Codd's 12 Rules in DBMS

Dr. Munesh Singh

Indian Institute of Information Technology
Design and Manufacturing,
Kancheepuram
Chennai-600127

January 8, 2019

Who is Codd?

- Edgar Frank "Ted" Codd (19 August 1923- 18 April 2003) was an English computer scientists who, while working for IBM, invented the relational model for database management, the theoretical basis for relational databases.

13 not 12

- Codd proposed thirteen rules, numbered 0 to 12
- According to him if a database meets these rules, it can be called relational database management system.

Rule 0

- The system must qualify as relational as a database and as a management system.
- For a system to qualify as a relational database management system, that system must use its relational facilities to manage database
- The other 12 rules derive from this rule.

Rule 1: Information Rule

- All information (including metadata) is to be represented as stored data in cell of tables.
- The rows and columns have to be strictly unordered.
- Meta data is name of table and attribute information (it is a information about structure)

Rule 2: Guarantied Access

- Each unique piece of data (atomic value) should be accessible by: Table Name + primary key (Row)+ Attribute(column).

Rule 3: Systematic Treatment of NULL

- Null has several meanings, it can mean missing data, not applicable or no value. It should be handled consistently.
- Primary key must not be null.
- Expression on NULL must give null.

Rule 4: Active online catalogue

- Database dictionary must have description of Database.
- Catalogue to be governed by same rule as rest of the database. The same query language to be used on catalog as on application database.

Rule 5: Powerful Language

- One well defined language must be there to provide all manners of access to data
- SQL (Structure Query Language)

Rule 6: View updation

- All view that are theoretically updatable should be updatable by the system
- View is a virtual table

Rule 7: Relational Level Operation

- There must be Insert, Delete, Update operations at each level of relations
- Set operation like Union, Intersection and Minus should also be supported.

Rule 8: Physical Data Independence

- The physical storage of data should not matter to the system
- If say, some file supporting table were renamed or moved from one disk to another it should not effect the application.

Rule 9: Logical Data Independence

- If there is change in the logical structure (table structures) of the database the user view of data should not change.
- Say, if a table is split into two tables, a new view should give result as the join of the two tables. This rule is most difficult to satisfy.

Rule 10: Integrity Independence

- The database should be able to conforce its own integrity rather than using other programs. Key and Check constraints, trigger etc should be stored in Data Dictionary.
- This also make RDMS independent of front-end.

Rule 11: Distribution Independence

- A database should work properly regardless of its distribution across a network. This lays foundation of distributed database.

Rule 12: Non-subversion rule

- If low level access is allowed to a system it should not be able to subvert or bypass integrity rule to change data.
- This can be achieved by some sort of looking or encryption