Special Relational Operations

Detailed Syllabus

Special Relational Operations

- Select or Restrict
- Project
- Join
- Different Types of Join
  - Theta join
  - Equi-join
  - Natural join
  - Outer join
- Divide

Selection (or Restriction)

- **σpredicate (R)**
  - **Works on a single relation R and defines a relation that contains only those tuples**
  - **(rows) of R that satisfy the specified**
  - **condition (*predicate*).**

  - **Produces a horizontal subset of a table.**

Example - Selection (or Restriction)

- List all employees working for department No. 7.

Employee

E-No	E-Name	D-No
179	Silva	7
857	Perera	4
342	Dias	7

→

Sales Employee

E-No	E-Name	D-No
179	Silva	7
342	Dias	7

$$\text{Sales-Emp} = \sigma_{\text{D-No}=7}\ (\text{Employee})$$

Projection

- **Πcol1, . . . , coln(R)**
  - Works on a single relation R and defines a relation that contains a vertical subset of R, extracting the values of specified attributes and eliminating duplicates.
  - Produces a vertical subset of a table

Example - Projection

- Produce a list of employees showing only E-No and E-Name.

Employee

E-No	E-Name	D-No
179	Silva	7
857	Perera	4
342	Dias	7

→

Employee Names

E-No	E-Name
179	Silva
857	Perera
342	Dias

$$\text{Emp-Names} = \pi_{\text{E-No, E-Name}}\ (\text{Employee})$$

Sequence of Operations

We may want to apply several relational algebra operations one after the other.

e.g. Retrieve the first name, last name and salary of all employees who work in department no. 5

We can write the operation as a single relational algebra operation by nesting the operations

e.g.

$$\pi \text{Fname,Lname,Salary} \ (\sigma \text{Dno=5} \ (\text{ Employee}))$$

Sequence of Operations

Or we can apply one operation at a time and create intermediate result relations.

DEP5_EMPS ← (σDno=5 (Employee))

RESULT ← πFname,Lname,Salary(DEP5_EMPS)

Join Operations

- Join is a derivative of Cartesian product.

- Equivalent to performing a Selection, using join
  predicate as selection formula, over Cartesian product
  of the two operand relations.

Join Operations

One of the most difficult operations to implement

efficiently in an RDBMS and one reason why

RDBMSs have intrinsic performance problems.

The ⋈ general form of a join operation on two

relations R(A1,A2,….,An) and S(B1,B2,….,Bm) is

R **<join condition> S**

Join Operations

- A general join condition is of the form

  <condition> AND <condition> AND….. AND <condition>

- Where each condition is of the form Ai θ Bj.
  - Ai is an attribute of R, Bj is an attribute of S.
  - Ai and Bj have the same domain and
  - θ is one of the comparison operators {=, <, <=,>,>=,≠}
  - A join operation with such a general join condition is called theta join.

Join Operations

- The most common use of JOIN involves join conditions with equality (=) comparisons only.

- This is known as EQUIJOIN.

- In the result of an EQUIJOIN we always have one or more pairs of attributes that have identical values in every tuple.

Example - Join Operations

Dept_Mgr

DName	DNo	MgrId	….	Fname	Lname	EId	…
Sales	5	123	….	aaa	xxx	123	…
Acct	4	345	….	bbb	yyy	345	…
Admin	1	678	….	ccc	zzz	678	…

A new operation called **Natural join** denoted by * is created to get rid of the second attribute.

Join Operations

- A new operation called **Natural join** denoted by * is created to get rid of the second attribute.
- Outer join
- **To display rows in the result that do not have matching values in the join column, use Outer join.**
- **R ⟕ S**
  - (Left) outer join is join in which tuples from R that do not have matching values in common columns of S are also included in result relation.

Join Operations

- Outer join example

List all employee names and the departments that they manage if they happen to manage a department.

Temp ← Employee $⟕_{\text{empId = mgrId}}$ Department

Result ← $\pi_{\text{Fname, Lname,Dname}}$(Temp)

Division Operation

- R ÷ S
  - Defines a relation over the attributes C that consists of set of tuples from R that match combination of *every* tuple in S.

R

A	B
a1	b1
a2	b1
a3	b1
a4	b1
a1	b2
a3	b2
a2	b3
a3	b3
a4	b3
a1	b4
a2	b4
a3	b4

S

A
a1
a2
a3

T

B
b1
b4

Division Operation

- Division example

Retrieve the names of employees who work on all projects that ‘Sunil Silva’ works on.

Sunil ← σFname=‘Sunil’ and Lname=‘Silva’ (Employee)

Sunil_Pnos ← πPno(Works_on ⋈ empId=EssnSunil)

Temp ← πEssn,Pno(Works_on)

Result1(empId) ← Temp ÷ Sunil_Pnos

Result ← πFname,Lname(Result1 * Employee)