Chapter 50
Supporting Domain Ontology through a Metamodel:
A Disaster Management Case Study

Siti Hajar Othman
Universiti Teknologi Malaysia, Malaysia & University of Wollongong, Australia

ABSTRACT

A metamodel is a model that has the ability to create the languages of many domain models. Domain models are conceptual models of a domain under study and contain all the entities, attributes, relationships, and constraints of the domain. As the artifact of a metamodeling technique, a metamodel could generalize most of the concepts used in existing domain models by unifying the views and structuring the language of the domain. In relation to ontology, the creation of a metamodel could assist in understanding, structuring, and analyzing the ontology. Other than its potential to engineer new ontology and re-engineer existing ontology, a metamodel can also be used to facilitate communication among communities regarding the ontology. The authors present how a metamodel can structure and manage knowledge of a domain it models. Through the Disaster Management Metamodel, they create a language for the disaster management domain.

INTRODUCTION

Various kinds of models have been found useful in many domains such as in management, mathematics, engineering, philosophy, and medical fields. A model is an abstract representation of a domain in the real world and has two key elements: a *concept* that characterizes things, and a *relationship* that characterizes links between these things in the domain. A model is powerful because it can help manage many complexities (Levendovszky, Rumpe, Schatz, & Sprinkle, 2010). Constructing a model is not an easy task. Observations have shown that the hardest task in the development of models is how the model can be developed with the concepts and notations that are truthful to the language of the models. Modeling Language

DOI: 10.4018/978-1-4666-4707-7.ch050

(ML) is a notion that specifies all the elements with which any domain model can be described. It is a language description, which includes the syntax (the notation) and semantics (the meaning) of each element to enable clear representation of recurring domain problems. According to Clark, Evans, and Kent (2002), the abstract syntax is a model of the legal sentences of the language and the semantic domain is a model of the legal meanings that sentences can take. The use of ML enables domain users to abstract and share knowledge, and makes it easier to solve problems using models as starting points. To describe the structure, behaviour, and properties of models, a special model that is capable of describing the language of these models is required. In addition, to make this special model able to describe the language of models, the model itself must reside at a level higher than the collection of models it wants to represent. A metamodel is an artifact that has the ability to represent the ML. It is a technique, which normally is used in the software engineering field. Its aim is to create interoperable, reusable, portable software activities and components. The creation of a metamodel is done by using an ML called metamodeling (Karagiannis & Kuhn, 2002). Its main task is to describe ML with classes for each of the concepts used in the metamodel.

Ontology, on the other hand, is a philosophical study dealing with models of reality (what exists). In the context of knowledge sharing, ontology is defined as a specification of a conceptualization (Gruber, 1995). In relation to a metamodel, thorough analysis of all elements, relationships and attributes of the ontology is required for the purpose of designing a metamodel (Davies, Green, Minton, & Rosemann, 2005). In the Artificial Intelligence view of ontology, ontologies differ among different domains (Jeusfeld, Jarke, & Mylopoulos, 2009). In this chapter, we present how a thorough analysis as specified by Davies et al. (2005) could be performed through the creation of our metamodel. In our context, we analyze the domain ontology by performing the detailed analysis of the models of our studied domain. Our studied domain is Disaster Management (DM). The result of the analysis of the DM models is the Disaster Management Metamodel (DMM). DM is a very complex and challenging domain to model. By providing the means to tailor a model to a particular domain and the means to specify the ML, the creation of the DMM can offer many benefits to the users of DM. Therefore, the aim of this research is to make a clear and principled analysis of the DM. As a result, the DMM can manage the complexities of many DM activities and make the coordination of DM operations easily accessible by DM users at all levels.

In the creation of the DMM, all the important DM concepts and relationships that we found in all the investigated DM models are derived and analyzed. The metamodel is the artifact that can contain the specification of the modeling environment of a certain domain. Therefore, it can define the syntax and semantics of the domain and can represent all the systems in the domain it modelled (Nordstrom, Sztipanovits, Karsai, & Ledeczi, 1999). Thus, it provides the context needed for expressing the well-formedness constraints for domain models (Paige, Brooke, & Ostroff, 2007). Without the metamodel, the syntax and semantics of the domain models will be ambiguous. Normally, the metamodel could be viewed from three (3) different perspectives:

1. As a set of building blocks and rules used to build new models,
2. As a model of a domain of interest, and
3. As an instance of another model.

DM is the management of risks and consequences of a disaster which includes all aspects of planning and responding to all phases of a disaster: these phases include mitigation, preparedness, response and recovery activities (W3C Incubator Group, 2008). Managing disasters is difficult because the domain often depends on various types of information systems such as modeling, simula-