Advances in Computational Sciences and Technology
ISSN 0973-6107 Volume 10, Number 6 (2017) pp. 1893-1901

http://www.ripublication.com

Duplicate Record Detection and Replacement within a Relational Database

S. Aquter Babu

Assistant Professor,
Department of Computer Science, Dravidian University,
Kuppam, Chittoor District, Andhra Pradesh, India.

Abstract

In a relational database, records are duplicated in many real time applications. Finding and then removing duplicate records with one corresponding and correct record is must. Sometimes one incorrect record must be replaced by another correct record and some other times many incorrect records must be replaced by one or more correct record. Note that in removing duplicate records one must take care of violations of integrity constraints and in particular referential integrity constraints and these must be controlled carefully and correctly without any data inconsistencies. A new technique is proposed to find and then remove one duplicate record with one correct record.

Keywords: Database, Duplicate record detection, removal.

1. INTRODUCTION

Data duplication is a very big problem in data management and it causes many potential real problems in database operations. The problem of duplicate data (also known as record linkage or entity resolution) adheres to the fact that there can exist multiple descriptions of the same real world entity within one or more databases [1]. There does not exist one exact deterministic algorithm that finds all duplicate records and replaces with one or more accurate records. If all the steps in finding and removing duplicate records are deterministic then it is possible to write an exact

algorithm. But in many real cases it may not be possible to find all the required steps deterministically at the beginning itself.

In the past decades, the detection and removal of duplicate data has gained a lot of attention in research [2]. In the context of relational databases [4], dealing with duplicates comes down to (a) identifying which tuples are duplicate and (b) replacing those tuples by a single tuple [3].

Needless to say, in the context of a relational database, the deletion of the original-duplicate-tuples should take into account integrity constraints, in particular referential integrity, which is assumed to be satisfied in the original database [1]. Also manual operations are costly and time consuming. Definitely there is a need to automate this record duplication problem without violating any data integrity problems in the relational databases and without violating referential integrity problems.

Existing strategies such as on delete cascade, on update cascade, set null and restrict are potentially incapable of controlling and managing record duplication in relational databases. That is, presently available data control and management techniques can control only syntactically correct relations but not semantically correct relations. But what actually needed are semantically correct relational databases.

Record duplication problem may exist within a single relational database or among the many relational databases. A framework is needed to manage not only existing data controlling techniques but also to manage semantically needed controls also.

There is a possibility to find and apply mathematically related functions or formulas to overcome the data duplication problems in the relational databases. Sometimes it may be needed more than one formula or combination of many formulas also for efficient and effective management of record duplication problems. A new framework is definitely needed that can handle existing data integrity constraints as well as new data duplication finding and removing procedures. Here, data duplication is discussed particularly with respect to the single relational database only. Sometimes duplicate record replacement may be performed in many stages using incremental approach.

2. LITERATURE SURVEY

There exist many reasons for data duplication. The other names for data duplication are entity resolution and record linkage. Various reasons for data duplications are:

1. Data missing
2. Typing errors
3. Copy errors
4. Poor management of data standards

5. Lack of data quality controls
6. Lack of integrity controls
7. Lack of data controls

In the literature of record duplication many frameworks have been proposed for the removal of duplicate records within a single relation of the relational databases. Many authors have also studied duplicate data removal from many relations simultaneously. Recently, several authors have studied the removal of redundant data from multiple relations at once [1]. Some of the mathematical formulas that can be used for detecting and removing duplicate records are – set union, set intersection and symmetric difference and so on. Many of the methods in the literature have been focused only on accurate representation of the records in the relational databases but not on accurate representation relationships.

