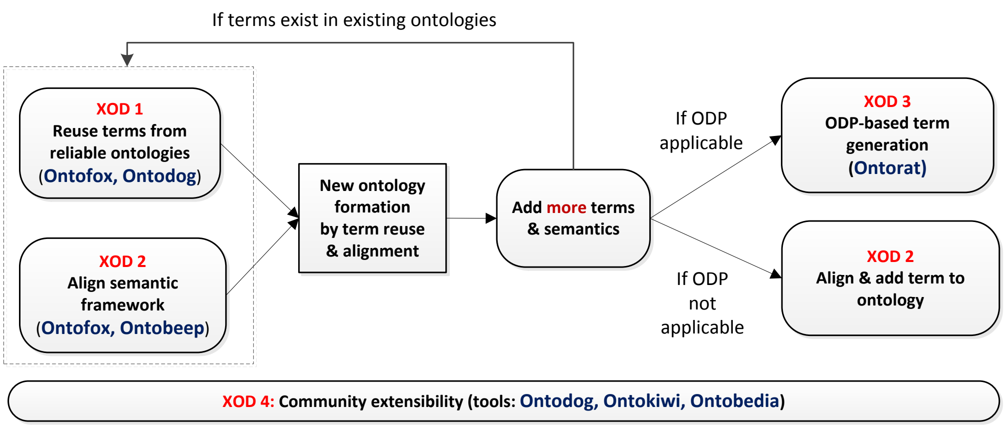

Fig. 1. Pipeline of applying XOD principles and associated Ontoanimal tools for eXtensible ontology development.

II. XOD principles supported by Ontoanimal tools

A. Reuse terms (XOD1) using Ontofox and Ontodog

Instead of reinventing the wheel, XOD 1 emphasizes the reuse of terms from existing reliable ontologies. Both Ontofox [2] and Ontodog [3] can be used to support the extraction of terms from other ontologies. Ontofox (http://ontofox.hegroup.org/) extracts selected classes, properties, annotations, and their related terms from source ontologies and saves the results in the OWL format [2]. Ontofox can extract different levels of intermediate terms between the required terms and a chosen higher level or top term.The levels of intermediate terms to be extracted can be customized in Ontofox. Ontodog (http://ontodog.hegroup.org/) can also extract a subset of ontology terms and axioms. Unlike Ontofox, Ontodog uses Excel or tab-delimited input files to identify which terms to retrieve and the users can also add user-specified annotations to the retrieved terms. Both Ontofox and Ontodog can generate OWL output files, which can be imported to a newly developed ontology.

As an example of how XOD1 can be applied to new ontology development, Ontofox was first used to extract related terms from many existing ontologies such as the Ontology for Biomedical Investigations (OBI), Cell Ontology (CL), and UBERONmulti-species anatomy ontology. All the Ontofox output OWL files were then imported to the CLO ontology using the owl:imports feature.

B. Semantic alignment (XOD2) using Ontofox and Ontobeep

XOD2 focuses on the alignment of imported terms from existingontologies and newly added terms with the same semantic framework. The top level ontology alignment can be designed and conducted manually. It is suggested to reuse terms from ontologies (e.g., OBO ontologies) that follow the same semantic structure. Ontofox supports semantic alignment by automatically extracting semantic axioms [2]. With this feature, all the OWL files of OBO ontologies imported to CLO were well aligned. In cases where Ontofox-assigned alignment did not align well, manual checking and alignment need to be used. For the manual status checking, Ontobeep (http://www.ontobee.org/ontobeep) can be used to compare two or three ontologies, e.g., CLO, OBI, and CL, and identify whether these different ontologies are aligned with the same semantic framework.

C. ODP-based term generation (XOD3) using Ontorat

Instead of adding one new term or editing one existing term at a time, XOD 3 emphasizes the addition or editing of a group of terms based on ontology design patterns (ODPs). Ontorat (http://ontorat.hegroup.org) generates and edits ontology terms and axioms and provides term annotations [4]. Ontorat uses reusable ODPs to solve recurrent modeling problems. Based on a specific ODP, an Excel or tab-delimited text file is used to provide a list of terms with associated annotations, and a set of rules are generated to define the relations among those terms and annotations. With these as input, an OWL format ontology file will be generated and be available for users to download on the Ontorat website.

