

# ONTOPARTS: a tool to select part-whole relations in OWL ontologies

Annette Morales-González<sup>1</sup>, Francis C. Fernández-Reyes<sup>2</sup>, C. Maria Keet<sup>3</sup>

<sup>1</sup>Advanced Technologies Application Center, CENATAV, Havana City, Cuba

<sup>2</sup>Instituto Superior Politécnico “José Antonio Echeverría” (CUJAE), Cuba

<sup>3</sup>School of Computer Science, University of KwaZulu-Natal, South Africa

amorales@cenatav.co.cu, ffernandez@ceis.cujae.edu.cu, keet@ukzn.ac.za

**Abstract.** Representing part-whole and mereotopological relations in an ontology is a well-known challenge. We have structured 23 types of part-whole relations and hidden the complexities of the underlying mereotopological theory behind a user-friendly tool: ONTOPARTS. It automates modelling guidelines using, mainly, the categories from DOLCE so as to take shortcuts in the selection process, and it includes examples and verbalizations to increase understandability. The modeller’s domain ontology, represented in any of the OWL species, can be updated automatically with the selected relation with a simple one-click button.

## 1 Introduction

There are a plethora of part-whole relations used in ontology development (e.g., [1, 2]), and for modellers who are not expert in this field, it is known to be difficult to obtain an overview of the options and subsequently to select the appropriate part-whole relation between entities. This is complicated further by the differences in expressiveness of the OWL species and that, when the appropriate relation is chosen, it can help overcome weaknesses in representation and reasoning over OWL 2-formalised ontologies [3]. To solve these issues, we have developed an ontology-inspired part-whole relation selection tool, ONTOPARTS, which can be used with the different OWL species, and covers part-whole relations, mereology, and mereotopology (parthood and location).

The theoretical analysis behind the design of ONTOPARTS is based on the extension of the taxonomy of part-whole relations of [1] with the addition of a taxonomy of formally defined mereotopological relations, which is driven by the KGEMT mereotopological theory [4], resulting in a taxonomy of 23 part-whole relations—mereological, mereotopological, and meronymic—ensuring a solid ontological and logic-based foundation; details can be found in [3]. Although some prior work in modelling guidance for part-whole relations exists in the context of the ORM conceptual modelling language [5] and for topological relations only [6], to the best of our knowledge, there is no other software tool for automating and guiding the selection of part-whole relations in general or applied to OWL ontologies.

The demonstration consists of the presentation of ONTOPARTS and its capabilities to assist modellers in making decisions regarding the correct part-whole relation that may exist among the classes in their ontology. The tool, additional files, and demo videos can be consulted in the online supplementary material at <http://www.meteck.org/files/ontopartssup/supindex.html>.

In the remainder of the demo paper, we describe aspects related to design and implementation of the tool, and provide an illustration how the tool works.

## 2 Design and implementation of ONTOPARTS

**Design.** The main requirement of the software is to hide the logic involved in the formal definitions of the 23 part-whole relations. Therefore, the user should be guided to make the decision through a series of steps in an intuitive and effortless way. Besides, the selection procedure should be made as short as possible, leading the user to a small subset of relationship suggestions relevant enough to make the selection of the most suitable one. Since users may be more familiar with the domain and range categories of the classes they are working with rather than with the relations' definitions, it is important to provide a set of (top-level) categories that will help discriminate among relations, which, in turn, streamlines the criteria for selecting the relations and eliminates the possibility of making errors in typing the relation. Usability is to be enhanced by providing simple examples for each relation and category, and pseudo-natural language sentences of the candidate axioms have to be generated. Last, the user should have the possibility to save the selected relation to the OWL ontology file from where the involved classes were taken.

Given the core requirements, several design decisions were made for ONTOPARTS. In order to quickly assess the contribution of the tool for the intended purpose, we chose to use a rapid way of prototyping to develop the software. A stand-alone tool that works with OWL files was developed, allowing the ontologist not to be bound to a single ontology editor. We chose to use the DOLCE top-level ontology categories to standardize the relationships' decision criteria (though another top-level ontology could have been chosen as well). Important for solid software design, we used activity diagrams to describe the steps to be executed to interact with ONTOPARTS and to select the appropriate relation, therewith making the selection principles technology-independent; Fig. 1 shows one such activity diagram.

**Implementation.** The tool has been implemented in C# using the .Net Framework. The Jena2 ontology API [7] has been used to handle OWL files and avail of its features to update an ontology. The updates are saved back into the OWL file using the standard RDF/XML syntax, including the axiom involving the selected part-whole relation between the classes that play the part- and whole-role, and the whole taxonomy of 23 part-whole relations. The tool has been evaluated experimentally [3], and feedback was incorporated, so that ONTOPARTS is a fully working prototype.