Motro [4] has approached the problem of data fusion as a multi-dimensional optimization problem. In his work, he has proposed a utility function that is a linear combination of six meta data dimensions. Of course, DBMSs offer default strategies like cascading as prescribed by the SQL standard [5], but again, there is no known research on the impact thereof on the quality in relationship tables [5]. Record duplication problem in the relations of relational databases is directly related to collective entity resolution methods. In data warehouse also record duplication problem exists and special methods and techniques are needed to overcome such problems. Some of the possible potential solutions are based on functional dependencies. Methods that are used for managing duplicate records in relations of the relational databases are:

1. Collective entity resolution
2. Markov logic networks
3. Multidimensional optimization
4. HumMer system
5. Utility function usage
6. Dependency graph approach
7. Relational clustering
8. Usage of special language called Dedupalog
9. Extended standard SQL syntax

Sometimes there is a need to consider a chain of relations and this particular chain reference is very useful whenever different parts of the database is modeled with hierarchical data structure.

Various ways that can be used for finding and then replacing duplicate values are:

1. f-optima function
2. f-optimal voting
3. Majority voting

A new framework is required for efficient and effective data management of record duplication problems in the relational databases. New framework must manage and control semantically correctness in addition to the syntactically correctness. In the literature of record duplication in the relations of the relational databases this problem is called data propagation of fusion operation. Propagation algorithms are based on fusion functions which performs recursively fusion operations.

3. PROBLEM DEFINITION

Record duplication is common in many relations of the relational databases. Higher level techniques are needed to find and then replace duplicate records with the correct records. Consider a database consisting of three relations-DEPT, EMP, and PROJECT_DETAILS. The relationships among these three relations are shown below:

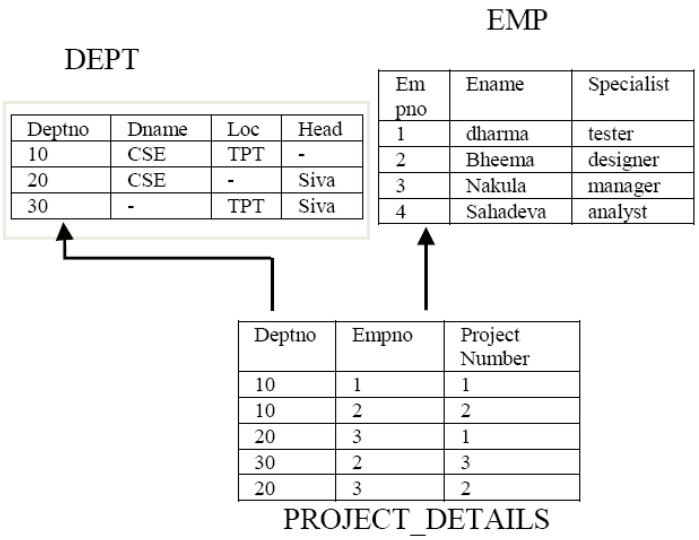

Figure 1: Project Database

DEPT relation contains three distinct tuples with respect to the primary key. But logically all these three tuples represent only one actual record. Unfortunately before identification of duplicate records in the DEPT relation its records have been referenced in the PROJECT_DETAILS relation by relating primary key and foreign key relationship. Later on after some time if somebody identifies that the relation

DEPT is actually contains duplicate records, then as a result of record duplication in DEPT all the referenced relations with respect to DEPT reference must be updated accordingly for data consistency of the referenced relations in a single database or across many databases.

In the first step all the referenced relations must be identified and then in the second step all such relations must be modified or updated accordingly by using any record duplication management techniques or by using any mathematics related techniques. Mathematics related techniques are theoretically feasible bust sometimes practically may not be feasible or possible. In such cases special methods are needed. Mathematical operation called union operation procedure is explained with the help of single example database consisting of three sample relations, DEPT, EMP, and PROJECT_DETAILS. For simplicity only single database is taken into consideration. But actually in real time applications it may not be the case.

Some of the potential mathematical operations are-union, intersection, set difference, string length difference, functional optimal methods, cosine similarity, and set symmetrical difference and so on. Here, union operation is considered for explaining running example of the database.

Union operation is one way to find and then remove duplicate records in the relational databases.

Union of DeptName = {CSE union CSE union null} = CSE

Union of Location = {TPT union null union TPT} = TPT

Union of Head = {null union Siva union Siva} = Siva

Therefore three duplicate records in DEPT relation are updated and replaced with only one correct record and now the DEPT relation contains the following data shown in DEPT2 relation in the FIGURE-2.