For example, Ontorat was applied to automatically generate over 1,000 Japan RIKEN cell line cell terms with both logical axioms and rich annotation axioms in the CLO [6]. In the Excel file that contains the detailed information for each of these cell line cells, each row represents the information of a cell line cell type, and a column represents a specific class type or annotation property (e.g., parent cell type, label, and ontology ID for parent cell type or label). The information of cell line cells with associated annotations is provided to the Ontorat program. With additional rule settings provided on the Ontorat website, Ontorat is able to generate an output OWL file which was then added and incorporated to the CLO.

D. Community extensibility (XOD4) using Ontodog and Ontokiwi

XOD 4 states that the ontology should be co-developed and applied to more use cases by more people in a broader community. Ontodog supports the generation of an ontology community view, which can include community-specific annotations, such as ‘user-preferred label’, and be generated for the complete ontology or a subset of it [3]. Ontokiwi is an application that combines the features ofWikipedia-like community discussion features and Ontobee-like semantical ontology display and visualization features on the same website [5]. As an Ontokiwi-based application, Ontobedia (http://ontobedia.hegroup.org/) provides the users a platform for importing, editing, annotating, and discussing structured ontologies. In our software demonstration, we will show to the attendees how we can edit the Ontobedia website for the described features [5].

III. Summary and Discussion

The XOD principles provide a state-of-the-art strategy for efficient, effective, and extensible ontology development. Ontoanimal tools provide a good suite of tools to support the XOD-based ontology development. The CLO development use casedemonstrates how such XOD strategy can be achieved with the Ontoanimal tools.

In addition to the web-based Ontoanimal tools, many other tools, such as the web-based WebProtege tool (https://webprotege.stanford.edu/) and the command-line tool ROBOT (https://github.com/ontodev/robot), are available to support XOD principles, and are summarized in the original XOD paper [1].

Acknowledgment

The development of Ontoanimal tools was primarily supported by a NIH R01 grant (1R01AI081062). The CLO and LINCS work were supported by an external data science research award of the grant U54HL127624 (BD2K LINCS Data Coordination and Integration Center, DCIC).

References

[1] Y. He, Z. Xiang, J. Zheng, Y. Lin, J. A. Overton, and E. Ong, "The eXtensible ontology development (XOD) principles and tool implementation to support ontology interoperability," *J Biomed Semantics,* vol. 9, p. 3, Jan 12 2018.

[2] Z. Xiang, M. Courtot, R. R. Brinkman, A. Ruttenberg, and Y. He, "OntoFox: web-based support for ontology reuse," *BMC Res Notes,* vol. 3:175, pp. 1-12, 2010.

[3] J. Zheng, Z. Xiang, C. J. Stoeckert, Jr., and Y. He, "Ontodog: a web-based ontology community view generation tool," *Bioinformatics,* vol. 30, pp. 1340-2, Feb 1 2014.

[4] Z. Xiang, J. Zheng, Y. Lin, and Y. He, "Ontorat: Automatic generation of new ontology terms, an-notations, and axioms based on ontology design patterns," *Journal of Biomedical Semantics,* vol. 6, p. 4 (10 pages), July 24-27 2015.

[5] E. Ong and Y. He, "Community-based Ontology Development, Annotation and Discussion with MediaWiki extension Ontokiwi and Ontokiwi-based Ontobedia," *AMIA Jt Summits Transl Sci Proc,* vol. 2016, pp. 65-74, 2016.

[6] S. Sarntivijai, Y. Lin, Z. Xiang, T. F. Meehan, A. D. Diehl, U. D. Vempati, *et al.*, "CLO: The Cell Line Ontology," *J Biomed Semantics,* vol. 5, p. 37, 2014.