Name:Nai-Fan Chen
NetId:nc41

1. What is the key need for this algorithm/approach/system?

Data independencies can help build a reliable database system which implements relational model. That is to say, it deals with the problems such as redundancy and inconsistency by applying Normal Form which the author detailed. Primary key, foreign key, and operations on Relations (e.g. Join, projection, permutation) enable database designers not only to build the relational model and queries the data they'd like to retrieve, but to avoid the redundancy and inconsistency as well.

2. What benefits does this algorithm/approach/system provide over existing algorithms/approaches/systems?

Database normalization ensures that a relational database model is free of data anomalies. Moreover, data retrieval becomes easy such that users could query the data from each table directly, or get the meaningful data by joining multiple tables(relations). On the system of the time, however, the method to retrieve the data needed to traverse a tree or hierarchical data.

3. What is the key object? (e.g. tweet, RDD, Tensor)

First Normal Form that maintains atomicity of data is the significant object.
One of the purposes of this paper is to introduce a new method that can reduce data redundancy and improve data integrity. Normal Form could prevent the system from undesirable side-effects such as update anomaly, insertion anomaly and deletion anomaly, and it will help system maintain data integrity.

4. Describe a technical challenge in implementing the author's solution

The model may be very often for the purpose of the business requirement. However, designing relational schemas is very time-consuming depending on complexity. Therefore, if we would like to build on the top of the database system, it will be a very complicated process. For example, because of well designed relationship or the elimination of redundancies in order to make the system faster, when designers would like to add only a simple column into a certain relation, the whole system might need to be carefully checked and the schemas sometime should be redesigned. Hence, nowadays, since the requirement of database system changes occurs frequently, the cost would be very high to modify a relational model.

5. What is a key insight or lesson learned from the paper?

The author presents the advantages of designing database system into the relational model. For example, the database should only consist of a collection of relations, so people just need to know the name of relations and their domains. Furthermore, the fundamental concepts, such as primary key, foreign key, etc., have great impact on the modern database system.

In addition, how to solve the data redundancies and the consistent issue, which are the big challenges at the time, can be answered because the author details the meaning of Normal Form such that it can simplify the database tables and address the problem.

6. What is something interesting your learned from this paper, or your thoughts about its strengths and/or weaknesses. Is there anything else interesting about this paper, or your interpretation of it that you want to share?

When I would like to build a whole new database system, normalization is ignored very often. After reading the paper, I understand the importance of the normalization since I could avoid creating redundant relations or have a clear picture of what the ER model should look like. In addition, users can update data very simple in relational database system because users just need to update the information on the certain table. Through this way, other relations could use foreign key to connect to this relation. Thus, users do not have to reenter all the records which we want to update. However, maintaining the structure of database should take time. It is a difficult task when the database is very complex.

7. Provide a citation for the additional paper you read. How does this paper relate to the assigned one? What new information did you learn by reading it?

Parallel Database Systems: The Future of High Performance Database Systems
David J. DeWitt and Jim Gray
University of Wisconsin and DEC

In this paper, the author leveraged the parallel database systems to optimize the efficiency of database system. By partitioning the input data among multiple processors and memories, an operator can often be split into many independent operators each working on a part of the data. For example, the author demonstrated how to improve the performance of relational operators. the hash join algorithm can break a large join into several little joins and this parallel algorithm could be a linear-time algorithm. However, the sort-merge join has nlog(n) execution cost which is higher than hash join. Hence, through this paper, I understand how to deal with the join operation by an efficient method. In the relational model, if we want to retrieve meaningful data from multiple relations, the system has to put lots of efforts into dealing with relational operations and large data flow. The parallel database can deal with this problem and minimize data flow to improve the performance. Thus, this paper helps create a more efficient way to deal with relational database problems.