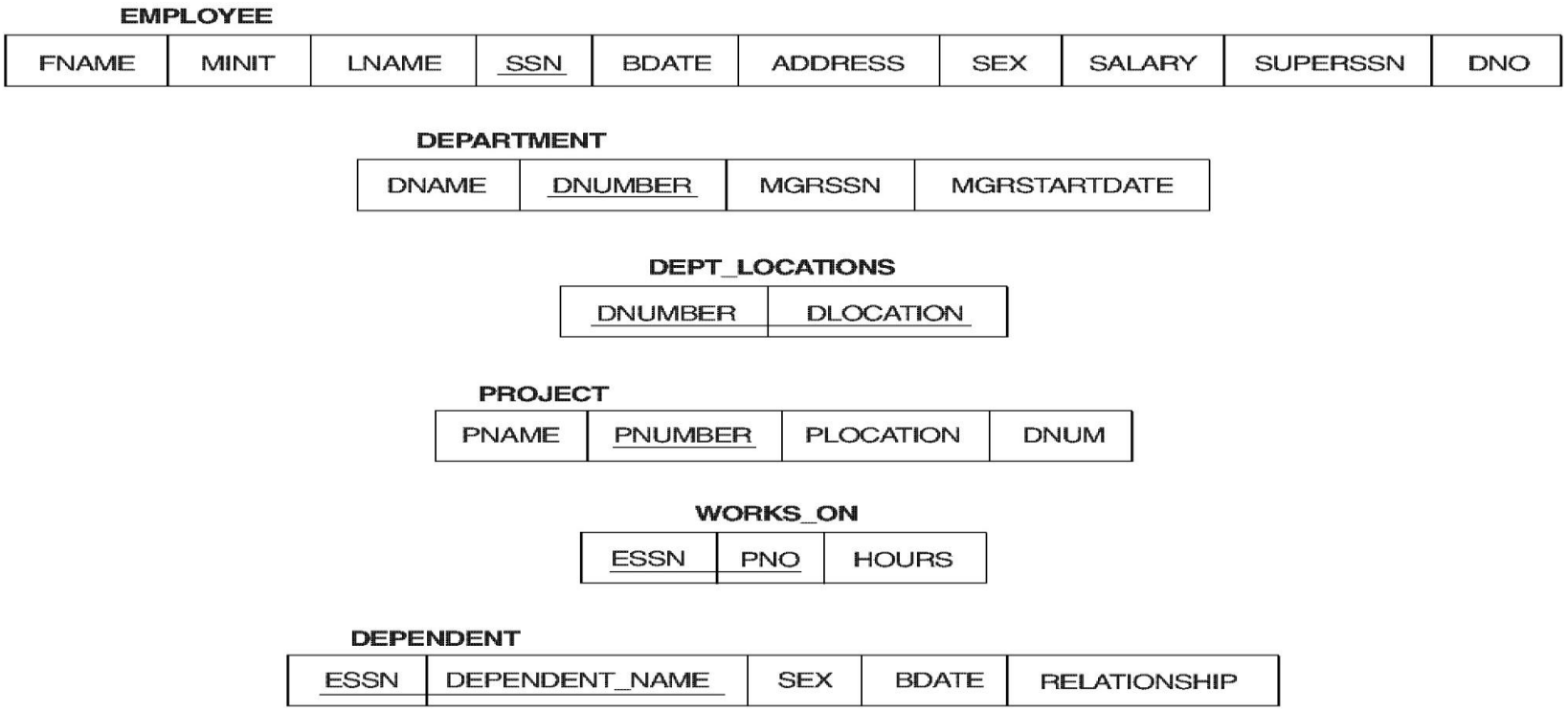
Unit IV – Relational Databases and Query Language

Chapter 8 – The Relational Algebra and Calculus

Database State for COMPANY

All examples discussed below refer to the COMPANY database shown here.

Figure 7.5 Schema diagram for the COMPANY relational database schema; the primary keys are underlined.



Relational Algebra

- The basic set of operations for the relational model is known as the relational algebra. These operations enable a user to specify basic retrieval requests.
- The result of a retrieval is a new relation, which may have been formed from one or more relations. The **algebra operations** thus produce new relations, which can be further manipulated using operations of the same algebra.
- A sequence of relational algebra operations forms a **relational algebra expression**, whose result will also be a relation that represents the result of a database query (or retrieval request).

Unary Relational Operations

SELECT Operation

SELECT operation is used to select a *subset* of the tuples from a relation that satisfy a **selection condition**. It is a filter that keeps only those tuples that satisfy a qualifying condition – those satisfying the condition are selected while others are discarded.

