Chapter II
Essential, Mandatory, and Shared Parts in Conceptual Data Models

Alessandro Artale
Free University of Bozen-Bolzano, Italy

C. Maria Keet
Free University of Bozen-Bolzano, Italy

ABSTRACT

This chapter focuses on formally representing life cycle semantics of part-whole relations in conceptual data models by utilizing the temporal modality. The authors approach this by resorting to the temporal conceptual data modeling language $\mathcal{ER}_{VT}$ and extend it with the novel notion of status relations. This enables a precise axiomatization of the constraints for essential parts and wholes compared to mandatory parts and wholes, as well as introduction of temporally suspended part-whole relations. To facilitate usage in the conceptual stage, a set of closed questions and decision diagram are proposed. The long-term objectives are to ascertain which type of shareability and which lifetime aspects are possible for part-whole relations, investigate the formal semantics for sharability, and how to model these kind of differences in conceptual data models.

INTRODUCTION

Modeling part-whole relations and aggregations has been investigated and experimented with from various perspectives and this has resulted in advances and better problem identification to a greater or lesser extent, depending on the conceptual modeling language (Artale et al., 1996a; Barbier et al., 2003; Bittner & Donnelly, 2005; Borgo & Masolo, 2007; Gerstl & Pribbenow, 1995; Guizzardi, 2005; Keet, 2006b; Keet & Artale, 2008; Motschnig-Pitrik & Kaasbøll, 1999; Odell, 1998; Sattler, 1995). Nowadays, part-whole relations receive great attention both in conceptual modeling community (e.g., the Unified Modeling Language, UML, the Extended Entity Relationship, EER, and the Object-Role Modeling, ORM) as well as in the semantic web community (e.g. the Description Logic based language OWL).

Several issues, such as transitivity and types of part-whole relations, are being addressed successfully with converging approaches from an ontological, logical, and/or linguistic perspectives (Borgo & Masolo, 2007; Keet & Artale, 2008; Varzi, 2004; Vieu & Aurnague, 2005). On the other hand, other topics, such as horizontal relations among parts and *life cycle semantics* of parts and wholes, still remain an open research area with alternative and complimentary approaches (Bittner & Donnelly, 2007; Guizzardi, 2005; Motschnig-Pitrik & Kaasbøll, 1999). For instance, how to model differences between an Information System for, say, a computer spare parts inventory compared to one for transplant organs? Indeed, organs are at the time before transplantation not on the shelf as are independently existing computer spare parts, but these organs are part of another whole and can only be part of another whole *sequentially*. For a university events database, one may wish to model that a seminar can be part of both a seminar series and a course, *concurrently*. Another long-standing issue is how to represent essential versus mandatory parts and wholes (Artale *et al*., 1996a). The solution proposed in Guizzardi (2005) as an extension to UML class diagrams is not easily transferable to other modelling/representation languages.

In this chapter we study representation problems related to the notion of *sharability* between parts and wholes. In particular, we are interested in representing that parts (i) cannot be shared by more than one whole; (ii) cannot exist without being part of the whole; (iii) can swap wholes in different ways. Clearly, these rich variations in shareability of parts cannot be represented in any of the common, industry-grade, UML class diagram, EER, or ORM CASE tools. In order to reach such a goal, we take a fist step by aiming to answer these main questions:

- Which type of sharability and which lifetime aspects are possible?
- What is the formal semantics for sharability?
- How to model these kind of differences in a conceptual data model?

To address these questions, we merge and extend advances in representing part-whole relations as in UML class diagrams with formal conceptual data modeling for temporal databases (temporal EER) and ORM's usability features. The various shareability constraints are reworded into a set of modeling guidelines in the form of closed questions and a decision diagram to enable easy navigation to the appropriate sharability case so as to facilitate its eventual integration in generic modeling methodologies.

Concerning the formalization of the sharability notion and the relationships between the lifespans of the involved entities, we use the temporally extended Description Logic $\mathcal{DLR}_{\mathcal{US}}$ (Artale *et al*., 2002). Indeed, while DLs have been proved useful in capturing the semantics of various conceptual data models and to provide a way to apply automatic reasoning services over them (Artale *et al*., 2007a; Berardi *et al*., 2005; Calvanese *et al*., 1998b, 1999; Franconi & Ng, 2000; Keet, 2007), temporal DLs have been applied to the same extent for temporal conceptual models (Artale & Franconi, 1999; Artale *et al*., 2003, 2002, 2007b). The formalization we present here is based on the original notion of *status relations* that captures the evolution of a relation during its life cycle. Furthermore, a set of $\mathcal{DLR}_{\mathcal{US}}$ axioms are provided and proved to be correct with respect to the semantic we provide for each particular sharability relation.

The remainder of the chapter is organised as follows. We start with some background in

Related Content

Measurement in Software Engineering: The Importance of Software Metrics
Ruya Samli, Zeynep Behrin Güven Aydnand Uur Osman Yücel (2020). *Applications and Approaches to Object-Oriented Software Design: Emerging Research and Opportunities (pp. 166-182).*
www.irma-international.org/chapter/measurement-in-software-engineering/249325

Architecture Description Languages for the Automotive Domain
Sebastien Faucou, Francoise Simonot-Lionand Yvon Trinquet (2010). *Behavioral Modeling for Embedded Systems and Technologies: Applications for Design and Implementation (pp. 353-376).*
www.irma-international.org/chapter/architecture-description-languages-automotive-domain/36349

Framework for Reusable Test Case Generation in Software Systems Testing
Kamalendu Pal (2020). *Software Engineering for Agile Application Development (pp. 212-229).*
www.irma-international.org/chapter/framework-for-reusable-test-case-generation-in-software-systems-testing/250444

Modelling Information Demand in an Enterprise Context: Method, Notation, and Lessons Learned
Magnus Lundqvist, Kurt Sandkuhland Ulf Seigerroth (2013). *Frameworks for Developing Efficient Information Systems: Models, Theory, and Practice (pp. 77-98).*
www.irma-international.org/chapter/modelling-information-demand-enterprise-context/76619

Applying Niche Theory to Measure Uses and Gratifications of Social Media in Malaysia
Ha Jin Hwang, Haeng Kon Kim, Monowar Mahmoodand Norazryana Mat Dawi (2022). *International Journal of Software Innovation (pp. 1-11).*
www.irma-international.org/article/applying-niche-theory-to-measure-uses-and-gratifications-of-social-media-in-malaysia/289594