Modularizing Ontologies

Author: Mathieu d'Aquin

Motivation

As large monolithic ontologies are difficult to handle and maintain, the task of modularizing an ontology consists in identifying components (modules) of this ontology that can be considered separately, while they are interlinked with other modules. The end benefit of modularizing an ontology can be, depending on the particular application or scenario, to improve performance by enabling the distribution or targeted processing, to facilitate the development and maintenance of the ontology by dividing it in loosely coupled, self-contained components, or to facilitate the reuse of parts of the ontology.

What is Ontology Modularization?

Ontology Modularization

Definition

Ontology Modularization refers to the activity of identifying one or more modules in an ontology with the purpose of supporting reuse or maintenance.

Goal

The modularization activity offers a way to cut-down potentially large ontologies into smaller, more manageable modules.

Input

An ontology.

Output

A module or a set of modules from the input ontology. In practice, ontology modules are themselves ontologies.

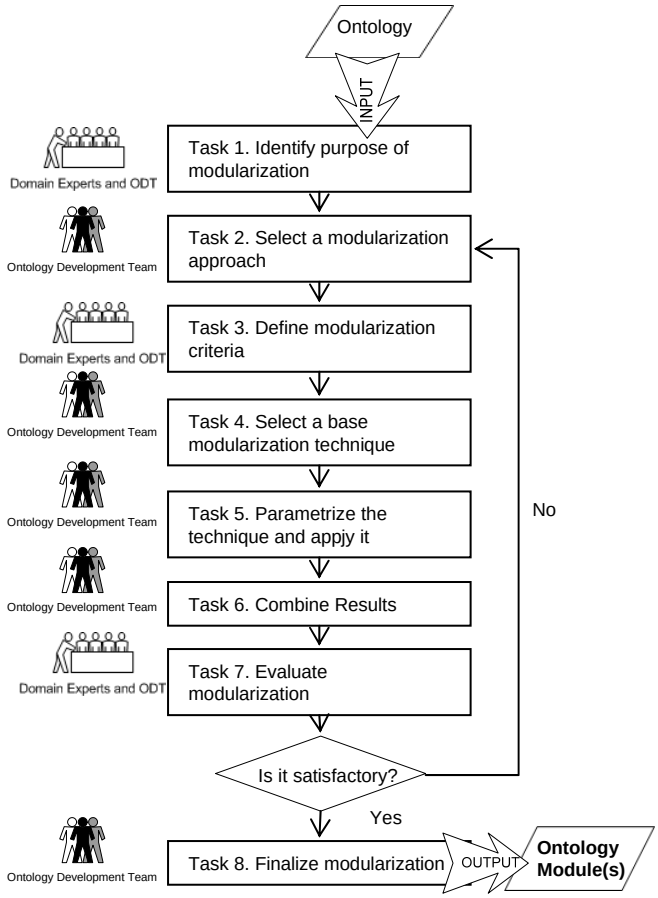
Who

Ontology engineer (ontology development team), curator of the ontology, preferably with the help of domain experts.

When

In scenario 3 for facilitating the reuse of ontological resources and in scenario 4, as part of the re-engineering process. Also in scenario 8, as part of the restructuring activity.

What is the process?

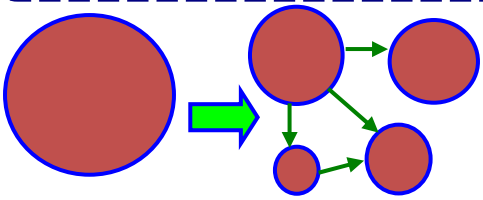


The process in detail (1)

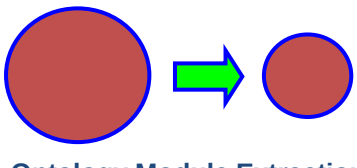
Task 1. Identify purpose of modularization.

The goal of this task is to make explicit the reason why the considered ontology should be modularized. In other terms, the outcome of this task is a clear description of the application scenario in which modularization and ontology modules are used, as well as the expected benefit of the modular approach. [1] describes some typical use case for modularization and concrete examples from NeOn.

