MIXED TRANSITIVITY FOR FUNCTIONAL AND MULTIVALUED DEPENDENCIES IN DATABASE RELATIONS

Carlo ZANIOLO
Sperry Research Center, Sudbury, MA 01776, U.S.A.

Received 1 June 1979

Relational data bases, functional dependencies, multivalued dependencies, interference rules

1. Introduction

In the framework of the relational approach to database systems [5,7] functional and multivalued dependencies have supplied the basic tools for a formal analysis of database semantics and the definition and design of database schemas [3].

Functional dependies have been part of the relational model since its inception [6]. Multivalued dependencies, more recently introduced by Zaniolo [10] and, independently, by Fagin [8], can be regarded as a generalization of functional dependencies with which they share many formal properties. Much of the work in relational database schema analysis, definition and design is based on these dependencies and their inference rules (also called axioms) [3]. Of particular importance is the availability of a complete system of inference rules (complete axiomatization). A complete axiomatization for the combined functional and multivalued dependencies was given in [2].

In this paper we present a rule, called *mixed transitivity,* which supplies an alternative and simpler complete axiomatization for the combined functional and multivalued dependencies of database relations.

2. Dependencies in relations

A relational database consists of a set of relations each defined on a set of attributes. A relation R with attribute set $U = \{A_1, ..., A_n\}$ is denoted R(U). Underlying each attribute A there is a domain, denoted DOM(A), which is the set of possible values for this attribute. Formally R(U) is defined as a subset of the cross product $DOM(A_1) \times \cdots \times DOM(A_n)$. The elements of a relation are called *tuples.* If t is a tuple of R(U) and A is an attribute in U, then t[A] denotes the A-component of t. Similarly if X is a subset of U, then t[X] is the sub-tuple of size |X|, containing the components of t corresponding to the elements of X. t[X] is called the X-value of t. If, for some tuple t in R, x = t[X], then x is called an X-value of R. In this paper we use the letters X, Y, W, Z and Z′ to denote subsets of U. Also we write XY to denote the union of sets X and Y. $\overline{Z}$ denotes the complement of Z with respect to U (i.e. $\overline{Z} = U - Z$). Thus $\overline{XY}$ denotes the complement of the union of X and Y.

Functional dependencies. Y is functionally dependent on X in R(U), written $X \to Y$, iff every two tuples of R which have the same X-value also have the same Y-value.

For each X and each X-value, x, of R(U) we can construct the function $Y_R(x)$ which supplies the set of Y-values appearing with this X-value in tuples of R(U):

$$Y_R(x) = \{y \mid \exists t \text{ in } R(U) \text{ such that } t[X] = x \text{ and } t[Y] = y\}.$$

Multivalued dependencies. Let $Z = \overline{XY}$. Y is multidependent on X in R(U), written $X \twoheadrightarrow Y$, if for every

XZ-value, xz, of R:

$Y_R(xz) = Y_R(x)$,

i.e., the set of Y-values associated with a given x appears with every combination of x and z.

The existence of certain dependencies in a relation R(U) implies the existence of other dependencies in this R(U) (i.e., no counterexample relation on U can exist where the former dependencies hold while the latter do not). *Inference rules* are means to construct implied dependencies. A system of inference rules is said to be *complete* when all the implied dependencies can always be constructed by (repeated) applications of these rules.

The following is a complete system of inference rules for functional dependencies [1,9]:

FD rules:

F1 (Reflexivity): if $Y \subseteq X$, then $X \to Y$,

F2 (Augmentation): if $Z \subseteq W$ and $X \to Y$, then $XW \to YZ$,

F3 (Transitivity): if $X \to Y$ and $Y \to Z$, then $X \to Z$.

The following system of inference rules is complete for multivalued dependencies [2]:

MD rules:

M0 (Complementation): let $XYZ = U$ and $Y \cap Z \subseteq X$, then $X \twoheadrightarrow Y$ iff $X \twoheadrightarrow Z$,

M1 (Reflexivity): if $Z \subseteq X$, then $X \twoheadrightarrow Y$,

M2 (Augmentation): if $Z \subseteq W$ and $X \twoheadrightarrow Y$, then $XW \twoheadrightarrow YZ$,

M3 (Transitivity): if $X \twoheadrightarrow Y$ and $Y \twoheadrightarrow Z$, then $X \twoheadrightarrow Z - Y$.