In general, the select operation is denoted by

σ <selection condition>(R)

where the symbol **σ** (sigma) is used to denote the select operator, and the selection condition is a Boolean expression specified on the attributes of relation R

Example: To select the EMPLOYEE tuples whose department number is four or those whose salary is greater than $30,000 the following notation is used:

σ DNO = 4 (EMPLOYEE)

σ SALARY > 30,000 (EMPLOYEE)

SELECT Operation Properties:

- The SELECT operation **σ** <selection condition>(R) produces a relation S that has the same schema as R
- The SELECT operation **σ** is **commutative;** i.e.,

  σ <condition1>(**σ** < condition2> (R)) = **σ** <condition2> (**σ** < condition1> (R))
- A cascaded SELECT operation **may be applied in any order;** i.e.,

  σ <condition1>(**σ** < condition2> (**σ** <condition3> (R))

  = **σ** <condition2> (**σ** < condition3> (**σ** < condition1> (R)))
- A cascaded SELECT operation may be replaced by a single selection with a conjunction of all the conditions**;** i.e.,

  σ <condition1>(**σ** < condition2> (**σ** <condition3> (R))

  = **σ** <condition1> AND < condition2> AND < condition3> (R)))

$$\sigma_{(DNO=4\ AND\ SALARY>25000)\ OR\ (DNO=5\ AND\ SALARY>30000)}(EMPLOYEE)$$

FNAME	MINIT	LNAME	SSN	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DNO
Franklin	T	Wong	333445555	1955-12-08	638 Voss,Houston,TX	M	40000	888665555	5
Jennifer		Wallace	987654321	1941-06-20	291 Berry,Bellaire,TX	F	43000	888665555	4
Ramesh		Narayan	666884444	1962-09-15	975 FireOak,Humble,TX	M	38000	333445555	5

Unary Relational Operations (cont.)

- **PROJECT Operation**

This operation selects certain *columns* from the table and discards the other columns. The PROJECT creates a vertical partitioning – one with the needed columns (attributes) containing results of the operation and other containing the discarded Columns.

Example: To list each employee's first and last name and salary, the following is used:

$$\pi_{\text{LNAME, FNAME,SALARY}}(\text{EMPLOYEE})$$

The general form of the project operation is

$$\pi_{<\text{attribute list}>}(R)$$

where π (pi) is the symbol used to represent the project operation and <attribute list> is the desired list of attributes from the attributes of relation R.

The project operation *removes any duplicate tuples*, so the result of the project operation is a set of tuples and hence a valid relation.

ROJECT Operation Properties:

- The number of tuples in the result of projection $\pi_{\langle list \rangle}(R)$ is always less or equal to the number of tuples in R.

- If the list of attributes includes a key of R, then the number of tuples is equal to the number of tuples in R

- $\pi_{\langle list1 \rangle}(\pi_{\langle list2 \rangle}(R)) = \pi_{\langle list1 \rangle}(R)$ as long as <list1> contains the attributes in <list2>

Unary Relational Operations (cont.)

$$\pi_{\text{SEX, SALARY}}(\text{EMPLOYEE})$$

SEX	SALARY
M	30000
M	40000
F	25000
F	43000
M	38000
M	25000
M	55000

$$\pi_{\text{LNAME, FNAME, SALARY}}(\text{EMPLOYEE})$$

LNAME	FNAME	SALARY
Smith	John	30000
Wong	Franklin	40000
Zelaya	Alicia	25000
Wallace	Jennifer	43000
Narayan	Ramesh	38000
English	Joyce	25000
Jabbar	Ahmad	25000
Borg	James	55000

Unary Relational Operations (cont.)

Rename Operation

If we want to apply several relational algebra operations one after the other, either we can

- write the operations as a single **relational algebra expression** by nesting the operations, OR
- apply one operation at a time and create **intermediate result relations**.
- In the latter case, we must give names to the relations that hold the intermediate results.

Example: To retrieve the first name, last name, and salary of all employees who work in department number 5, we must apply a select and a project operation. We can write a single relational algebra expression as follows:

$\pi_{FNAME, LNAME, SALARY}(\sigma_{DNO=5}(EMPLOYEE))$

OR We can explicitly show the sequence of operations, giving a name to each intermediate relation:

$DEP5\_EMPS \leftarrow \sigma_{DNO=5}(EMPLOYEE)$

$RESULT \leftarrow \pi_{FNAME, LNAME, SALARY}(DEP5\_EMPS)$

The rename operator is **ρ**

The general Rename operation can be expressed by any of the following forms:

- **ρ** $_{S (B1, B2, \ldots, Bn)}$ (R) is a renamed relation S based on R with column names B_1, B2, ……..B_n.
- **ρ** $_S$ (R) is a renamed relation S based on R (which does not specify column names).
- **ρ** $_{(B1, B2, \ldots, Bn)}$ (R) is a renamed relation with column names B_1, B2, …..B_n which does not specify a new relation name.

