Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence (2008)

An Architecture and Formalism For Handling Modular Ontologies

Faezeh Ensan
Faculty of Computer Science,
University of New Brunswick, Fredericton, Canada
faezeh.ensan@unb.ca

Abstract

The goal of my ongoing work is to provide an architecture for developing and manipulating modular ontologies in such a way that each ontology module can plug into or unplug from an ontology. This architecture builds on top of a fundamental formalism for modular ontologies. Through this formalism we are able to define mechanisms for integrating different modules and develop algorithms for reasoning over the integrated modules. The resolution of inconsistencies arisen by conflicting axioms in different modules as well as the investigation of the impact of changes in a module on the other ontology modules are two important issues that need to be taken into consideration during the development of the formalism. Here, we briefly review the overall structure of the research work that I intended to conduct.

Problem Statement

The advantage of developing software applications in a modular manner has inspired knowledge engineering researchers to work on the application of modularity in the engineering of ontologies. In comparison with monolithic ontologies, reasoning in modular ontologies is more efficient, since reasoning engines only need to consider the knowledge bases of the relevant modules to the reasoning process (Stuckenschmidt and Klein 2003). Locating and resolving inconsistencies, especially in cases where some parts of an ontology may evolve during time, can be performed more conveniently through the existence of a well-defined formalism for modular ontologies.

The aim of my thesis is to design an architecture for the integration of independent ontology modules over an interface-based formalism for modular ontologies. Within the framework of this architecture, an ontology can be seen as the integration of different ontology modules (referred to as a modular ontology). Here, ontology modules behave as plug-ins i.e. they can plug to or unplug from a modular ontology. The contribution of the thesis will be to clearly define a flexible process for the development of ontologies through the notion of modular ontologies. Ontology modules are well suited for this purpose, since they can be defined such that they are completely independent of each others' signature and underlying description logic language. Furthermore, they can be easily developed by different groups of experts in a distribute environment, while the existence of well-defined and expressive mechanisms allows them to seamlessly communicate, share and exchange their knowledge base.

An interface-based modular ontology formalism is at the core foundation of my proposed architecture, which makes the integration of ontology modules possible. I am currently developing a new interface-based formalism for modular ontologies that can thoroughly satisfy the requirements of the plug-in architecture. Currently, some formalisms such as Distributed Description Logic (Borgida and Serafini 2003), $\mathcal{E}$-connections (Kutz et al. 2004) and semantic import (Bao, Slutzki, and Honavar 2007) have been proposed for modular ontologies. However, some important problems remain open in these proposals. For instance, $\mathcal{E}$-connections requires an assumption that domains of the ontology modules are disjoint, DDL restricts the usage of other modules to only concepts, and in semantic import modules are dependent on each others' signature. I plan to resolve these limitations thorough the interface-based modular ontology formalism.

I intend to shape this formalism in a way that it offers two main contributions. First, the introduction of the notion of interfaces as a set of concepts, roles and inclusion axioms. Each ontology module can use or implement[1] a set of interfaces. Consequently, ontology modules can communicate thorough interfaces and not directly, which is an important step to reduce the tightly-coupled-ness of ontologies in the current formalisms. In addition, interfaces can be exploited to enable the practical integration of modules (*plug-ins*) in a modular ontology. Second, furnishing ontology modules with a black-box behavior such that each module assumes that the current knowledge base of the other modules which implement its necessary interfaces are complete enough (*closed-world assumption*) (Donini et al. 1998). Using this feature, an ontology module can query the knowledge base of the other modules for those concepts and role which are present in its interfaces.

[1] implementation means the provisioning of definitions and assertions for roles and concepts

Research Status

In our prior work (Ensan and Du 2007), we have provided a critical survey of the existing frameworks proposed for the development and maintenance of ontologies. Furthermore in (Ensan and Du 2008b), we have exploited the notion of goals to form a methodology that assists experts in the process of developing ontological frameworks.

The following is the list of the completed work (two first items) and the future plans for my thesis:

(1) I have developed the formal basis for the interface-based modular ontology formalism in (Ensan and Du 2008c). I have shown that the *satisfiability* problem of a modular ontology developed using this formalism is *decidable*. Furthermore, I have shown that the logical consequences of the axioms presented in an interface will propagate to all modules which use that specific interface. It has been further shown that the employment of interfaces in this formalism implies *polymorphism* in the sense that the meaning of a concept or role in a module may be further specialized at configuration time through other realizer modules.

