Chapter 9
Metamodels Construction Based on the Definition of Domain Ontologies

Carlos Enrique Montenegro-Marin
Universidad Distrital "Francisco José de Caldas", Colombia

Rubén González Crespo
Universidad Pontificia de Salamanca, Spain

Oscar Sanjuán Martínez
Universidad Carlos III de Madrid, Spain

Juan Manuel Cueva Lovelle
Universidad de Oviedo, Spain

B. Cristina Pelayo García-Bustelo
Universidad de Oviedo, Spain

Patricia Ordóñez de Pablos
Universidad de Oviedo, Spain

ABSTRACT

This chapter proposes a mechanism for mapping domain ontologies to metamodels by a direct mechanism; this proposal is necessary because there is no formal mechanism for obtaining requirements in model driven engineering. Specifically, here the authors propose the use of a domain ontology as the main input for defining metamodels. They define a point in common between domain ontologies and metamodels to apply a method of direct conversion between domain ontology and the metamodel. At the end of the chapter, the authors present a real case study in which they use the technique described and the conclusions of the investigation.

INTRODUCTION

One of questions of model-driven engineering is how to start building a metamodel. The formal part of the requirements engineering is so important in any software development process and the whole project in general. Many authors have raised the possibility of using ontologies to meet this need. We propose the use of a domain ontology as the main input for defining metamodels.

Papers such as "Bridging Metamodels and Ontologies in Software Engineering" in which the authors present and investigate the literature on both metamodelling and ontologies in order to identify ways in which they can be made compatible and linked in such a way as to ben-

DOI: 10.4018/978-1-4666-2494-8.ch009

efit both communities and create a contribution to a coherent underpinning theory for software engineering. Analysis of a large number of theoretical and semi-theoretical approaches using as a framework a multi-level modelling construct identifies strengths, weaknesses, incompatibilities, and inconsistencies within the extant literature. A metamodel deals with conceptual definitions while an ontology deals with real-world descriptors of business entities and is thus better named "domain ontology." A specific kind of ontology (foundational or high-level) provides "metalevel" concepts for the domain ontologies. In other words, a foundational ontology may be used at the same abstraction level as a metamodel and a domain ontology at the same abstraction level as a (design) model, with each pair linked via an appropriate semantic mapping (Henderson-Sellers, 2011), and "Ontologies, Meta-Models, and the Model Driven Paradigm" that presents meta-modelling hierarchy that is aware of ontologies—that is, an ontology-aware mega-model of MDE. Based on the insight of that the main difference of models and ontologies lies in their descriptive prescriptiveness, the role of ontologies in this meta-pyramid is to describe the existing world, the environment, and the domain of the system (analysis), while the role of system models is to specify and control the system under study itself on various levels of abstraction (design and implementation). Consequently, in this scheme, MDE starts from ontologies, refines, and augments them towards system models, respecting their relationships to prescriptive models on all metalevels (Aßmann, Zschaler, & Wagner, 2006), are the main basis for this proposal.

This chapter presents the reader a formal method to map an ontology in a metamodel. However, more importantly it also presents a methodology for obtaining a domain ontology and with the domain ontology is shown how map it to a metamodel of M1 or M2 level. In addition, it shows a scenario where this methodology was used to obtain an ontology of LMS platforms which was then transformed to a metamodel to finally make a visual domain-specific-language—DSL. With this, we use the DSL for modeling a course to be deployed on any platform specific LMS.

The chapter is organized as follows: first there are some basics on Ontologies, then presents a chapter that describes how ontologies relate to metamodels, then explain the concepts of Model Driven Engineering—MDE—and metamodels, then addresses how to map an ontology to a metamodel, and finally presents a case study and the conclusions.

ONTOLOGY CONCEPTS

Much information is found regarding the issue of ontology; for this reason, the definitions in the area of ontologies are taken from *Ontology Development 101: A Guide to Creating Your First Ontology* (Natalya & Deborah, 2005). It is official guidance provided by Stanford University and founder of Protege (Stanford Center for Biomedical Informatics Research, 2010), which is the most used tool in the creation of ontologies.

The Semantic Web literature contains several definitions of ontology. In addition, many definitions contradict others definitions. In order to standardize definitions in this chapter, an ontology is a formal and explicit concepts in a specific domain (classes [also called concepts]), properties of these concepts that contain various features and attributes (slots [also called roles or properties]). Finally, also have restrictions on the properties (facets [also called role restrictions]). An ontology together with a group of individuals of classes constitutes a knowledge base. Actually, there is a fine line where the ontology ends and the knowledge base begins. Classes are the focus of most ontologies. And the classes describe concepts in a domain (Natalya & Deborah, 2005).

Related Content