Functional Dependency and Other Related Dependencies

Robert A. Schultz
Woodbury University, USA

INTRODUCTION

The invention and rise of the relational database starting in the 1960s was accompanied by the remarkable development of canonical design techniques, making it possible to avoid many database designs with unintended bad consequences (Codd, 1970, 1971). These techniques, called "normal forms," prevent the occurrence of what are called "anomalies." If a database design contains an anomaly, the implemented database will behave in unintended ways. If there is a deletion anomaly, data will unexpectedly disappear; an insertion anomaly will produce difficulties in adding data; and a modification anomaly will result in extra and unexpected operations in the course of changing data in the database (Kroenke, 2002).

In a relational database, data is organized into tables*;* tables are composed of records; and records are made up of fields (or data items or attributes). One or more fields or data items, called keys, are used to locate records and the remaining data items in them. A primary key uniquely identifies each record. If more than one field is needed to guarantee unique identification, the primary key is called a concatenated key or composite key. When more than one combination of data items or fields could serve as a primary key, the keys not actively used are called candidate keys.

The following (partial) table contains potential examples of all three anomalies:

- *Deletion anomaly*: If all the instructors who teach a given course are deleted, we will lose information about the course name.
- *Insertion anomaly*: We cannot input information about a course or register a student for the course unless we have an instructor for the course.
- *Modification anomaly*: In order to change the name of a course, we have to go through the records of all the instructors in the table and change the name there.

Intuitively, it is somewhat obvious that Instructor is the wrong field to choose as a key for finding information about classes. The advantage of the normal forms is that they provide a standard procedure for finding a database design that avoids the anomalies.

BACKGROUND

The normal forms are almost all defined in terms of the concept of dependency, and, in particular, the concept of functional dependency. Dependency in almost all cases is between data items (equivalently, fields or attributes); intuitively, one attribute A is functionally dependent on another attribute B if one needs to know the value of B in order to determine the value of A.

These are the most commonly found normal forms:

- **First Normal Form (1NF):** The table designs must have no repeating groups. That is, there are no fields with more than one distinct value within the same record. This normal form is automatically satisfied if a relational database is used, because data must be entered in a table with single values for each field. Primary Key 2 is an example if we were allowed to enter multiple values in a field.

To put this table into First Normal Form, we have to create a second table containing the repeating group and the original key. The two tables are as follows (with primary key underlined):

Course(<u>CourseID</u>,CourseName,Instructor,Rank) and
Course Student(<u>CourseID,StudentID,</u>StudentName,Grade)

(Primary Key 1)

Instructor	InstructorRank	CourseNumber	CourseName	StudentName	Grade
Dr. Wilson	Assistant Prof.	CI 110	Computer Apps	Rod Hudson	C
Mrs. Day	Associate Prof	EN 111	Freshman Comp	Alice Adams	B

(Primary Key 2)

CourseID	CourseName	Instructor	Rank	Student ID	StudentName	Grade
CI110.F03.sec1	Computer Apps	Dr. Wilson	Asst Prof	555-44-3323 430-22-1123 554-55-6689 5454-92-5587	Rod Hudson, Joan Crawford, Andy Bierce, Luigi Nono	A B A C
EN111.S04.sec3	Freshman Comp	Mrs. Day	Full Prof	490-40-2221 554-55-6689	Alice Adams, Andy Bierce	D B

- **Second Normal Form (2NF):** The table designs are in 1NF, and there are no partial functional dependencies, that is, functional dependencies on part of a primary key.

In the example above, **StudentName** in the second table Course-Student is functionally dependent only on **StudentID**. So, it is a violation of Second Normal Form. We convert this table to 2NF by creating a new table with the partially dependent fields and the key they were partially dependent on. Now we have the following:

Course(CourseID,CourseName,Instructor,Rank)
Course-Student(CourseID,StudentID,Grade)
Student(StudentID,StudentName)

- **Third Normal Form (3NF):** The table designs are in 2NF, and there are no nonkey functional dependencies, that is, no functional dependencies on data items that are not keys.

Continuing the example, in the Course table, **Rank** is functionally dependent on **Instructor**, which is not a key. To produce 3NF, we remove the functionally dependent field(s) to another table and include the original nonkey field as a key. So, we get the following:

Instructor(Instructor,Rank)
Course(CourseID,CourseName,Instructor)
Course-Student(CourseID,StudentID,Grade)
Student(StudentID,StudentName)

- **Boyce–Codd Normal Form (BCNF):** The definition of BCNF is that the table designs are in 3NF and continue to remain so for all candidate keys.

Examples grow more complex for the remaining normal forms. It is not too hard to construct an example of BCNF. If we add fields to the Student table, say, which could also serve as primary keys such as **DriversLicense** or **CreditCardInfo,** then if one of these fields were made key,

the Student table would no longer be in 3NF, because **StudentName** would now be dependent on a nonkey field. Basically, the result would be to once again remove **StudentName** to a separate table dependent only on **StudentID**, and to leave all the candidate keys by themselves in a separate table.

- **Fourth Normal Form (4NF):** The table designs are in BCNF, and there are no multivalued dependencies.

- **Fifth Normal Form (5NF):** The table designs are in 4NF, and there are no join dependencies.

- **Domain Key/Normal Form (DK/NF):** All constraints on values can be derived from domain and key dependencies.

As you will note, Second and Third Normal Forms explicitly mention functional dependencies, and the Boyce–Codd Normal Form extends the application of the Third Normal Form to other choices of primary key. Fourth Normal Form uses a more complex kind of dependency defined in terms of functional dependency. The remaining two, Fifth Normal Form and Domain Key Normal Form, introduce somewhat different kinds of dependencies. The last two Normal Forms are not often met with in practice. Near the end, I will comment on the different forms of dependency these Normal Forms employ. For the commonly employed Normal Forms, functional dependency is the key concept.

UNDERSTANDING FUNCTIONAL DEPENDENCY

There are two different ways of defining functional dependency: an intuitive intensional way and a precise extensional way. Intensional definitions use psychological or meaning elements involving dependencies in knowledge—for example, we need to know customer name in

3 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/functional-dependency-other-related-dependencies/14416

Related Content

Technology of Formal Education
Donald A. Hantulaand Dareleen M. DeRosa (2005). *Encyclopedia of Information Science and Technology, First Edition (pp. 2761-2764).*
www.irma-international.org/chapter/technology-formal-education/14689

Interactive Multimedia: A Tool for Government Transformation
David E. Barbeeand Ronnie B. Lowenstein (1993). *Information Resources Management Journal (pp. 4-14).*
www.irma-international.org/article/interactive-multimedia-tool-government-transformation/50983

Building Wireless Grids
Marlyn Kemper Littman (2009). *Encyclopedia of Information Science and Technology, Second Edition (pp. 433-437).*
www.irma-international.org/chapter/building-wireless-grids/13610

An Empirical Assessment of Technology Adoption as a Choice between Alternatives
Ernst Bekkering, Allen C. Johnston, Merrill Warkentinand Mark B. Schmidt (2009). *Information Resources Management Journal (pp. 23-44).*
www.irma-international.org/article/empirical-assessment-technology-adoption-choice/37207

User Satisfaction with EDI: An Empirical Investigation
Mary C. Jonesand Robert C. Beatty (2001). *Information Resources Management Journal (pp. 17-26).*
www.irma-international.org/article/user-satisfaction-edi/1197