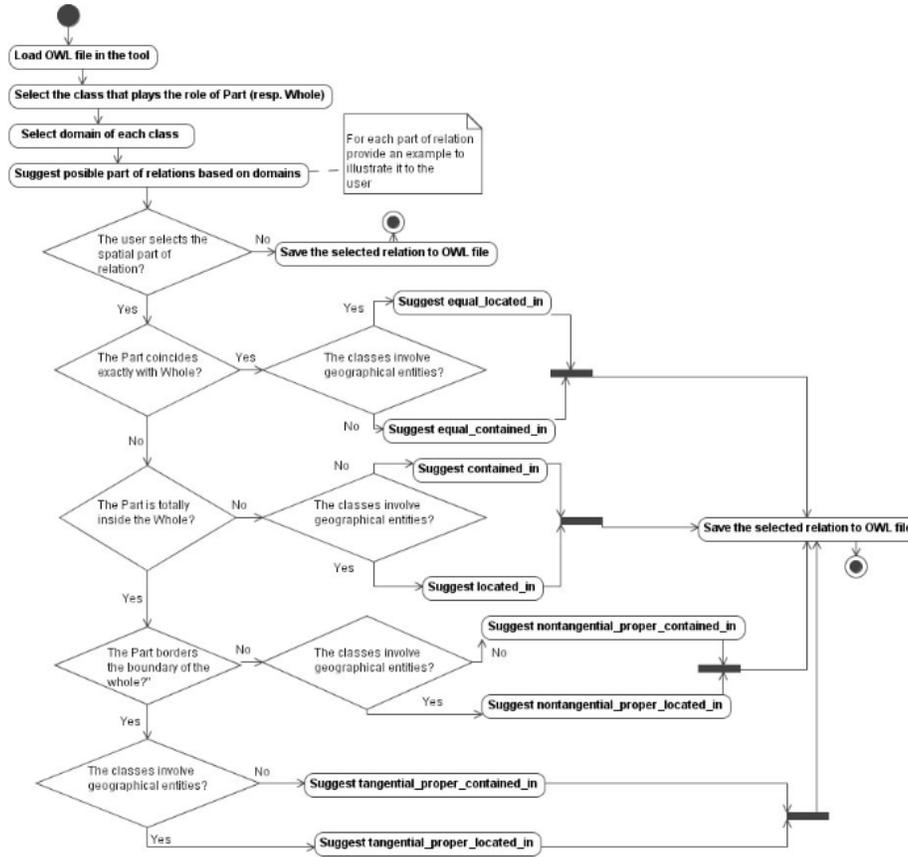
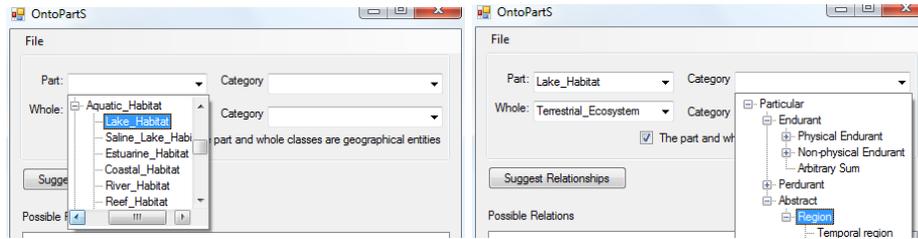


Fig. 1. One of the activity diagrams; this one shows the steps for using the tool and the decision making process to select the correct mereotopological relation.

### 3 Description of the selection procedure

ONTOPARTS guides the ontologist in the process of making the decision in an intuitive way. We illustrate the selection procedure through two examples.

**Example 1** Suppose a modeler is developing an earth science ontology and has to figure out the relation between `Lake_Habitat` and `Terrestrial_Ecosystem`. The first step is to load the OWL file in ONTOPARTS, and then select the class that represents the part, `Lake_Habitat`, and the class that represents the whole, `Terrestrial_Ecosystem` (see Fig. 2a). ONTOPARTS will provide the set of possibly suitable part-whole relations (0-4) once the user has specified the domain and range categories (currently taken from DOLCE), as depicted in Fig. 2b. Examples for each DOLCE category are shown by hovering the pointer over the terms in the taxonomy; e.g., hovering over `Achievement` shows “Ex: A conference, an ascent, a performance”. If the selected categories are regions (or any



