Enriched Support for Ring Constraints

Terry Halpin[1] and Matthew Curland[2]

[1] LogicBlox, Australia and INTI International University,
Malaysia
[2] LogicBlox, USA
{terry.halpin,matt.curland}@logicblox.com

Abstract. Fact-oriented modeling approaches such as Object-Role Modeling (ORM) have long supported several varieties of ring constraints, such as irreflexivity, asymmetry, intransitivity, and acyclicity, on pairs of compatible roles. The latest version of the Web Ontology Language (OWL 2) supports five kinds of ring constraint on binary predicates. Recently, three more ring constraint types (local reflexivity, strong intransitivity, and transitivity) were added to ORM. This paper discusses these new additions to ORM, as implemented in the Natural ORM Architect (NORMA) tool, and identifies important ways in which ORM and OWL differ in their support for ring constraints, while noting different mapping alternatives. We determine which combinations of elements from this expanded set of ring constraints are permitted, and provide verbalization patterns for the new additions. Graphical shapes for the new constraints and constraint combinations are introduced and motivated, and NORMA's new user interface for entry of ring constraints is illustrated.

1 Introduction

Data modeling approaches have long recognized the importance of integrity constraints to help ensure that data entered into the information system is consistent with the business domain being modeled. At the implementation level, such constraints are captured in code in a language relevant to that logical data model. For example, in a relational database the constraints are coded in SQL, in a semantic Web ontology the constraints might be coded in the *Web Ontology Language* (*OWL*) [20], and in a deductive database the constraints might be declared in datalog [1]. However, to ensure that the constraints enforced in the implementation code correspond to actual constraints in the universe of discourse, the constraints should first be specified in a conceptual schema, where they can be reliably validated with the business domain experts using concepts and language that are intelligible to them, and then transformed automatically into implementation code.

One class of constraints that is commonly found in business domains but less often captured in industrial data models deals with logical restrictions on how *pairs of compatible fact roles* may be populated. These constraints are called *ring constraints*, since in the most common case the role pairs form a *ring predicate*, where each role in the relationship is played by instances of the same type (see Figure 1(a)). Here a path from the type *A* through the predicate *R* and back to *A* intuitively forms a ring.

R. Meersman, T. Dillon, and P. Herrero (Eds.): OTM 2011 Workshops, LNCS 7046, pp. 309–318, 2011.