The Problem: Determination of Functional Dependencies between Attributes Relation Scheme in the Relational Data Model

Ignacio Vega-Páez[1], José Angel Ortega[2] and Georgina G. Pulido[3]
Mexico, D.F

***Abstract*— An alternative definition of the concept is given for "functional dependency" among the attributes of the relational scheme in the Relational Model. This definition is obtained in terms of the set theory. For that which a theorem is demonstrated that establishes equivalence and on the basis theorem, an algorithm is built for the search of the functional dependencies among the attributes. The algorithm is illustrated by a concrete example.**

***Index Terms*—** Data Scheme, Database, DB, Data Scheme, Functional Dependency and Relational Model

I. INTRODUCTION

All of the projects for the development and construction of Entrepreneurial Information Systems are based on the concept of Relational Databases. In general, a Database of an Entrepreneurial Information System will be understood as a collection of basic data that is free of redundancies; while the content and the organization of this collection assures an effective solution to the Entrepreneurial Information System. In other words, the Data Base is a model of the knowledge of the Company, which abstract the properties of the "real company" that are relevant concerning the operational, administrative, business and other points of view.

At present, as a formal tool for the description at a logical level of a Data Base, the Codd Relational Model [1] is used; this model is based on the concept of *relation scheme* or *relational object*. This concept is used for the description of objects of the same type, and it reflects the connections between them. Each one of the properties of the objects is expressed through the concept of attribute within at a relation scheme (or commonly called "*relation*") considered as sets of values of such property. As a rule, a single name is assigned to each attribute; it is made to coincide with the name of such property to maintain the semantic meaning of the attribute. The elements of the relation scheme are the ordered *n-tuplets* of the values of the attributes, called, more formally "*sequences*", to the set of elements of a relation scheme it is a subset of the Cartesian product $A_1 \times A_2 \times \ldots \times A_n$, where each A_i, is *the i-th* attribute, which we will denote as $R(A_1, A_2, \ldots, A_n)$. The number of attributes of the relation is called the degree of the relation.

When a Relational Database is constructed, the problems to identify the elements in the relations and the selection of the representation method for such relations must be the most optimal from the points of view of manipulation as well as the non-redundancy in the storage of the data and both have considerable relevance in the design phase. The basic solution to both problems is the concept of functional dependency among attributes of the relation schemes.

A function dependency of attribute A of a relation R over the set of attributes B of the same relation is defined as the dependency for which each "sequence" of values of the set of attributes B is assigned as not higher than the value of A, which is induced with some sequence of R. We will use de traditional notation $A \to B$ to denote the functional dependency and we will denote $A \underset{R}{\to} B$ for the functional dependency of the attributes A and B of relation R if there are several relations involved.

The presence of functional dependencies among attributes may be explicitly postulated when a database is described in the design phase, following the meaning of the data to be modeled (semantic of the objects to be described). Armstrong [2] researches possible families of such functional dependencies. The incomplete knowledge of the semantic of the objects to be described may, nevertheless, leave the problem of functional dependencies undiscovered/hidden. The problem of discovering such functional dependencies in a real database is the problem outlined in this work; furthermore it is a latent problem that "normally" does not appear until the start up of production.

The algorithm that we propose at the end is the solution to this problem.

[1] I. Vega-Páez. is with Section of Graduate Studies and Research of ESIME IPN (SEPI ESIME IPN), Zacatenco México, D.F., Department of CS, CINESTAV-IPN, México D.F. & Direction of Technology and Systems of Acciones y Valores Banamex Casa de Bolsa, S.A. de C.V. (e-mail: ivegap0800@ipn.mx).

[2] J. Angel Ortega is now with the Section of Graduate Studies and Research of ESIME IPN (SEPI ESIME IPN), Zacatenco México, D.F (e-mail: oeha430210@hotmail.com).

[3] G. G. Pulido is with the Department of Basic Sciences and Engineering UAM Azcapotzalco, México, D.F. (e-mail: gpulido@att.net.mx).