a) Selecting the part and whole classes from the loaded OWL file

b) Selecting the categories for each entity

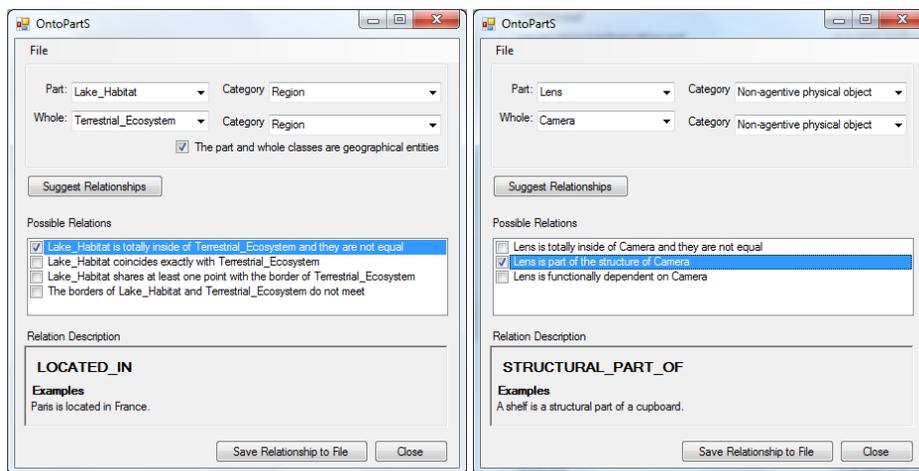
**Fig. 2.** ONTOPARTS interface: selection of classes and categories.

of its subclasses), we enter the branch of mereotopological relations and one must specify whether the regions correspond to geographical entities (2D, versus 3D containment). `Lake_Habitat` and `Terrestrial_Ecosystem` have the dual notion indicating a region and being a particular type of enduring entity, and in a map-making context, the 2D perspective is chosen. Subsequently, one clicks on the button “Suggest relationships”. The amount of relations suggested depends on the chosen categories; which are four in this case (Fig. 3a); if the selected classes were, e.g., processes, then there is only one option (`involved_in`, as ONTOPARTS includes the taxonomy of [1]). Each proposed relation is verbalised to make the option more understandable, e.g., `Lake_Habitat` is totally inside of `Terrestrial_Ecosystem` and they are not equal (i.e., located in), and an illustrative example is shown as an additional guide (see Fig. 3a). Once the desired relation is selected by marking the corresponding checkbox, one can choose to add it to the OWL file by clicking the button “Save relationship to file” (Fig. 3a bottom) and continue with other classes and selection of a part-whole relation or with developing the ontology in the ontology development environment of choice.  $\diamond$

**Example 2** Suppose a modeler or domain expert is developing a photography ontology and wants to choose a relation between `Lens` and `Camera`, where `Lens` is selected as the part class, and `Camera` as the whole. The categories for both classes correspond to `Non-agentive physical objects` in DOLCE. When the user clicks `Suggest relationships`, we obtain three possible relations between these classes. By selecting each of the suggested relationships, one can read its verbalization in context, thereby clarifying examples for each option, so as to help deciding which one is the correct one. This process is summarized in Fig. 3b.  $\diamond$

## 4 Discussion

With this demonstration we will show how ONTOPARTS can sensitize the modeller to the myriad of part-whole relations in a piecemeal fashion, how to make a selection of the appropriate part-whole relation for the identified classes, and achieve the corresponding ontology update effortlessly. Attendees also can bring



a) Example of the meretopological branch

b) Example of the mereological branch

**Fig. 3.** Relationships suggested by ONTOPARTS.

their own ontology to augment it with part-whole relations or to check its quality with respect to what was already represented in their ontology and compare it with ONTOPARTS' suggestions.

ONTOPARTS was experimentally evaluated using two groups of students with different degrees of expertise and a smaller group of experienced researchers. Although we cannot claim that the tool leads to statistically significant less modelling errors, it does assist performing the selection process so that it is done more efficiently and quickly [3].

Current and future work pertains to adding more modelling guidance features, and developing a plugin for Protégé.

## References

1. Keet, C.M., Artale, A.: Representing and reasoning over a taxonomy of part-whole relations. *Applied Ontology* **3**(1-2) (2008) 91–110
2. Mejino, J.L.V., et al.: Representing complexity in part-whole relationships within the foundational model of anatomy. In: *Proc. of AMIA Fall Symp.* (2003) 450–454
3. Keet, C.M., Fernández-Reyes, F.d.l.C., Morales-González, A.: Representing meretopological relations in OWL ontologies with OntoPartS. In Simperl, et al., eds.: *Proc. of ESWC'12*. Volume 7295 of LNCS., Springer (2012) 240–254
4. Varzi, A.: Spatial reasoning and ontology: parts, wholes, and locations. In: *Handbook of Spatial Logics*. Berlin Heidelberg: Springer Verlag (2007) 945–1038
5. Keet, C.M.: Part-whole relations in object-role models. In: *Proc. of ORM'06, OTM Workshops 2006*. Volume 4278 of LNCS., Springer (2006) 1116–1127
6. Yang, W., et al.: A model for classification of topological relationships between two spatial objects. In: *FSKD* (2). Volume 3614 of LNCS., Springer (2005) 723–726
7. Dickinson, I.: The Jena ontology API (2009) <http://jena.sourceforge.net/ontology/index.html>.