26-05-2023

**MAPPING CARDINALITY IN RELATIONAL DATABASE MANAGEMENT SYSTEMS**

Mapping cardinality, also called as cardinality ratio is a fundamental concept in database design, as it helps define the relationships between entities and determines how data is structured and stored in tables. By understanding and properly representing mapping cardinalities, database designers can establish accurate and efficient relationships between entities, enabling data retrieval and manipulation in a meaningful and consistent manner.

In the view of databases, cardinality refers to the uniqueness of data values that are contained in a column. High cardinality is nothing but the column contains a large percentage of totally unique values. Low cardinality is nothing but the column which has a lot of “repeats” in its data range.

Cardinality between the tables can be of type one-to-one, many-to-one or many-to-many.

Mapping cardinality is expressed as the number of entities to which another entity can be associated via a relationship set.

For binary relationship set there are entity set A and B then the mapping cardinality can be one of the following −

* One-to-one
* One-to-many
* Many-to-one
* Many-to-many

**Detailed explanation of the types of relationships possible in RDBMS:**

* **One-to-One (1:1) Cardinality:**
* In a one-to-one relationship, each record in one entity is associated with at most one record in another entity, and vice versa.
* This means that for every record in the first entity, there is only one corresponding record in the second entity, and vice versa.
* For example, in a database representing employees and their contact information, each employee may have only one corresponding contact record, and each contact record is associated with only one employee.
* **One-to-Many (1:N) Cardinality:**
* In a one-to-many relationship, each record in one entity can be associated with multiple records in another entity, but each record in the second entity is associated with at most one record in the first entity.
* For instance, in a database representing a customer and their orders, one customer may have multiple orders associated with them, but each order is associated with only one customer.
* **Many-to-Many (M:N) Cardinality:**
* In a many-to-many relationship, multiple records in one entity can be associated with multiple records in another entity.
* This type of relationship requires the use of an intermediate table, often referred to as a junction or associative table, to store the associations.
* For example, in a database representing students and courses, each student can be enrolled in multiple courses, and each course can have multiple students enrolled.
* **Many-to-One (N:1) Cardinality:**
* In a many-to-one relationship, multiple records from one entity can be associated with a single record in another entity.
* This means that for every record in the "many" entities, there is only one corresponding record in the "one" entity. However, the "one" entity can have multiple records associated with it from the "many" entities.
* For example, consider a database representing employees and departments. Each employee belongs to a specific department, but multiple employees can belong to the same department. In this case, the relationship between the "employees" entity and the "departments" entity is a many-to-one relationship.
* Each employee record (many) is associated with a single department record (one), while the department record (one) can be associated with multiple employee records (many).

**Advantages of cardinality mapping in RDBMS:**

Mapping cardinality in RDBMS offers advantages such as accurate representation of relationships, data integrity, query optimization, efficient data retrieval, effective data modelling, enhanced data analysis, and maintainability of database systems. By leveraging cardinality mapping, database designers and administrators can create robust, efficient, and reliable database solutions.

* **Accurate Representation of Relationships:**

Cardinality mapping allows for the accurate representation of relationships between entities in a database. It defines the nature and constraints of relationships, specifying how entities are connected and how data flows between them. By understanding and applying cardinality mapping, database designers can ensure that relationships are properly defined, maintained, and understood.

* **Data Integrity:**

Cardinality mapping helps enforce data integrity by defining the appropriate constraints on relationships. By specifying the cardinality of relationships, it ensures that data is linked correctly and consistently between entities. This helps prevent data anomalies, such as orphaned or inconsistent data, by enforcing rules that maintain the integrity and consistency of related data.

* **Query Optimization:**

Mapping cardinality information can assist the query optimizer in generating efficient execution plans. With knowledge of the cardinality of relationships, the query optimizer can make informed decisions on join algorithms, access paths, and indexing strategies. This optimization leads to improved query performance and faster data retrieval.

* **Data Retrieval:**

Mapping cardinality provides insights into how data is related, enabling efficient data retrieval. By understanding the cardinality of relationships, developers and database administrators can design queries and constructs that efficiently navigate through the database, retrieving the required data accurately and quickly. This results in improved data access and retrieval times.

* **Data Modelling and Design:**

Cardinality mapping supports effective data modelling and design. It helps database designers define the appropriate relationship types, choose the right keys, and determine the correct normalization levels for entities and tables. By understanding cardinality, designers can create well-structured, normalized, and efficient database schemas that accurately reflect the relationships between entities.

* **Enhanced Data Analysis:**

Mapping cardinality allows for more accurate and insightful data analysis. By understanding the relationships and cardinality between entities, analysts can perform various analytical tasks, such as aggregations, summaries, and comparisons, with confidence. Cardinality information helps analysts interpret and make sense of the data, enabling effective decision-making and business intelligence.

* **Maintainable and Scalable Database Systems:**

Cardinality mapping helps create maintainable and scalable database systems. It provides a clear understanding of relationships, allowing for easier modification, expansion, and evolution of the database schema. By properly mapping cardinality, database systems can adapt to changing requirements, accommodate growth, and support future enhancements with minimal disruptions.