Relational Algebra Concept

- **Relational Algebra** is the basic set of operators for the relational model.
- Relational algebra is a procedural query language, which takes instances of relations as **input** and produces instances of relations as **output**.
- It uses operators to perform queries.
- These operators enable a user to specify Quires.

Cont.....

- The result of an operation is a new relation , which may have been formed from one or more input relations.
- Relational algebra mainly provides a theoretical foundation for relational databases and SQL.
- As it is pure mathematics, there is no use of English Keywords in Relational Algebra and operators are represented using symbols.

Fundamental Operators of RA

- The fundamental operators of RA are as follows:

1. **Select**, **Project**, **Rename** (Unary Relational Algebra Operators)
2. **Union**, **Intersection**, **Set Deference**, **(**Relational Algebra Operators From Set Theory)
3. **Cartesian product**
4. **Joins** (Binary Relational Operators)

The operators have their own symbols.

Operation	Symbol
Projection	π
Selection	σ
Renaming	ρ
Union	$\cup$
Intersection	$\cap$
Assignment	$\leftarrow$

Operation	Symbol
Cartesian product	$\times$
Join	$\bowtie$
Left outer join	$\mathbin{⟕}$
Right outer join	$\mathbin{⟖}$
Full outer join	$\mathbin{⟗}$
Semijoin	$\ltimes$

Select Operator

Select Operator (σ): Select Operator is used to select a sub set of the tuples from a relation based on a selection condition.

- The selection condition acts as a filter
- Keeps only those tuples that satisfy the qualifying condition.

- **Notation − σ(r): -** Where **σ** stands for selection predicate and **r** stands for relation.

Select Operations: Examples

1- $\sigma_{subject = "database"}$**(Books)**

Output: - Selects tuples from books where subject is 'database'.

2- $\sigma_{subject = "database" \text{ and } price = "450"}$**(Books)**

Output: - Selects tuples from books where subject is 'database' and 'price' is 450.

3- $\sigma_{subject = "database" \text{ and } price = "450" \text{ or } year > "2016"}$**(Books)**

Output : - Selects tuples from books where subject is 'database' and 'price' is 450 or those books which published after 2016.

Relational Algebra and SQL----------Selection

Same table E (for EMPLOYEE)

<table>
<tr><th>SQL</th><th>Result</th><th>Relational algebra</th></tr>
<tr>
<td>

```
select *
from E
where salary < 200
```

</td>
<td>
<table>
<tr><th>nr</th><th>name</th><th>salary</th></tr>
<tr><td>1</td><td>John</td><td>100</td></tr>
<tr><td>7</td><td>Tom</td><td>100</td></tr>
</table>
</td>
<td>$\mathbf{SELECT}_{salary < 200}(E)$</td>
</tr>
<tr>
<td>

```
select *
from E
where salary < 200
and nr >= 7
```

</td>
<td>
<table>
<tr><th>nr</th><th>name</th><th>salary</th></tr>
<tr><td>7</td><td>Tom</td><td>100</td></tr>
</table>
</td>
<td>$\mathbf{SELECT}_{salary < 200 \text{ and } nr >= 7}(E)$</td>
</tr>
</table>

Select Operator Properties

- A cascade of select operator may be replaced by a single selection with a conjunction of all the condition.

$$\sigma_{<cond1>}(\sigma_{<cond2>}(\sigma_{<cond3>}(R)) = \sigma_{<cond1> \text{ AND } <cond2> \text{ AND } <cond3>}(R)$$

- The number of tuples in the Output of select is less than or (equal) to the number of tuples in the input relation R.

Project Operator

Project Operator (∏): - It projects column(s) that satisfy a given predicate.

- PROJECT Operator is denoted by π pi)
- PROJECT creates a vertical partitioning
- Notation: - $\prod_{A1, A2, An} (r)$
- Where A_1, A_2 , A_n are attribute names of relation **r**.
- Duplicate rows are automatically eliminated.