Out of three duplicate records in the relation DEPT one correct record is shown in DEPT2 relation. That is after removal of duplicate records DEPT2 contains only one correct record without any duplication with 10 as its primary key value. Duplicate record identification and removal consists of two steps. In the first step, duplicate records are identified and replaced with correct record. In the second step based on the modified details shown in DEPT2 all of its referenced relations are identified and then updated.

DEPT2

Deptno	Dname	Loc	Head
10	CSE	TPT	-

Figure 2

In the DEPT table initially three records are present with primary key values 10, 20, and 30 respectively. But after modification only one record is there with primary key value 10. A rule of thumb is that out of all the duplicated records select the one with the minimum primary key value. Here 10 is the minimum and hence it is selected as primary key after DEPT relation modification. The updated relation is shown in FIGURE-3. In the literature there are any methods to select one accurate record after replacement.

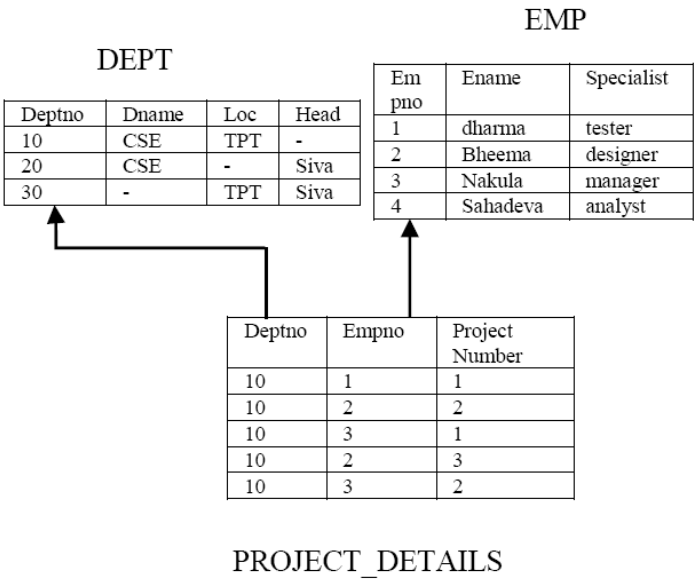

Figure 3: Modified Project Database

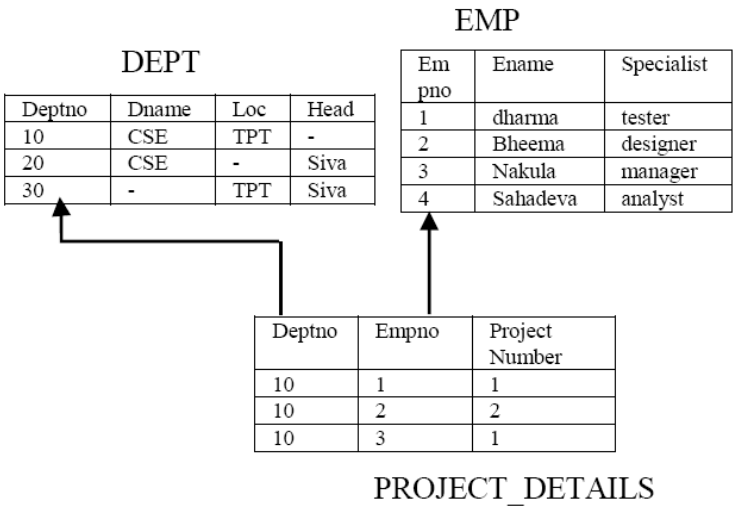

Figure 4: Updated Project Database

Now observe the project relation, PROJECT, it contains duplicate values in the primary key. But duplicate values in the primary key are not allowed. Temporarily primary key property must be disabled, and then PROJET relation must be updated so that it satisfies the entity integrity property and the latest modified database details are shown in FGURE-4.