Unary Relational Operations (cont.)

Figure 8.2

Results of a sequence of operations. (a) $\pi_{Fname, Lname, Salary}(\sigma_{Dno=5}(EMPLOYEE))$.
(b) Using intermediate relations and renaming of attributes.

(a)

Fname	Lname	Salary
John	Smith	30000
Franklin	Wong	40000
Ramesh	Narayan	38000
Joyce	English	25000

(b)

TEMP

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston,TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston,TX	M	40000	888665555	5
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble,TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5

R

First_name	Last_name	Salary
John	Smith	30000
Franklin	Wong	40000
Ramesh	Narayan	38000
Joyce	English	25000

Relational Algebra Operations from Set Theory

UNION Operation

The result of this operation, denoted by R ∪ S, is a relation that includes all tuples that are either in R or in S or in both R and S. Duplicate tuples are eliminated.

Example: To retrieve the social security numbers of all employees who either work in department 5 or directly supervise an employee who works in department 5, we can use the union operation as follows:

$$\text{DEP5\_EMPS} \leftarrow \sigma_{\text{Dno}=5}(\text{EMPLOYEE})$$
$$\text{RESULT1} \leftarrow \pi_{\text{Ssn}}(\text{DEP5\_EMPS})$$
$$\text{RESULT2(Ssn)} \leftarrow \pi_{\text{Super\_ssn}}(\text{DEP5\_EMPS})$$
$$\text{RESULT} \leftarrow \text{RESULT1} \cup \text{RESULT2}$$

RESULT1

Ssn
123456789
333445555
666884444
453453453

RESULT2

Ssn
333445555
888665555

RESULT

Ssn
123456789
333445555
666884444
453453453
888665555

Figure 8.3
Result of the UNION operation RESULT ← RESULT1 ∪ RESULT2.

The union operation produces the tuples that are in either RESULT1 or RESULT2 or both. The two operands must be "type compatible".

Type Compatibility

- The operand relations $R_1(A_1, A_2, ..., A_n)$ and $R_2(B_1, B_2, ..., B_n)$ must have the
  - ✓ same number of attributes, and
  - ✓ compatible domains of corresponding attributes; that is, $dom(A_i) = dom(B_i)$ for i=1, 2, ..., n.

- The resulting relation for $R_1 \cup R_2$, $R_1 \cap R_2$, or $R_1 - R_2$ has the same attribute names as the *first* operand relation R1 (by convention).

Relational Algebra Operations from Set Theory (cont.)

INTERSECTION Operation

The result of this operation, denoted by R ∩ S, is a relation that includes all tuples that are in both R and S. The two operands must be "type compatible"

Example: The result of the intersection operation (figure below) includes only those who are both students and instructors.

(a) STUDENT

Fn	Ln
Susan	Yao
Ramesh	Shah
Johnny	Kohler
Barbara	Jones
Amy	Ford
Jimmy	Wang
Ernest	Gilbert

INSTRUCTOR

Fname	Lname
John	Smith
Ricardo	Browne
Susan	Yao
Francis	Johnson
Ramesh	Shah

Fn	Ln
Susan	Yao
Ramesh	Shah

STUDENT ∩ INSTRUCTOR.

Relational Algebra Operations from Set Theory (cont.)

Set Difference (or MINUS) Operation

The result of this operation, denoted by R - S, is a relation that includes all tuples that are in R but not in S. The two operands must be "type compatible".

Example: The figure shows the names of students who are not instructors, and the names of instructors who are not students.

STUDENT

Fn	Ln
Susan	Yao
Ramesh	Shah
Johnny	Kohler
Barbara	Jones
Amy	Ford
Jimmy	Wang
Ernest	Gilbert

INSTRUCTOR

Fname	Lname
John	Smith
Ricardo	Browne
Susan	Yao
Francis	Johnson
Ramesh	Shah

(d)

Fn	Ln
Johnny	Kohler
Barbara	Jones
Amy	Ford
Jimmy	Wang
Ernest	Gilbert

(e)

Fname	Lname
John	Smith
Ricardo	Browne
Francis	Johnson

(d) STUDENT – INSTRUCTOR. (e) INSTRUCTOR – STUDENT.

