DiCoMEF: A Distributed Collaborative Model Editing Framework

Amanuel Koshima  
University of Namur  
PReCISE Research Center, Belgium  
ammanuel.koshima@unamur.be

Vincent Englebert  
University of Namur  
PReCISE Research Center, Belgium  
victor.englebert@unamur.be

Abstract

Domain specific modeling languages have matured over-time and are widely used, but facilities for collaborative modeling are still limited. Hence, there is a demand for tool support that facilitates sharing of modeling artifacts (collaboration), conflict detection, reconciliation, and merging of concurrently edited models. This paper presents a collaborative model editing framework called DiCoMEF.

1 Introduction

Domain Specific Modeling Languages (DSML) have matured over time and widely used as an efficient alternative to General Purpose Modeling languages (e.g., UML, Petri Nets) for modeling complex systems [2]. DSML describes concepts at different levels of abstraction using models, meta-models and meta-meta-models. A model is an abstraction of a software system and a meta-model is a language that describes concepts and constraints of models. A meta-meta-model is a minimum set of concepts which defines languages (e.g. EMF/Ecore 1).

Several metaCASE tools have been developed to support the use of models defined with DSML, but most of these tools do not support collaboration [3]. Nevertheless, whenever the complexity of a problem increases, the diversity of users in groups increases as well [4]. Hence, methods and tools to facilitate the cooperative work of these users are required. A model version control system, which facilitates sharing of modeling artifacts, conflict detection, merging, and versioning of models, is then required. More particularly, conflicts (textual, syntactic, or semantic [1]) that cause inconsistency should be identified and resolved.

Different approaches have been adopted to ensure collaboration, for instance, a central repository (e.g. SVN) with locking techniques and merging mechanism. However, the locking technique is not scalable [1] and this approach forces all users to be dependent on a central repository. This might introduce unnecessary access right bureaucracy that would lead to dissatisfaction among members. There is another mode of collaborative work where each member has his/her local copy of a shared modeling artifact and works in isolation with other members. This approach provides users a better control of modeling artifacts and mitigates the problem of being dependent on a central server. However, it is challenging to keep all copies of concurrently edited models consistent. Most of available versioning tools detect conflicts in text or tree based documents, but they are not suitable for models that have a graph based nature [1].

2 DiCoMEF Framework

DiCoMEF is a distributed collaborative model editing framework based on EMF/Ecore meta-model and it lets each member of the group having his/her local copy of (meta)model (see Figure 1) [3]. In DiCoMEF, modifications are controlled by human agents (controllers). They manage the evolution of either models or meta-models depending of their role in the cooperative work. Model (resp. meta-model) controller roles are flexible meaning that they can be assigned (delegated) to other members of a group as long as there is one unique coordinator per group. Whenever members of a group modify (meta-)models locally, elementary change operations (create, delete, and update) are stored locally. DiCoMEF extends a history meta-model of EDAPT 2 to capture edit operations. These operations are used later as a means of communication among members of a group. Peer-to-peer communication could hinder convergences of all copies of (meta)models.

DiCoMEF relies on two concepts (main-line and branches) to ensure the communication framework. The main-line stores different versions of a copy (meta-)model locally at each editors site. Editors cannot modify (meta-)models stored on the main-line; they can only adapt those stored on the branch and send then their local modifications.

---

1http://www.eclipse.org/modeling/emf/

2http://www.eclipse.org/edapt/
with master students to assess: (1) the feasibility of collaborative methods and processes with DiCoMEF, (2) the correctness of conflict detection mechanisms (recall and precision), (3) the usability of the merge tool and DiCoMEF framework (4) measuring user efforts (time) needed to merge concurrently edit meta-models manually and using DiCoMEF merge tool. Overall result of the evaluation was positive and it showed that DiCoMEF can be used for collaborative meta-modeling.

4 Future Work

The reconciliation process of DiCoMEF will be improved by letting users work with concrete syntax editors and meta-model semantics (i.e., OCL rules for instance). Besides, more advanced collaborative workflows will also be investigated and defined on top of DiCoMEF. In addition, we plan to conduct more experiments and evaluations in the future.

References