ISSN: 2393-8528

Contents lists available at www.ijicse.in

International Journal of Innovative Computer Science & Engineering

Volume 1 Issue 2; Page No. 22-23

Managing Relational data bases with Normalization

Rajeev Kumar Yadav

Research scholar, Sam Higginbottom Institute of Agriculture, Technology and Sciences, Allahabad, Uttar Pradesh

Salam.rajeev@gmail.com

ARTICLE INFO

Received 09 Oct. 2014
Accepted 20 Nov. 2014

Corresponding Author:

Rajeev Kumar Yadav
1 Research scholar, Sam Higginbottom Institute of Agriculture, Technology and Sciences, Allahabad, Uttar Pradesh
E-mail: Salam.rajeev@gmail.com

ABSTRACT

The traditional vertical decomposition methods in relational database normalization fail to prevent common data anomalies. Although a database may be highly normalized, the quality of the data stored in this database may still deteriorate because of potential data anomalies. In this paper, we first discuss why practitioners need to further improve their databases after they apply the traditional normalization methods, because of the existence of functional entanglement, a phenomenon we defined. We outline two methods for Identifying functional entanglements in a normalized database as the first step toward data quality improvement. The goal of this paper is to reveal shortcomings of the traditional database normalization methods with respect to the prevention of common data anomalies, and offer practitioners useful techniques for improving data quality.

1. INTRODUCTION

Database designers create and work with relational databases on a regular basis. However, these practitioners can face numerous problems when building a database using tables. One common problem is data redundancy, which occurs when data are duplicated in a database table, or relation. These duplicated data can cause anomalies that affect data quality and provide users with incorrect information. Therefore, practitioners must follow certain rules while designing and normalizing their databases. The traditional method of preventing data redundancy and the resulting anomalies is called "database normalization." In the normalization process, practitioners examine functional dependency, multi-valued dependency, project-join dependency, and several other data dependencies to decompose a relation into multiple relations. The end result of normalization is a set of relations that meet the requirements of different levels of the normal form. The higher the level of the normal form we reach in a database, the lower the possibility that data anomalies can occur. In this paper, we continue to build on this research findings. We first discuss methods of identifying functional entanglements in a normalized database. Then we analyze several practical approaches for restricting the potential effects of these functional entanglements. The discussion provides practitioners with tools they can use to improve database design and implementation, extending what is typically done during the traditional normalization process.

1.1 Functional dependency in relational data bases normalization:

It suggests that normal form definition based solely on functional dependency will not eliminate some basic data redundancies. It suggests that normal form definitions based solely on Functional Dependency will not eliminate some basic data redundancies. As background, in the relation R with the 2 attributes X and Y, Y is functionally dependent on X (X→Y) if only 1 Y can be determined from 1 X.

The Zip Code to City relation (X→Y) is provided as an example that fails this functional dependency, since a zip code can relate to 1 or more cities. However, the relation exhibits characteristics of functional dependency. There are subsets of each domain X and Y in the relation that are functionally dependent. For example, some zip codes do belong to 1 and only 1 city. For this reason the authors claim a Sub-Domain Dependency exists.

The authors conclude by suggesting that instances where such unavoidable violations were not caught during database design can be overcome with programmed restrictions or database management techniques.

1.2 Ternary Relationship Decomposition and Higher Normal From Structures Derived from Entity Relationship Conceptual Modeling:

It seeks to prove that although ternary relationships involve composite keys, their decomposition into simpler binary forms should not follow the decomposition of composite key relations into higher normal forms. As background, Ternary relationships can be expressed as 1:1:1, 1:1:M, 1:M:N or M:N:P where there functional dependencies are defined by which 2 entities, if any, act as determinants for a 3rd entity. The authors also describe a Semantically Constrained Binary (SCB) relationship that can independently occur between 2 participating entities in a ternary relationship. **For example:** a binary relationship between the 1st and 2nd entity can be M:N whereas the ternary cardinality between the 1st, 2nd and 3rd entities is M:1:N.

2. Identifying Functional Entanglements:

We discuss two methods for identifying some of the most common functional entanglements. The first method is based on detection of “sub domain dependencies.

a. Detecting Sub domain Dependencies:

The cause of the data redundancy problem in the *Employees* relation (Table 4) is a functional dependency relationship between a domain subset of *Title* ({Mr, Ms}) and *Sex* ({M, F}). This phenomenon is called a sub domain dependency and defined formally as follows:

Given a relation R, field Y of R is functionally dependent on field X of R in subdomain (noted as X →s Y) if and only if, 1. X → Y does not hold, and 2. There exists at least one instance x*i*∈Domain(X) so that x*i* is associated with one and only one Y-value in R.1

b. Identifying Restricted Domains:

Functional entanglements can appear in forms other than sub domain dependency. To illustrate these functional entanglements, we first introduce two terms related to field domain—“unrestricted domain” and “restricted domain.” Specifically, for any given field, if all possible values in its domain can be assigned to any record in this field, we call the domain of this field an unrestricted domain; otherwise, we call it a restricted domain. Analyzing the domain of each field in a relation can help root out functional entanglements. If database designers detect any fields with restricted domains caused by the values of another field, then they need to provide an extra mechanism to prevent data anomalies. We can look at another example to illustrate how analyzing the domains of fields can help identify functional entanglements in a relation. So far, in all our examples, functional entanglements among fields appear within the same relations. Sometimes, however, similar functional entanglements can come from different relations, as illustrated by considering salary information as yet another part of the hypothetical eHR system.

3. Practical Approaches for Preventing Data:

Three practical methods for preventing data anomalies caused by functional entanglements. We summarize them briefly below, explore their applicability, and discuss their strengths and shortcomings in dealing with different types of functional entanglement. Following that discussion, we introduce and analyze another practical method that can be applied to prevent data anomalies.

a. Preventing Data Anomalies by Changing Relation Design:

After having normalized a database into BCNF or even 5NF with the vertical decomposition method, database designers need to take further steps to refine their databases to prevent data anomalies. One of these steps is to analyze and modify the data model at the design level.

Field-level Disentanglement:

The first option, which we call “field-level disentangle-ment,” seeks to untangle data interrelationships at the field level. We can demonstrate this approach by further analyzing the relation *Degrees*(*EmployeeID, Total_Degree, IT_Degree*) shown in Table 7, in which both *Total_Degree* and *IT_Degree* have restricted domains because they do not represent two disjoint subsets in terms of categorical classification of degrees.

Horizontal Decomposition:

As the name suggests, this method decomposes a relation horizontally by splitting a relation into multiple relations with the same table structure. It targets mainly relations with restricted domains that are caused by a limited number of domain subsets. Practitioners use decomposition along the line of these domain subsets to remove the restrictions on domains**.**

Conclusion:

In creating and maintaining relational databases, merely meeting the traditional normalization requirements is not enough to eliminate some basic data anomalies. Common data anomalies can exist in high-level normal forms because of the existence of functional entanglements. By identifying functional entanglements in a database and restricting their effects, practitioners can greatly improve data quality. We introduced two different methods to identify functional entanglements by detecting sub domain dependencies and restricted domains. We also examined two methods of eliminating functional entanglements at the design level in a normalized database: field-level disentanglement and horizontal.