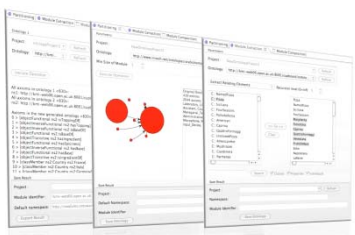
Identifying the purpose of modularization is essential for the next tasks, in particular, to select the appropriate modularization technique and criteria to maximize the expected benefit of modularization.



Ontology Partitioning



Ontology Module Extraction



Modularizing Ontologies

The process in detail (2)

Task 2. Select a modularization approach.

We distinguish two different types of modularization: ontology partitioning and ontology module extraction. Ontology partitioning relates to the automatic decomposition of an ontology into a number of interrelated modules covering all together the entire ontology. Ontology module extraction corresponds to creating a module from a sub-part of the ontology which is specifically relevant to a sub-domain of the ontology.

It is generally easy to decide which ontology to choose according the modularization purpose:

- Whenever the purpose relates to the entire ontology (i.e., improving maintenance and, in some cases, performance), a partitioning approach should be considered.
- Whenever the purpose relates to extracting specific parts of an ontology (e.g., to customize it or reuse it partially), module extraction should be considered.

However, considering that the present methodological guideline favours an iterative approach, it may occur that the two approaches can be combined, extracting, for example, modules from the result of a partitioning technique.

Task 3. Defining modularization criteria.

The modularization criteria define the basic characteristics that the resulting modules should have, i.e., what should go into a module. In [2] a set of criteria typically employed for modularization is given (e.g., logical completeness and correctness with respect to the original ontology, size, relation between modules). The criteria to emphasize should be decided depending on the purpose of modularization. For example, if the goal is to improve the reasoning procedure, logical criteria should be favoured. Unfortunately, while work in [2] provides a list of common criteria and insights on their importance in different scenarios, the choice of the right criteria to apply is highly dependent on a particular situation and has to be left to the ontology engineers to decide.

Task 4. Selecting a base modularization technique.

There is a great variety of techniques and tools for ontology modularization. In [2] we showed that these techniques implement a different intuition of what should be in a module, and so, there is no universal definition of what an ontology module should contain. In other words, it is necessary to select the most appropriate technique depending on the criteria to apply.

Task 5. Parameterizing the technique and applying it.

Depending on the technique selected in the previous task, there may be various parameters required to obtain interesting and useful results. For example, methods based on graph analysis often use parameters such as the density of interconnection between entities or the recursion level to which the graph should be traversed.

Task 6. Combining results.

As mentioned earlier, we favour an iterative process where the adequate modules are produced by refining and combining the results obtained with various parameters, techniques and approaches. Therefore, at every iteration, every time a new (set of) module(s) is produced, it is necessary to integrate it with the modules that were produced at previous iterations.

Operators for combining modules have been included as a plug-in in the NeOn toolkit (Module Composition) [3], as well as plug-ins for ontology partitioning and module extraction, to facilitate this task.

Task 7. Evaluating the result.

The evaluation of the result of the modularization (meaning the complete set of modules generated) is a crucial part of the iterative and interactive process we promote. Indeed, it depends on this evaluation whether a new iteration is necessary, applying a new set of criteria and a new technique, or if the current (set of) modules is satisfactory considering the application scenario. There are two ways in which the modularization could be evaluated: by checking the criteria or by testing against the purpose of modularization.

Task 8. Finalizing modularization.

Once the produced modularization has been judged satisfactory, an additional step can be required for it to be deployed and exploited in an application. For example, it is usually necessary to revise the identifiers of each of the modules so that they follow the conventions employed in the target application, re-establish links between modules, or simply deploy the resulting modules in a way that it is made accessible in the target application and the editorial workflow.

Additional information:

Contact person: m.daquin@open.ac.uk

- *NeOn Deliverable D5.4.2 (http://www.neon-project.org/web-content/images/Publications/neon_2009_d542.pdf)*
- *NeOn Deliverable D1.1.3 [1]*
- *d'Aquin, M., Schlicht, A., Stuckenschmidt, H., Sabou, M., (2009) Criteria and Evaluation for Ontology Modularization Technique. In Modular Ontologies: Concepts, Theories and Techniques. [2]*
- *NeOn Deliverable D1.1.4 [3]*