II. DEFINITION AND OUTLINE OF THE PROBLEM

Let us consider a relation $R(A_1, A_2, ..., A_n)$, where A_i, are the attributes of the relation. Let us introduce attribute K that will be made up by an ordinal number with the role of the sequence that enumerates all of the elements that make up relation R, which we will call "enumeration" of R, this list, obviously, defines uniquely each and every one of the sequences of this relation,

$$R(K, A_1, A_2, ..., A_n) \tag{1}$$

The projection of this set of sequences of the relation R over the K and A_i, formally $\pi_{\{K,A_i\}}(R)$ (Ulman[3]), may, be written as,

$$F_i = \{\langle k, a_i \rangle | k \in K, a_i \in A_i\} \tag{2}$$

Then the triplet $\langle F_i, K, A_i \rangle$ defines the function f_i, for each $i \in [\![1, n]\!]$, that meets the following properties:

a) $F_i \subseteq K \times A_i$

b) In consideration of the sole assessment (single-valued) of the enumeration, and in (2) there is no par with the first repeated element.

$$f_i \underset{Df}{=} \langle F_i, K, A_i \rangle \quad \text{or} \quad A_i = f_i(K) \tag{3}$$

The function is suprajective or above, because the following is met:

$$(\forall a_i^k \in A_i)(\exists k \in K)[f_i(k) = a_i^k] \tag{4}$$

and, consequently, the system of $\overline{K}_i$ class, generated by $f_i^{-1}(A_i) = \overline{K}_i$, is a partition of the K set.

If we denote f_i^{-1}, by ϕ_i, we can write:

$$\overline{K}_i = \Phi_i(A_i) \tag{5}$$

III. FOUNDATIONS

PROPOSITION: For each i, the function ϕ_i, is bijective.

In effect: for each $a_i^m \in A_i$ is identified as only one class $\overline{K}_i^m \in \overline{K}_i$, and different classes $\overline{K}_i^{m_1}$ and $\overline{K}_i^{m_2}$ are identified by different $a_i^{m_1}$ and $a_i^{m_2}$; each class $\overline{K}_i^m$ is identified with only one a_i^m.

The following theorem uses the above constructions (1)-(5).

THEOREM: (Set Functional Dependency). Attribute A_{i_2} functionally depends on attribute A_{i_1} if, and only if each class of the partition $\overline{K}_{i_1}$ is a subset of at least one class of the partition $\overline{K}_{i_2}$, or formally

$$\left(\forall m, \exists n, \overline{K}_{i_1}^m \subseteq \overline{K}_{i_2}^n\right) \Leftrightarrow A_{i_1} \underset{R}{\rightarrow} A_{i_2} \tag{6}$$

Proof:

Necessity: Based on (5) for i_1 and i_2 we have

$$\overline{K}_{i_1} = \Phi_{i_1}(A_{i_1}),\ \overline{K}_{i_2} = \Phi_{i_2}(A_{i_2}) \tag{7}$$

By the bijectivity of the functions ϕ_i

$$A_{i_2} = \Phi_{i_2}^{-1}(\overline{K}_{i_2}) \tag{8}$$

We have $\forall m, \exists n, \overline{K}_{i_1}^m \subseteq \overline{K}_{i_2}^n$, thus a function φ on exists, such that

$$\overline{K}_{i_2} = \varphi(\overline{K}_{i_1}) \tag{9}$$

then based on the expressions of 7-9, we can write

$$A_{i_2} = \Phi_{i_2}^{-1}(\overline{K}_{i_2}) = \Phi_{i_2}^{-1}(\varphi(\overline{K}_{i_1})) = \\ = \Phi_{i_2}^{-1}(\varphi(\Phi_{i_1}(A_{i_1}))) = F(A_{i_1})$$

Due to the composition of the functions it is a function, A_{i_2}, functionally depends of A_{i_1}.