Relational Algebra Operations from Set Theory (cont.)

Notice that both union and intersection are;

- *commutative operations;* that is

$$\mathbf{R} \cup \mathbf{S} = \mathbf{S} \cup \mathbf{R}, \text{ and}$$

$$\mathbf{R} \cap \mathbf{S} = \mathbf{S} \cap \mathbf{R}$$

- *associative operations;* that is

$$\mathbf{R} \cup (\mathbf{S} \cup \mathbf{T}) = (\mathbf{R} \cup \mathbf{S}) \cup \mathbf{T}, \text{ and}$$

$$(\mathbf{R} \cap \mathbf{S}) \cap \mathbf{T} = \mathbf{R} \cap (\mathbf{S} \cap \mathbf{T})$$

Therefore, both union and intersection can be treated as n-ary operations applicable to any number of relations

❖ The minus operation is *not commutative;* that is, in general

$$\mathbf{R} - \mathbf{S} \neq \mathbf{S} - \mathbf{R}$$

Relational Algebra Operations from Set Theory (cont.)

CARTESIAN or cross product Operation

- This operation is used to combine tuples from two relations in a combinatorial fashion.

  In general, the result of $R(A_1, A_2, \ldots, A_n)$ x $S(B_1, B_2, \ldots, B_m)$ is a relation Q with
  - ✓ degree n + m attributes $Q(A_1, A_2, \ldots, A_n, B_1, B_2, \ldots, B_m)$, in that order.
  - ✓ one tuple for each combination of tuples—one from R and one from S.
- If R has n_R tuples (denoted as $|R| = n_R$), and S has n_S tuples, then

  $| R \times S |$ will have $n_R * n_S$ tuples.
- The two operands do NOT have to be "type compatible”

Example: To retrieve the list of each female employee's dependents

$$\text{FEMALE\_EMPS} \leftarrow \sigma_{\text{Sex='F'}}(\text{EMPLOYEE})$$
$$\text{EMPNAMES} \leftarrow \pi_{\text{Fname, Lname, Ssn}}(\text{FEMALE\_EMPS})$$
$$\text{EMP\_DEPENDENTS} \leftarrow \text{EMPNAMES} \times \text{DEPENDENT}$$

The CARTESIAN PRODUCT (CROSS PRODUCT) operation.

FEMALE_EMPS

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
Alicia	J	Zelaya	999887777	1968-07-19	3321Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291Berry, Bellaire, TX	F	43000	888665555	4
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5

EMPNAMES

Fname	Lname	Ssn
Alicia	Zelaya	999887777
Jennifer	Wallace	987654321
Joyce	English	453453453

EMP_DEPENDENTS

Fname	Lname	Ssn	Essn	Dependent_name	Sex	Bdate	...
Alicia	Zelaya	999887777	333445555	Alice	F	1986-04-05	...
Alicia	Zelaya	999887777	333445555	Theodore	M	1983-10-25	...
Alicia	Zelaya	999887777	333445555	Joy	F	1958-05-03	...
Alicia	Zelaya	999887777	987654321	Abner	M	1942-02-28	...
Alicia	Zelaya	999887777	123456789	Michael	M	1988-01-04	...
Alicia	Zelaya	999887777	123456789	Alice	F	1988-12-30	...
Alicia	Zelaya	999887777	123456789	Elizabeth	F	1967-05-05	...
Jennifer	Wallace	987654321	333445555	Alice	F	1986-04-05	...
Jennifer	Wallace	987654321	333445555	Theodore	M	1983-10-25	...
Jennifer	Wallace	987654321	333445555	Joy	F	1958-05-03	...
Jennifer	Wallace	987654321	987654321	Abner	M	1942-02-28	...
Jennifer	Wallace	987654321	123456789	Michael	M	1988-01-04	...
Jennifer	Wallace	987654321	123456789	Alice	F	1988-12-30	...
Jennifer	Wallace	987654321	123456789	Elizabeth	F	1967-05-05	...
Joyce	English	453453453	333445555	Alice	F	1986-04-05	...
Joyce	English	453453453	333445555	Theodore	M	1983-10-25	...
Joyce	English	453453453	333445555	Joy	F	1958-05-03	...
Joyce	English	453453453	987654321	Abner	M	1942-02-28	...
Joyce	English	453453453	123456789	Michael	M	1988-01-04	...
Joyce	English	453453453	123456789	Alice	F	1988-12-30	...
Joyce	English	453453453	123456789	Elizabeth	F	1967-05-05	...