Collaborative Ontology Development with Inheritance Support

Boyan Bontchev[1] and Ricardo Jardim-Gonçalves[2]

[1] Chair of Software Engineering, FMI - Sofia University St Kl. Ohidski,
5 J. Bourchier Blvd., Sofia, Bulgaria
bbontchev@fmi.uni-sofia.bg
[2] Head of the UNINOVA Group for Research in Interoperability of Systems
Faculty of Sciences and Technology,
New University of Lisbon,
Quinta de Torre, Monte de Caparica, Portugal
rg@uninova.pt

Abstract. Modern development of huge ontologies in domain of crucial importance is inconceivable without appropriate means for effective collaboration between knowledge stakeholders. The paper presents an approach for collaborative ontology development using inheritance of classes and their properties, relations and metadata. Collaborative actions for maintenance of ontology content are supported by a special annotation ontology facilitating updates, discussions, proposals and voting. The approach is to be incorporated into a platform for collaborative cultural content creation based on ontologies.

Keywords: ontology, collaborative, inheritance, metadata

1 Introduction

Creation of methods and tools for collaborative development of ontologies became a trend in last decade. People appreciated added value gained from collaboration in both ontology construction and maintenance and put effort on development of such platforms and their practical usage. Collaborative Protégé [1] proposes a Web-based extension to the client–server Protégé facilitating ontology development in collaborative mode by using discussions, chats, annotating ontology components and change control. The MENTOR collaboration methodology [2] includes several essential steps as semantic comparisons, basic lexicon establishment, ontology mappings and some other operations to build domain reference ontology. The present paper[1] presents very shortly an implementation of collaborative ontology harmonization allowing better reuse by using inheritance of classes and their properties, relations and metadata. It provides multiuser collaboration support, version management like in MENTOR, assess policies and user control via flexible role management and, as well, usage of annotations supporting iterative collaborative ontology development – for classes, properties, metadata and taxonomy.

2 Collaborative Ontology Construction through Inheritance

According object-oriented paradigm, data properties of class ontology are referred as attributes and object properties are called inter-object relationships. While classes (ontology types) are inherited by sub-classes (sub-types) though IS_A relations, inter-object relationships are presented by so called HAS_A relations. Our approach applies inheritance not only to the class attributes but also to metadata (annotations) of both classes and attributes which makes easier specification and harmonization. Conflicts posed by a multiple inheritance are resolved by explicit choice of the one of the super-classes containing the same attribute (fig. 1.a). For collaborative ontology development through inheritance, an ontology of annotations was created (fig. 1.b).
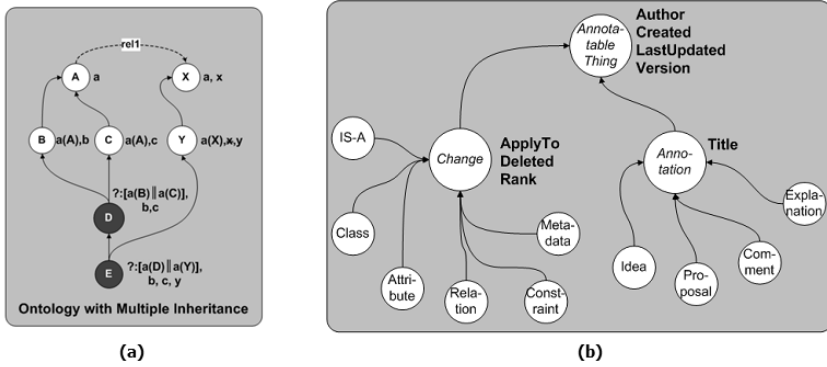

Fig. 1. Multiple inheritance (a) and the proposed ontology for changes and annotations (b).

3 Conclusions

The presented approach for collaborative ontology development using inheritance with resolution of conflicts of multiple inheritance promises benefits for flexible and easier ontology specification, maintenance, merging and reuse of classes and their properties and metadata. The proposed ontology for annotating collaboration actions (for classes, attributes, relations and metadata) and annotations for proposals, comments, voting, etc. supports a flexible multi-user development process.

References

1. Tudorache, T., Noy, N., Vendetti, J., Redmond, T.: Collaborative Ontology Development in Protégé, LNCS, Volume 5318, pp.17-32 (2010).
2. Sarraipa, J., Silva, J., Gonçalves, R., Monteiro, A.: MENTOR – Methodology for Enterprise Reference Ontology Development, 4th Int. Conf. on Intelligent Sys., pp 6.32-6.40 (2008).