Relational Algebra

Ortiz Vega Angelo
Course Code: CE3101
Name: Databases.
Academic Area of Computer Engineering.
Cartago, Costa Rica.

-Keywords: Relational Algebra, Relational Data Model, DataBases, Mapping ER to Relational Model, SQL, DBMS, DBA, Data Abstraction, Data Model, Database Schema, Schema Diagram, Snapshot, Database Languages.

-Content:

Relational Algebra

- It is the set of basic operations of the relational model.
- These operations enable the user to specify data retrieval requests through expressions of relational algebra.
- The result of a petition is a new relationship that is formed from one or more relationships.
- A sequence of relational operators is known as relational algebra expression.

Operators can be divided into two groups:

- Operations that belong to the mathematical theory of sets. For example:
  - UNION
  - INTERSECTION
  - SET DIFFERENCE
  - CARTESIAN PRODUCT (CROSS PRODUCT).
- Specific operations of the relational model. By example:
  - SELECT
  - PROJECT
  - JOIN

These operations can be Unary or Binary. The unaria are applied on a single relation at a time, the binaries on two relationships. Additionally, operations were created to make summaries of data, called aggregate functions. They were added to relational algebra since their use is indispensable for many of the applications. The SQL language is based on relational algebra.

SELECT Operator

Used to select a subset of tuples from a relation that satisfy a selection condition. It is a filter that leaves only the rows that meet a specific condition. On the contrary, it can be defined as a filter that discards those that do not meet a specific condition.

Notation: $\sigma_{<select\ condition>}R$

PROJECT operator

Select certain columns and discard others. If the user is interested in retrieving only certain columns of a relation, the PROJECT operator must be used. It can be understood as a vertical partition of the tables.

Notation: $\Pi_{<attribtue\ list>}R$

RENAME operator

To rename the columns of a resulting relation, the RENAME operator can be used. It is a unary operator that renames the relation or the attributes.

Notation: $\rho_{S<B_1,B_2,...B_n>}R$

UNION operator

It is applied on sets. Join the tuples of two sets. A UNION operation applied on R and S, includes the tuples that appear in R or appear in S, eliminating duplicates.

Notation: $R \cup S$

INTERSECTION operator

It is applied on sets. Generate a relation that includes the tuples that are in R and S. Relationships must be compatible.

Notation: $R \cap S$

MINUS operator

It is applied on sets. Generates a relation that includes all the tuples that are in R but that are not in S. The relations must be compatible.

Notation: $R - S$

CARTESIAN PRODUCT operator

Also known as CROSS PRODUCT or CROSS JOIN, it is a binary set operator, but relationships do not have to be compatible as in UNION. Combine all tuples of one set with the tuples of another set.

Notation: $R\ x\ S$

JOIN operator

It is a relational binary operator. Integrate the CARTESIAN PRODUCT operation with SELECT. Referential integrity plays a fundamental role in this operation.

Notation: $R \bowtie_{<join\ condition>} S$

Given two relations R (A1, A2,…, An) and S (B1, B1,…, B1), the result of the JOIN is a relation of the form:

Q (A1, A2,…, An, B1, B2,…, Bn)

Where Q has a tuple for each combination of tuples that satisfy the

JOIN condition. This is the main difference with the CARTESIAN PRODUCT, the JOIN only retrieves the rows that match the condition.

NATURAL JOIN

Omits one of the columns used to map relationships.

OUTER JOIN

The JOIN operation only retrieves the rows that “tie” or “match” between two relationships and specifies a condition of JOIN.

In the JOIN then, any row that does not meet the condition or that has NULL in the JOIN column is excluded from the result.

The OUTER JOIN operation allows the user to keep the tuples in R, S or both, regardless of whether or not they have a corresponding tuple in the other relation.

LEFT OUTER JOIN

It is usually known as a LEFT JOIN. Keeps all tuples in the relation on the left. The tuples of S that do not have a corresponding tuple in R are filled with NULL.

RIGHT OUTER JOIN

It is commonly known as the RIGHT JOIN. Keeps all tuples in the relation on the right. The tuples of R that do not have a corresponding tuple in S are filled with NULL.

FULL OUTER JOIN

It is usually known as a FULL JOIN. Maintains the tuples of the relation on the left and the relation of the right when there are no tuples that match. Tuples are filled with NULL when there is no match in either of the two relations.

Additional Operators

Generalized Projection

Works the same as the PROJECT operator. Allows you to use functions in the attribute list, generating attributes that do not exist in the original relationship. It is denoted as $\pi F1, F2,\ldots, Fn\ (R)$ where F1, F2,…, Fn are functions that are applied to each of the attributes corresponding R.

Aggregate and Grouping Functions

Some user queries cannot be expressed with the basic relational algebra. These queries include the use of aggregate functions mathematics. These functions are used in simple statistical queries that total tuples:

- AVERAGE
- MAXIMUM
- MINIMUM
- COUNT