Sufficiency: By reduction to the absurd, we will prove that if,

$$\exists m, \forall n, \overline{K}_{i_1}^m \not\subseteq \overline{K}_{i_2}^n \tag{10}$$

Then $A_{i_1} \underset{R}{\not\rightarrow} A_{i_2}$

from the condition (9) it follows that k' and $k'' \in \overline{K}_{i_1}^{m_1}$ exist, as long as $k' \in \overline{K}_{i_2}^{n_1}$ while $k'' \in \overline{K}_{i_2}^{n_2}$ and, this is a consequence of the fact that the partition $\overline{K}_{i_1}$, k' and k'' pertain to one sole class $\overline{K}_{i_1}^{m_1}$, and by the bijectivity of ϕ_{i_1} (see (5)) en one sole value $a_{i_1}^{m_1}$, correspond to both. In the partition $\overline{K}_{i_2}$, k' is in class $\overline{K}_{i_2}^{n_1}$, and has value $a_{i_2}^{n_1}$. While k'' is in class $\overline{K}_{i_2}^{n_2}$, and it has the value $a_{i_2}^{n_2}$. In other words, one and the same value of attribute A_{i_1}, in different enumerations of the scheme of the relation are matched for different values of attribute A_{i_2}, which contradicts the definition of functional dependency. ■

IV. COROLLARIES AND RESULTS

***REMARK**:* If the set of classes of the partition correspond to the set of values of the collection of attributes $A_{j_1}, A_{j_2}, ..., A_{j_x}$, it is defined as :

$$\overline{K}_{j_1,j_2,...,j_x} = \overline{K}_{j_1} \cap \overline{K}_{j_2} \cap ... \cap \overline{K}_{j_x}$$

then for the functional dependency of attribute A_i, over the collection of attributes $A_{j_1}, A_{j_2}, ..., A_{j_x}$, the theorem is taken as follows:

$$\left(\forall m, \exists n, \overline{K}^{m}_{j_1,j_2,...,j_x} \subseteq \overline{K}^{n}_{i}\right) \Leftrightarrow \left(A_{j_1}, A_{j_2}, ..., A_{j_x}\right) \underset{R}{\rightarrow} A_i \quad (11)$$

COROLLARY 1.

$$\overline{K}_{i_1} = \overline{K}_{i_2} \Leftrightarrow A_{i_1} \underset{R}{\leftrightarrow} A_{i_2}$$

The corollary is proven based on the symmetric substitution of the indexes i_1 and i_2 in the theorem.

In order to indentify an element of this relation we can use any attribute or a collect of attributes over any other attributes of the relation on which they functionally depend. Such attribute or collection of attributes (from which we cannot take out any of the attributes, without perturbing this dependency) is called a candidate key. The values of the attributes of any candidate key identify uniquely the elements of the relation.

COROLLARY 2. A minimal set $A_{j_1}, A_{j_2}, ..., A_{j_x}$, to which the set of classes of the relevant partition

$$\overline{K}_{j_1} \cap \overline{K}_{j_2} \cap ... \cap \overline{K}_{j_x} = \{\{1\}, \{2\}, ..., \{p\}\}$$

It is the candidate key of the relation.

ALGORITHM

By construction, attribute K is a candidate key of the relation. The partition of set K for this attribute is the set

$$\{\{1\}, \{2\}, ..., \{p\}\}$$

Among the "candidate keys" of one relation a mutual functional dependency exists; thus, based on "Corollary 1" the partitions of set K correspond to these collections of attributes that must coincide.

Based on the proven theorem we propose the following algorithm in order to determine the functional dependency among the attributes of a relation.

1. For each attribute of the relation we construct a lexicographically ordered projection of the set of enumerations of the relation R over K and A_i $(i = 1,2,...n)$.

   As a result of the construction of the pairs, the set of the second elements where the first elements are equal for each projection make up the partition of set K for the relevant attributes.