➢ **Notation – ∏ <attribute list> (R)** Where π stands for Projection predicate and **R** stands for relation.

Project Operator's Examples

1- $\prod_{subject,\ author}$ (Books)

Selects and projects columns named as subject and author from the relation Books.

2- $\prod_{name,\ fname,salary}$ (Employee)

Select and project columns named as name, fname, and salary from the relation Employee.

3- $\prod_{gender\ ,\ address}$ (Student)

Select and project columns named as gender and address from the relation Student

Relational Algebra and SQL---------Projection

nr	name	salary
1	John	100
5	Sarah	300
7	Tom	100

The table E (for EMPLOYEE)

SQL	Result	Relational algebra
`select salary from E`	salary: 100, 300	$\mathbf{PROJECT}_{\text{salary}}(E)$
`select nr, salary from E`	nr, salary: 1 100; 5 300; 7 100	$\mathbf{PROJECT}_{\text{nr, salary}}(E)$

Note: - there are no duplicate rows in the result.

Project Operator Properties

- The number of tuples in the result of projection is always less or equal to the number of tuples in R.
- If the list of attributes include a key of R, then the number of tuples in the result of Project is equal to the number of tuples in R.
- The project operator removes any duplicate tuples.

Differences B/W Select and Project Operators

- With the selection operator you specify which rows you want but with the projection operator you specify which columns you want.
- If the user is interested in selecting the values of a few attributes, then one should go for **Project** Operator.
- If the user is interested in selecting the values of all attributes rather than selection all attributes, , then one should go for **Select** Operator.

Combination of Selection and Projection

<table>
<tr><th>SQL</th><th>Result</th><th>Relational algebra</th></tr>
<tr>
<td>

```
select name,
salary from
E where
salary < 200
```

</td>
<td>
<table>
<tr><th>name</th><th>salary</th></tr>
<tr><td>John</td><td>100</td></tr>
<tr><td>Tom</td><td>100</td></tr>
</table>
</td>
<td>π name, salary (σ salary < 200(E))</td>
</tr>
</table>

Rename Operator

Rename Operation (ρ): -The rename operator allows us to rename the output relation.

- In some cases, we may want to **Rename** the attributes of a relation or the relation name or both.
- 'rename' operator is denoted with small Greek letter **rho *ρ***.
- **Notation: - ρ x (E)** Where the result of expression E is saved with name of x.

Example of Rename Operator

1- $\rho_{s(B1, B2,\ldots\ldots,Bn)}(R)$

- **The relation name to S, and**
- **The column (attribute) names to B1,B2,…….Bn**

2- $\rho_s(R)$ changes:

- **The relation name only to S**

3- $\rho_{(B1,B2,\ldots\ldots,Bn)}(R)$ changes

- **The column (attribute) names to B1, B2, …….,Bn**

The table **E** (for **EMPLOYEE**)

nr	name	dept
1	Bill	A
2	Sarah	C
3	John	A

The table **D** (for **DEPARTMENT**)

nr	name
A	Marketing
B	Sales
C	Legal

We want to join and Rename these tables

SQL	Result	Relational algebra

`select *`
`from E as E(enr, ename, dept),`
`  D as D(dnr, dname)`
`where dept = dnr`
	enr | ename | dept | dnr | dname
	1 | Bill | A | A | Marketing
	2 | Sarah | C | C | Legal
	3 | John | A | A | Marketing
		$(\mathbf{RENAME}_{(enr,\ ename,\ dept)}(E))\ \mathbf{JOIN}_{dept\ =\ dnr}\ (\mathbf{RENAME}_{(dnr,\ dname)}(D))$

`select *`
`from E, D`
`where dept = D.nr`
	nr | name | dept | nr | name
	1 | Bill | A | A | Marketing
	2 | Sarah | C | C | Legal
	3 | John | A | A | Marketing
		$E\ \mathbf{JOIN}_{dept\ =\ D.nr}\ D$


Q & A time
© Presentation-Process.com