Duplicate values in the primary key attributes of the PROJECCT_DETAILS relation are handled in the following way. There exist only one record, {10, 1, 1}, with primary key value {10, 1}. There exist two records, $\begin{Bmatrix}10, 2, 2\\10, 2, 3\end{Bmatrix}$, with the primary key value {10, 2}. A rule is proposed for selecting only one record so that entity integrity rule is satisfied. The rule is that in both the records select the one with lowest project number. Lowest project number implies highest priority number for the project. Hence, based on priority number of the project number the record {10, 2, 2} is selected instead of the record {10, 2, 3}. Similarly there are two records $\begin{Bmatrix}10, 3, 1\\10, 3, 2\end{Bmatrix}$ with the same primary key values {10, 3}. Again based on the highest priority of the project number record {10, 3, 1} is selected instead of the record {10, 3, 2}. Final and updated complete database of the PROJECT are shown in FIGURE-4.

4 ALGORITHM

Algorithm Description

Assume that R is the original database relation that contains duplicate tuples. K is the primary key of the original relation R. Let delta be the set of duplicate records. Let R^{star} be the referenced relation with respect to the original relation R. Let FK be the foreign key of the referenced relation R^{star}.

ALGORITHM-for removing duplicated records in the relations of the relational databases.

INPUT:

Database of relations along with a relation containing duplicate records are input to the algorithm.

OUTPUT

Duplicated records are removed and modifications are propagated to the all referenced relations either within a single database or multi-databases.

1.for each record t belongs to the delta do

2.find set of records, S, from the referenced relation

such that S = {t^{star} / t^{star} ϵ to R^{star} and t^{star} [FK] = t[K]

3. for each record $t^{star} \epsilon S^{star}$ do

4. t^{star}[FK] = original relation replaced record[K]

5. end-of-for

6.end-of-for

7. T^{delta} = records remaining after applying priority principle

8.update all records in the T^{delta} relation

9.remove duplicate values in the primary key of the final relation by removing duplicates

S is the relation consisting of set of tuples of the referenced relation such that each tuple satisfies primary key and foreign key relationship. The algorithm executes for each duplicate tuple of the original relation.

Input to the algorithm is three relations. First relation is called original relation that contains one or more duplicate records. First duplicate records are identified. These duplicate tuples are represented by a delta set symbol. Duplicate tuples are replaced by one or more correct records. Once the replacement is completed then for each duplicate record corresponding referenced relations must be identified first and then modifications to the original base relation must propagated to all the referenced relations in same database or among the many databases. For simplicity, here only single database with three relations are considered. But in real life there may exist many more relations either in the single database or in the group of databases.

5. CONCLUSIONS

Record duplication is common and frequent problem in many relational databases. One solution to this problem is find duplicate tuples and then replace with correct tuples using best optimal techniques. Union operation is one such technique for handling duplicate records problem in the relational databases. In the future there is a possibility to invent other methods for finding and then replacing duplicate records. The other methods could be difference operator, join operation, symmetrical operation, and string matching operation, string length difference operation and so on.

REFERENCES

[1] Antoon Bronselaer, Daan Van Britson, and GUY De Tre, “Propagation of Data Fusion”, IEEE Transactions on Knowledge and Data Engineering Vol., 27, No. 5, MAY 2015

[2] A. Elmagarmid, P. Ipeirotis, and V. Verykios, “Duplicate record detection: A

survey," IEEE Trans. Knowl. Data Eng., vol. 19, no. 1, pp. 1–16, Jan. 2007.

[3] E. F. Codd, "A relational model of data for large shared data banks," Commun. ACM, vol. 13, no. 6, pp. 377–387, 1970.

[4] A. Motro and P. Anokhin, "Fusionplex: Resolution of data inconsistencies in the integration of heterogeneous information sources," Inf. Fusion, vol. 7, no. 2, pp. 176–196, 2006.

[5] (2011) [Online]. Available: ISO/IEC 9075-1:2011: Information technology database languages SQL part1: Framework (SQL/ framework)