ISSN (E): 2181-4570 ResearchBib Impact Factor: 6,4 / 2023 SJIF 2024 = 5.073/Volume-2, Issue-5

DATABASE NORMALIZATION: OPTIMIZING DATA MANAGEMENT

[1]Shamuratov Ulug‘bek Alisher o‘g‘li, [2]Raxmatullayeva Aziza Abdunabi qizi, [3]Ochilboyev Umidjon Ilxom o‘g‘li

[1,2,3]In the name of Muhammad al-Khorazmi,
Students of Tashkent University of Information Technologies.

Annotation. This article introduces the process of structuring a database to reduce redundancy and improve data integrity. The use of normal forms such as 1NF, 2NF, 3NF, 4NF, 5NF and BCNF and their differences are covered.

Keywords: Database normalization, Relational database, Data redundancy, Data integrity, Normal Forms(1NF, 2NF, 3NF, BCNF, 4NF, 5NF), Data Consistency, Database design.

Аннотация. В этой статье описывается процесс структурирования базы данных для уменьшения избыточности и улучшения целостности данных. Описано использование нормальных форм, таких как 1NF, 2NF, 3NF, 4NF, 5NF и BCNF, а также их различия.

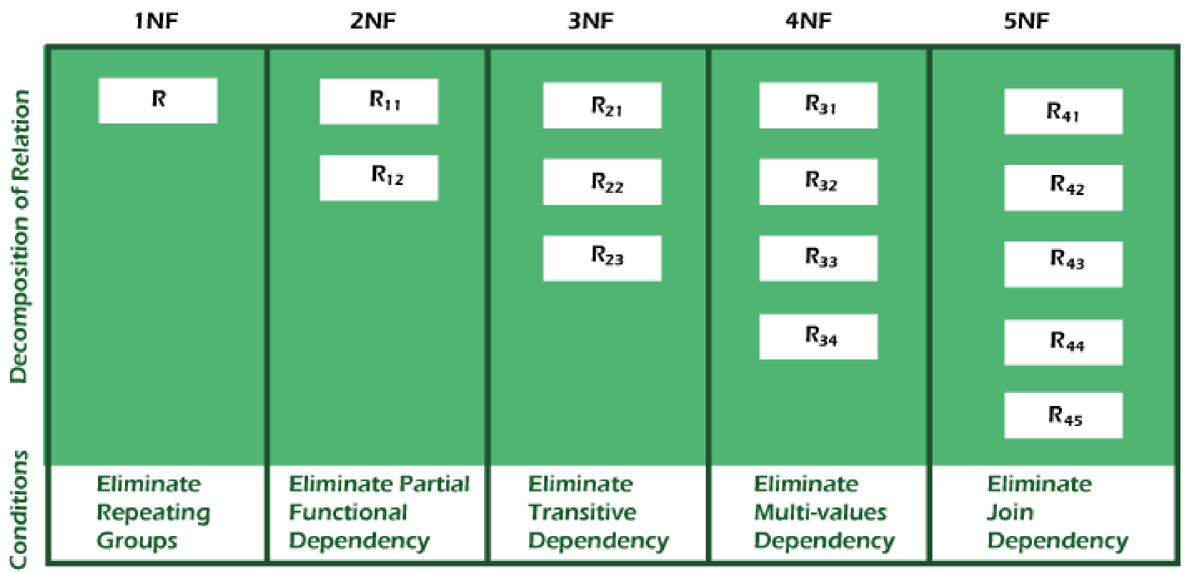
Ключевые слова: нормализация базы данных, реляционная база данных, избыточность данных, целостность данных, нормальные формы (1NF, 2NF, 3NF, BCNF, 4NF, 5NF), согласованность данных, проектирование базы данных.

In the realm of database management, ensuring the efficiency, consistency, and integrity of data is paramount. Database normalization is a critical process in this regard, aimed at organizing the data within relational databases to reduce redundancy and improve data integrity. This process involves structuring a database in a way that minimizes duplicate data and dependencies, thereby enhancing its overall performance and maintainability.

Database Normalization:

Database normalization is the process of decomposing a database into smaller, more manageable tables and defining relationships between them. The primary goals are to eliminate redundancy, ensure data dependencies make sense, and protect the data against anomalies that can occur during data operations like insertions, deletions, and updates.