2. In order to determine the presence of a functional dependency between the attributes of the relation the intersection of each class of the partition generated by an attribute must be verified against the classes of the partition generated by the other attribute. If at least one class of the partition of the first attribute is not a sub-set of the same class of partition of the second attribute, we can conclude the absence of a functional dependency over the first attribute (for the necessity of our theorem, see expression (6)).

   When the class of the first attribute, which consists of one sole element, is always a sub-set of any class of partition of the second attribute, it is advisable to just carry out one verification for the classes containing more than two elements.

   There is another obvious affirmation, in order to reduce the number of comparisons, which consists of the fact that we only have to compare those classes of the second attribute for which the number of elements is not less than the number of elements of the class of the first attribute to be compared.

By means of the presented algorithm we can find all of the functional dependencies of the relation. To that effect, based on (11) it follows that, we must construct all the sets of the partition classes corresponding to the sets of values of all of the collections of attributes $A_{j_1}, A_{j_2}, ..., A_{j_x}$.

V. EXAMPLE

Example: Let's take relation

PostalDelivery (Code, Color, Volume, Weight)

Let's apply enumeration K to relation *PostalDelivery*

PostalDelivery (K, Code, Color, Volume, Weight)

The Table 1 shows the aforesaid:

Table 1:

K	*Code*	*Color*	*Volume*	*Weight*
1	A1	RED	15	150
2	B2	BLUE	20	230
3	CA1	YELLOW	18	160
4	CB2	GREEN	40	420
5	C4	YELLOW	18	160
6	3	BLUE	25	210
7	4	GREEN	40	360
8	5	BLACK	60	540

The following partitions of set K correspond to the attributes of the relation *PostalDelivery* represented in the above Table 1.

$$\overline{K}_{Code} = \{\{1\},\{2\},\{3\},\{4\},\{5\},\{6\},\{7\},\{8\}\}$$
$$\overline{K}_{Color} = \{\{1\},\{2,6\},\{3,5\},\{4,7\},\{8\}\}$$
$$\overline{K}_{Volumen} = \{\{1\},\{2\},\{3,5\},\{4,7\},\{6\},\{8\}\}$$
$$\overline{K}_{Weight} = \{\{1\},\{2\},\{3,5\},\{4\},\{6\},\{7\},\{8\}\}$$

In the following Table 2:

Table 2:

ATTRIBUTES	CODE	COLOR	VOLUME	WEIGHT
CODE	1	0	0	0
COLOR	1	1	1	1
VOLUME	1	0	1	1
WEIGHT	1	0	0	1

The value 1 expresses the presence of a functional dependency of the row n of the relevant column. Thus, the key role of the relation may be verified or chosen to "KEY" attribute.

VI. CONCLUSION

The presented algorithm enables us to verify the presence or absence of a functional dependency among the attributes of a relation for an enumeration. It is evident, the larger the number of enumerations to be analyzed, the higher the certainty of the conclusion obtained/reached. The last question concerning the presence of a functional dependency between the attributes of a relation of an arbitrary set of enumerations can be resolved by means of an analysis of the semantics of the attributes or simply by verifying the semantics of the attributes against the functional dependencies inferred in the design.

REFERENCES

[1] E. F. Codd, "A relational model of data for large shared data banks", Commun. ACM, 13, No. 6, 377 (1970).

[2] W. W. Armstrong, "Dependency structures of database relationships", Proc. IFIP Congr.3, 580 (1974)

[3] Jeffrey D. Ullman, "Principles of Database Systems", Computer Science Press, Inc. (1980)

[4] Vega-Páez I. y F.D. Sagols T., "Space Program Language (SPL/SQL) for Relational Approach of the Spatial Databases", Proceedings in Information Systems Analysis and Synthesis ISAS '95, 5th International Symposium on Systems Research, Informatics and Cybernetics, pp. 89-93 August 16-20, 95, Baden-Baden, Germany (1995)