Atomicity and Semantic Normalization

Andy Carver, INTI Education Group, Malaysia

Terry Halpin, LogicBlox, Australia, & INTI Education Group, Malaysia

ABSTRACT

This paper contrasts two different approaches to designing relational databases that are free of redundancy. The Object-Role Modeling (ORM) approach captures semantics in terms of atomic (elementary or existential) fact types, before grouping the fact types into relation schemes. Normalization by decomposition instead focuses on “non-loss decomposition” to various, and progressively more refined, “normal forms”. Traditionally, non-loss decomposition of a relation requires decomposition into smaller relations that, upon natural join, yield the exact original population. Non-loss decomposition of a table scheme (or relation variable) requires that the decomposition of all possible populations of the relation scheme is reversible in this way. This paper shows that the dependency requirement for “all possible populations” is too restrictive for definitions of multi-valued and join dependencies over relation schemes. By exploiting ORM modeling heuristics, the authors offer new definitions of these data dependencies and non-loss decomposition, to enable these concepts to be addressed at a truly semantic level.

Keywords: Data Modeling, Database Schema, Non-loss Decomposition, Normal Forms, Normalization, Object Role Modeling, Relational Database

INTRODUCTION

In relational database design, being able to achieve a fully normalized schema is generally considered desirable, mainly because relations are then guaranteed to be free of redundancy, thus simplifying the process of maintaining consistency as the database is updated. The acceptance of that value-premise is in fact the starting point of the current paper. The question which this paper addresses is not, whether we need a procedure for producing normalized relation schemes, but rather, which procedure is both effective and most appropriate for achieving this desired result1.

The question does not have an obvious answer: indeed, various approaches are recommended. Conceptual data modeling approaches such as Entity-Relationship Modeling (ER) and Object-Role Modeling (ORM) use a two phase process: conceptualization, in which information is first portrayed in terms of conceptual schemas suitable for communication with domain experts (Griethuysen, 1982), and then de-conceptualization where these structures are mapped into relational schemas.

In contrast, the normalization approach to database design ignores conceptualization,
instead representing information directly in terms of relational database structures, such as relation schemes (i.e., relation variables) and various dependencies. This paper’s treatment of normalization focuses on normalization by decomposition, ignoring normalization by synthesis. Normalization by decomposition basically follows a process of achieving progressively higher levels of normalization (called “normal forms”) through “non-loss decomposition” of given relational table schemes. Just how the original tables became “given” in the first place, the procedure does not say.

The ER approach captures data in terms of entities, attributes, and relationships, and applies a mapping procedure to transform these structures into a relational database schema (e.g., Batini, Ceri, & Navathe, 1992). The ORM approach captures information in terms of atomic fact types, and then applies an algorithm such as Rmap to map these fact types and associated constraints into a relational schema (Ritson & Halpin, 1993; Halpin & Morgan, 2008). ORM is a prime example of the fact-oriented modeling approach, which uses the fact type (relationship type) as its sole data structure. Features modeled as attributes in ER (e.g., Person.birthdate) are modeled in ORM as relationships (e.g., Person was born on Date). Other examples of fact-orientation include Natural language Information Analysis Method (NIAM) (Verheijen & Bekkum, 1982) and the Predicator Set Model (PSM) (Hofstede, Proper, & Weide, 1993). Overviews of ORM may be found in (Halpin, 2006; Halpin, 2007; Halpin, 2010), and a detailed coverage in (Halpin & Morgan, 2008).

Non-loss decomposition of a relation traditionally requires decomposition into smaller relations that, upon natural join, yield the exact original population. Non-loss decomposition of a table scheme requires that the decomposition of all possible populations of the relation scheme is reversible in this way. In this paper we show that the dependency requirement for “all possible populations” is too restrictive for definitions of multi-valued and join dependencies over relation schemes. Unlike ORM’s conceptual-schema-design-and-relational-mapping procedure, the traditional normalization procedure neither seeks, nor invokes the concept of, “atomic” fact types, and this is the source of its problem. By exploiting the fact-oriented nature and modeling heuristics of ORM, we offer better, more accurate definitions of these data dependencies, and of “non-loss decomposition”, thus enabling these concepts to be addressed at a truly semantic level. We do not attempt to present a detailed normalization procedure.

The next section reviews the traditional notions of non-loss decomposition and data dependency in normalization theory. The section after that illustrates the failure of the accepted definitions of multi-valued dependency and 4th normal form. The subsequent section solves these problems by defining a semantic notion of non-loss decomposition, and applies this notion to define semantic versions of multi-valued and join dependencies that overcome the defects in the commonly accepted notions of 4th and 5th normal form. The final section summarizes the main contributions.

THE TRADITIONAL VERSION OF THE NORMALIZATION PROCEDURE

As mentioned above, the traditional version of the normalization procedure comprises a process of achieving progressively higher levels of normalization (called “normal forms”) through “non-loss decomposition” of given relational table schemes. In mathematics, the term “relation” means a set of ordered n-tuples. In relational databases (Codd, 1970), tuples are ordered by name (by pairing value entries with their attribute name). By the Principle of Extensionality, sets are determined by their membership, and hence are fixed (unchanging). So in relational databases a relation is basically a table population (fixed set of tuples).

In contrast, a relation scheme is a table structure or table variable, whose population may vary at different times. A relation scheme
Related Content

**Integrating Patient Consent in e-Health Access Control**
[www.igi-global.com/article/integrating-patient-consent-health-access/55267?camid=4v1a](www.igi-global.com/article/integrating-patient-consent-health-access/55267?camid=4v1a)

**Mitigating Type Confusion on Java Card**
[www.igi-global.com/article/mitigating-type-confusion-java-card/77915?camid=4v1a](www.igi-global.com/article/mitigating-type-confusion-java-card/77915?camid=4v1a)
A Question for Research: Do We mean Information Systems or Systems of Information?
www.igi-global.com/chapter/question-research-mean-information-systems/38171?camid=4v1a

A Formal Language for XML Authorisations Based on Answer Set Programming and Temporal Interval Logic Constraints
www.igi-global.com/article/formal-language-xml-authorisations-based/52594?camid=4v1a