The process is typically divided into several normal forms (NF), each with specific requirements:


1NF
2NF
3NF
4NF
5NF
Decomposition of Relation
Conditions
R
R_{11}
R_{12}
R_{21}
R_{22}
R_{23}
R_{31}
R_{32}
R_{33}
R_{34}
R_{41}
R_{42}
R_{43}
R_{44}
R_{45}
Eliminate Repeating Groups
Eliminate Partial Functional Dependency
Eliminate Transitive Dependency
Eliminate Multi-values Dependency
Eliminate Join Dependency

Types of Normal Form

1. First Normal Form (1NF): Ensures that the values in each column are atomic (indivisible). Each column must contain unique values, and each table must have a primary key that uniquely identifies each row.
2. Second Normal Form (2NF): Achieved when the table is in 1NF, and all non-key columns are fully dependent on the primary key. This means eliminating partial dependencies of any column on the primary key.
3. Third Normal Form (3NF): Requires that the table is in 2NF, and all the columns are non-transitively dependent on the primary key, meaning no transitive dependencies exist (no column is dependent on a column that is not a primary key).
4. Boyce-Codd Normal Form (BCNF): A stronger version of 3NF where every determinant is a candidate key. This ensures that there are no anomalies due to functional dependencies.
5. Fourth Normal Form (4NF): Ensures that a table has no multi-valued dependencies other than a candidate key.

6. Fifth Normal Form (5NF): Addresses cases where information can be reconstructed from smaller pieces of information that can be meaningfully joined.

The Benefits of Database Normalization		
1	Reduction of Redundancy	Normalization removes duplicate data, which reduces the amount of storage required and eliminates the potential for inconsistencies.
2	Improved Data Integrity	By ensuring that data dependencies are logical, normalization enhances the accuracy and reliability of the data within the database.
3	Easier Maintenance	With a normalized database, updates, deletions, and insertions are more straightforward, reducing the risk of anomalies and errors.
4	Enhanced Query Performance	Although normalized databases might require more joins in queries, the reduction in redundant data can lead to faster query performance overall.
5	Consistency and Accuracy	Normalization helps in maintaining data consistency across the database by ensuring that each piece of data is stored only once.

Challenges and Considerations

While normalization brings many benefits, it is not without its challenges:

1. Complexity: Highly normalized databases can become complex, with many tables and relationships, making them harder to design and manage.

2. Performance Issues: In some cases, the need for numerous joins to retrieve data can impact performance. This trade-off needs to be balanced based on specific application requirements.

3. Over-Normalization: Going beyond the necessary level of normalization can lead to excessive complexity and performance overhead without significant benefits.

Practical Application of Normalization

The process of normalization typically involves a series of steps:

1. Analyze Data Requirements: Understand the data needs and the relationships between different data elements.

2. Define Primary Keys: Ensure each table has a primary key that uniquely identifies each record.

3. Eliminate Redundancies: Decompose the database into smaller tables to remove duplicate data.

4. Establish Relationships: Define foreign keys and relationships between tables to ensure data integrity.

5. Iterate Through Normal Forms: Apply the rules of each normal form iteratively to achieve the desired level of normalization.

Conclusion: Database normalization is a foundational aspect of relational database design that ensures data is stored efficiently and logically. By following the principles of normalization, database designers can create robust databases that minimize redundancy, maintain data integrity, and support efficient data management. While normalization can introduce complexity, the benefits of a well-structured, maintainable, and reliable database far outweigh the challenges. As such, understanding and applying normalization principles is essential for anyone involved in database management and design.

REFERENCES

1. Elmasri, R., & Navathe, S. B. (2015). Fundamentals of Database Systems (7th ed.). Pearson.
2. Silberschatz, A., Korth, H. F., & Sudarshan, S. (2010). Database System Concepts (6th ed.). McGraw-Hill.
3. Codd, E. F. (1970). A Relational Model of Data for Large Shared Data Banks. Communications of the ACM, 13(6), 377-387.
4. Date, C. J. (2003). An Introduction to Database Systems (8th ed.). Addison-Wesley.
5. Connolly, T., & Begg, C. (2014). Database Systems: A Practical Approach to Design, Implementation, and Management (6th ed.). Pearson.
6. Ramakrishnan, R., & Gehrke, J. (2003). Database Management Systems (3rd ed.). McGraw-Hill.