(2) I have studied the various sources of inconsistencies in the proposed formalism and provided models to resolves such cases in (Ensan and Du 2008a). Two sources may lead a modular ontology to become inconsistent: First, when there is an inconsistent module which affects the consistency of the other modules in the modular ontology, and Second, when each module is internally consistent but their integration leads to inconsistencies. For the resolution of the first type of inconsistency, I have enhanced the semantics of the proposed formalism such that it *tolerates* the inconsistent module, and for the second problem, I have presented an algorithm that *resolves* inconsistencies by weakening the imported knowledge from the conflicting interfaces.

(3) One of the major issues in the modular ontology formalism is to investigate how a single or set of changes in an element of an ontology module can effect the other connected modules. This issue is important in modular ontologies, since a modification in a module which implements an interface may change the meaning of that interface for the user modules. Consequently, the semantics of some of the modules will be effected by any incoming change in the composite ontology modules. I am planing to investigate this issue in the proposed interface based formalism. I expect that I be able to derive some initial ideas from the proposed solutions for existing formalisms such as those presented in (Stuckenschmidt and Klein 2003) for DDL . However, the final approach will be completely different due the different underlying expressiveness power of the formalisms as well as the closed-world assumption in the interface-based formalism. For the algorithms that I may develop to handle changes, I also intend to analyze their's time complexity.

(4) Having the interface-based formalism as a foundation, the next ongoing work will be to propose a plug-in architecture for creating modular ontologies. The architecture will probably introduce different layers, each of which is responsible for performing different tasks. There should obviously be a layer dedicated to ontology modules, and also another that is responsible for manipulating plug-ins, formalizing the affect of insertions to and removals from any modules of the modular ontology and also performing the consistency checking and change analysis algorithms.

(5) The last step in my thesis is to implement a software which allows ontology users to design, develop and manipulate modular ontologies in the context of the proposed architecture. The main contribution of such an integrated development environment would be to allow the architecture, formalism, the reasoning and consistency checking algorithms and the mechanism for change analysis to be examined in real-world case studies. I expect that the obtained evaluation results may lead to the need for modifications in the proposed algorithms, especially some algorithms may need more efficient time complexity to be suitable for real-world applications. For example, the current proposed inconsistency resolution algorithm, finds the least number of conflicting interfaces, but it has a rather high time complexity when the number of interfaces are high. I anticipate that this stage would provide me with enough insight to be able to develop algorithms whose time complexity are acceptable for real-world application.

References

Bao, J.; Slutzki, G.; and Honavar, V. 2007. A semantic importing approach to knowledge reuse from multiple ontologies. In *AAAI'07*, 1304–1309.

Borgida, A., and Serafini, L. 2003. Distributed description logics: Assimilating information from peer sources. *J. Data Semantics* 1:153–184.

Donini, F. M.; Lenzerini, M.; Nardi, D.; Nutt, W.; and Schaerf, A. 1998. An epistemic operator for description logics. *Artif. Intell.* 100(1-2):225–274.

Ensan, F., and Du, W. 2007. Towards domain-centric ontology development and maintenance frameworks. In *Interna-tion Conference on Software Engineering and Knowledge Engineering (SEKE'07)*.

Ensan, F., and Du, W. 2008a. Aspects of inconsistency resolving in modular ontologies. In *Canadian Conference on Artificial Intelligence*.

Ensan, F., and Du, W. 2008b. Formalizing the role of goals in the development of domain-specific ontological framew orks. In *Forty-First Hawai'i International Conference on System Sciences (HICSS'08)*.

Ensan, F., and Du, W. 2008c. On the formalization of interface-based modular ontologies under the closed-world assumption. Technical report.

Kutz, O.; Lutz, C.; Wolter, F.; and Zakharyaschev, M. 2004. E-connections of abstract description systems. *Artif. Intell.* 156(1):1–73.

Stuckenschmidt, H., and Klein, M. C. A. 2003. Integrity and change in modular ontologies. In *International Joint Conference on Artificial Intelligence (IJCAI'03)*, 900–908.