If F and G respectively denote a set of functional and multivalued dependencies of R(U), then F and G combined may imply additional dependencies which are neither inferrable from F using the FD rules nor from G using the MD rules. The mixed inference rules provide the means to construct these additional dependencies. We will consider two rules:

Mixed rules:

MX1: if $X \to Y$, then $X \twoheadrightarrow Y$,

MX2: if $X \twoheadrightarrow Z$ and $Y \to Z'$, where Y and Z are disjoint and $Z' \subseteq Z$, then $X \to Z'$.

The system of rules,

{F1, F2, F3, M0, M1, M2, M3, MX1, MX2}

is complete for the combined functional and multivalue dependencies [2].

Other useful properties can be derived from the ones listed above. For instance,

$X \to Y$ iff $X \to Y - X$.

For a list of other useful properties and their applications, see e.g. [2,11].

3. Mixed transitivity

Among the rules previously considered MX2 is probably the most complex and the last intuitive. We will show that MX2 can be replaced by a simpler rule called mixed transitivity. This can be stated as follows:

MX0 (Mixed Transitivity): if $X \twoheadrightarrow Y$ and $Y \to Z$, then $X \to Z - Y$.

MX0 is a simple rule, very easy to remember because of its obvious similarity to the transitivity rules of functional and multivalued dependencies.

Let us prove that MX0 is correct. Assume that $X \twoheadrightarrow Y$ and $Y \to Z$, in R(U). By complementation we obtain $X \twoheadrightarrow \overline{Y}$. Moreover $Y \to Z - Y$, where $Z - Y \subseteq U - Y = \overline{Y}$. But since Y and $\overline{Y}$ are disjoint, we can apply MX2 to obtain $X \to Z - Y$.

Let us now prove that

F1, F2, F3, M0, M1, M2, M3, MX0, MX1

is a complete system of inference rules for the combined functional and multivalued dependencies of a relation. We need to show that MX2 is derivable from this last set of rules. So, let us assume that, in R(U),

(1) $X \twoheadrightarrow Z$ and
(2) $Y \to Z'$, where
(3) Y and Z are disjoint and
(4) $Z' \subseteq Z$.

Now $X \rightarrow\rightarrow \overline{Z}$ by complementation on (1). According to (3), $Y \subseteq \overline{Z}$. Thus $\overline{Z} \rightarrow Z'$, by augmentation on (2). Thus applying the mixed transitivity rule we obtain $X \rightarrow Z' - \overline{Z}$, where $Z' - \overline{Z} = Z'$ according to (4).

Thus MX0 can be used *in lieu of MX2* to provide a complete axiomatization for the combined functional and multivalued dependencies of a relation.

A recent report by Beeri also discusses mixed transitivity and illustrates its role in constructing relational database schemas [4].

Acknowledgment

I would like to thank Richard Flower for his comments and suggestions.

References

[1] W.W. Armstrong, Dependency structures of database relationships, in: J.L. Rosenfeld, Ed., Information Processing 74 (North-Holland, Amsterdam 1974) 580–583.

[2] C. Beeri, R. Fagin and J.H. Howard, A complete axiomatization for functional and multivalued dependencies in database relations, in: Proc. ACM SIGMOD International Conference on Management of Data, Toronto, Canada, August 3–5 (1977).

[3] C. Beeri, P.A. Bernstein and N. Goodman, A sophisticate's introduction on database normalization theory, in: Proc. 4th International Conference on Very Large Databases, West Berlin, Germany (1978).

[4] C. Beeri, On the role of data dependencies in the construction of relational data base schemas, Dept. of Computer Science, the Hebrew University of Jerusalem, Report No. 43 (1979).

[5] E.F. Codd, A relational model for large shared data bases, Comm. ACM 13 (6) (1970).

[6] E.F. Codd, Further normalization of the database relational model, Database Systems, Courant Computer Science Series 6 (Prentice-Hall, New York, 1972).

[7] C.J. Date, An Introduction to Database Systems (Addison-Wesley, New York, 2nd ed. 1977).

[8] R. Fagin, Multivalued dependencies and a new normal form for relational databases, ACM Trans. Database Systems (September 1977).

[9] R. Fagin, Functional dependencies in a relational database and propositional logic, IBM J. Res. Develop. 21 (6) (1977) 534–544.

[10] C. Zaniolo, Analysis and design of relational schemata for database systems, Ph. D. Thesis, Computer Science Dept. Report, University of California at Los Angeles, UCLA-ENG-7669 (1976).

[11] C. Zaniolo and M.A. Melkanoff, On the design of relational database schemas, ACM Trans. Database Systems, to appear.