



Chapter XVII

**Dealing with Relationship
Cardinality Constraints in
Relational Database
Design**

Dolores Cuadra Fernández
Universidad Carlos III de Madrid, Spain

Paloma Martínez Fernández
Universidad Carlos III de Madrid, Spain

Elena Castro Galán
Universidad Carlos III de Madrid, Spain

ABSTRACT

Conceptual models are well-known tools to achieve a good design of information systems. Nevertheless, the understanding and use of all the constructs and constraints which are presented in such models are not an easy task and sometimes it is cause of loss of interest.

In this chapter we have tried to study in depth and clarify the meaning of the features of conceptual models. The disagreements between main conceptual models, the confusion in the use of some of their constructs and some open problems in these models are shown. Another important topic treated in this chapter is the conceptual-to-logic schemata transformation process.

Some solutions are presented in order to clarify the relationship construct and to extend the cardinality constraint concept in ternary relationships. How to preserve the cardinality constraint semantics in binary and ternary relationships for their implementation in a DBMS with active capabilities has also been developed.

INTRODUCTION

Database modeling is a complex task that involves conceiving, understanding, structuring and describing real Universes of Discourse (UD) through the definition of schemata using abstraction processes and data models. Traditionally, three phases are identified in database design: conceptual, logical and physical design. The conceptual modeling phase represents the most abstract level since it is independent of any Database Management System (DBMS) and, consequently, it is very close to the user and allows him to collect almost completely the semantics of the real world to be modeled.

A conceptual schema, independently of the data formalism used, plays two main roles in the conceptual design phase: a *semantic* role, in which user requirements are gathered together and the entities and relationships in a UD are documented, and a *representational* role that provides a framework that allows a mapping to the logical design of database development. Three topics are involved in the database conceptual modeling process: data modeling formalism, methodological approach and CASE tool support. One of the most extended data modeling formalisms, the Extended Entity Relationship (EER) model has proven to be a precise and comprehensive tool for representing data requirements in information systems development, mainly due to an adequate degree of abstraction of the constructs that it includes. Although the original ER model was proposed by Chen (1976), many extensions and variations as well as different diagrammatic styles have been defined (Hull & King, 1987; McAllister, 1998; Peckhan & Maryanski, 1988).

In database conceptual analysis, among the most difficult concepts to be modeled are relationships, especially higher-order relationships, as well as their associated cardinalities. Some textbooks (Boman et al., 1997; Ullman & Widom, 1997) assume that any conceptual design can be addressed by considering only binary relationships since its aim is to create a computer-oriented model. We understand the advantages of this approach although we believe that it may produce certain loss of semantics (some biases are introduced in user requirements) and it forces one to represent information in rather artificial and sometimes unnatural ways.

Concerning the logical design, the transformation process of conceptual schemata into relational schemata should be performed trying to completely preserve the semantics included in the conceptual schema; the final objective is to keep the semantics in the database itself and not in the applications accessing the database. Nevertheless, sometimes a certain loss of semantics is produced, for instance, foreign key and not null options in the relational model are not enough to control ER cardinality constraints.

This chapter is devoted to the study of the transformation of conceptual into logical schemata in a methodological framework focusing on a special ER construct: the relationship and its associated cardinality constraints. The section entitled “EER Model Revisited: Relationships and Cardinality Constraint” reviews the relationship and cardinality constraint constructs through different methodological approaches in order

28 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: www.igi-global.com/chapter/dealing-relationship-cardinality-constraints-relational/9217

Related Content

Comprehension of Hierarchical ER Diagrams Compared to Flat ER Diagrams

Revital Danoch, Peretz Shoval and Mira Balabaan (2005). *Information Modeling Methods and Methodologies: Advanced Topics in Database Research* (pp. 241-257). www.irma-international.org/chapter/comprehension-hierarchical-diagrams-compared-flat/23017

Transformation-Based Database Engineering

Jean-Luc Hainaut (2005). *Encyclopedia of Database Technologies and Applications* (pp. 707-713). www.irma-international.org/chapter/transformation-based-database-engineering/11228

Object-Relational Modeling in the UML

Jaroslav Zendulka (2005). *Encyclopedia of Database Technologies and Applications* (pp. 421-426). www.irma-international.org/chapter/object-relational-modeling-uml/11183

Evaluation of Component-Based Development Methods

Nicky Boertien, Maarten W.A. Steen and Henk Jonkers (2005). *Information Modeling Methods and Methodologies: Advanced Topics in Database Research* (pp. 323-343). www.irma-international.org/chapter/evaluation-component-based-development-methods/23021

Agile Modeling, Agile Software Development, and Extreme Programming: The State of Research

John Erickson, Kalle Lyytinen and Keng Siau (2005). *Journal of Database Management* (pp. 88-100). www.irma-international.org/article/agile-modeling-agile-software-development/3343