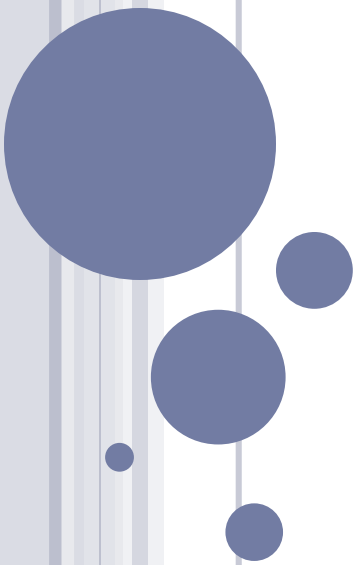


Relational Algebra

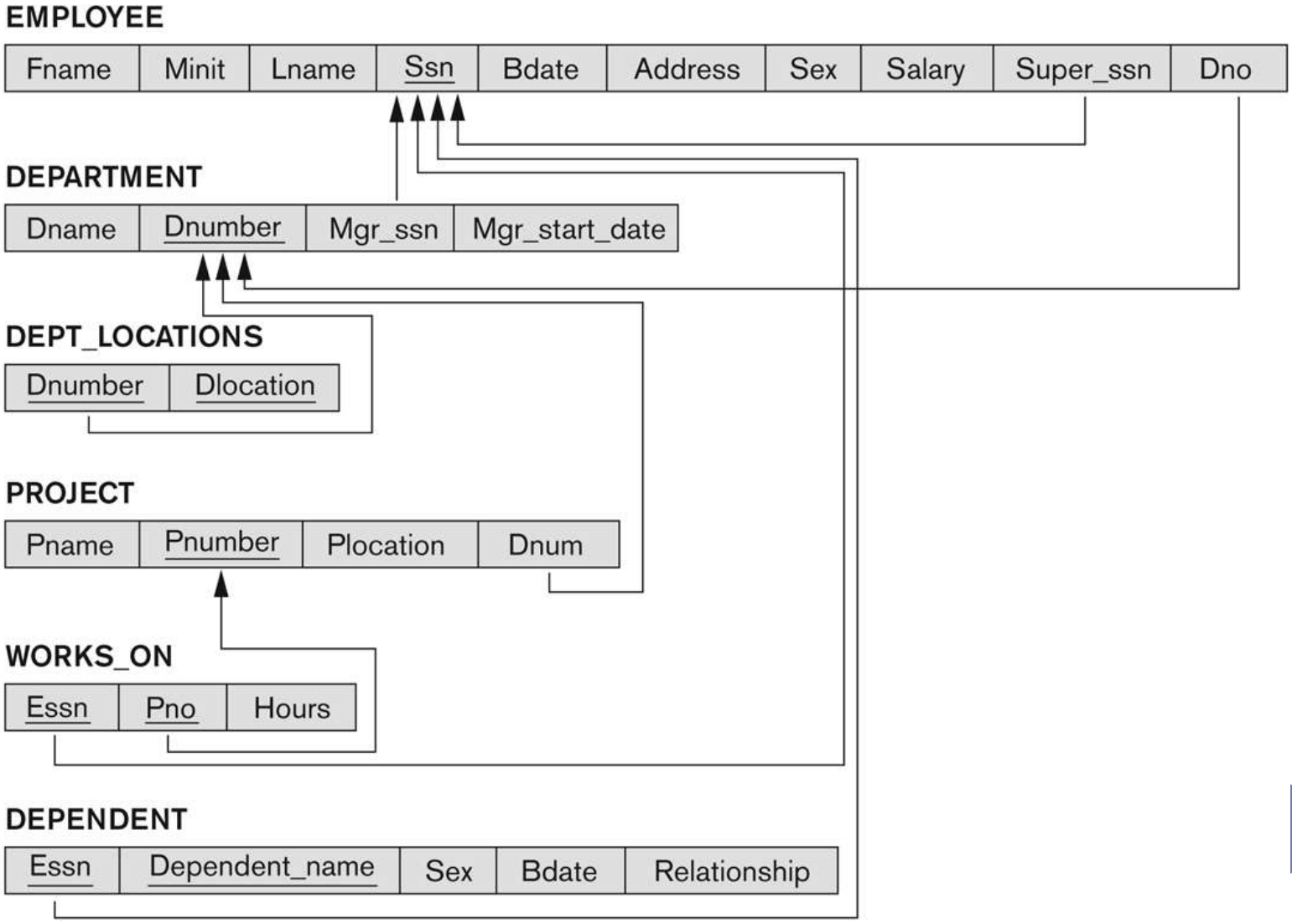
Relational Algebra

- There are two types of operations in RDBMS
  - Retrieval
  - Update

- The set of operations for specifying **retrieval requests** (or **queries**) in relational model is called Relational Algebra.

- A sequence of relational algebra operations forms a **relational algebra expression**.

Company Database Considered in Examples



SELECT OPERATION(UNARY OPERATION)

- This operation selects a subset of tuples from a relation that satisfy a selection condition.
- Select is denoted by : $\sigma_{\text{<selection condition>}}(R)$

EMPLOYEE

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
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
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
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date
Research	5	333445555	1988-05-22
Administration	4	987654321	1995-01-01
Headquarters	1	888665555	1981-06-19

DEPT_LOCATIONS

Dnumber	Dlocation
1	Houston
4	Stafford
5	Bellaire
5	Sugarland
5	Houston

EXAMPLES : SELECT OPERATION

- Select the employees whose department number is 4:

$$\sigma_{DNO = 4} (EMPLOYEE)$$

- Select all the projects in department 5
- Select the employees whose salary is greater than $35,000

EMPLOYEE

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
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
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
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date
Research	5	333445555	1988-05-22
Administration	4	987654321	1995-01-01
Headquarters	1	888665555	1981-06-19

DEPT_LOCATIONS

Dnumber	Dlocation
1	Houston
4	Stafford
5	Bellaire
5	Sugarland

WORKS_ON

Essn	Pno	Hours
123456789	1	32.5
123456789	2	7.5
666884444	3	40.0
453453453	1	20.0
453453453	2	20.0
333445555	2	10.0
333445555	3	10.0
333445555	10	10.0
333445555	20	10.0
999887777	30	30.0
999887777	10	10.0
987987987	10	35.0
987987987	30	5.0
987654321	30	20.0
987654321	20	15.0
888665555	20	NULL

PROJECT

Pname	Pnumber	Plocation	Dnum
ProductX	1	Bellaire	5
ProductY	2	Sugarland	5
ProductZ	3	Houston	5
Computerization	10	Stafford	4
Reorganization	20	Houston	1
Newbenefits	30	Stafford	4

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship
333445555	Alice	F	1986-04-05	Daughter
333445555	Theodore	M	1983-10-25	Son
333445555	Joy	F	1958-05-03	Spouse
987654321	Abner	M	1942-02-28	Spouse
123456789	Michael	M	1988-01-04	Son
123456789	Alice	F	1988-12-30	Daughter
123456789	Elizabeth	F	1967-05-05	Spouse

SELECT OPERATION

- Selection condition is a Boolean expression specified on the attributes of relation R
  - It can include boolean operators AND, OR, NOT applied on relational operators <, > <=,>=, !=, =
- Select σ is commutative:

  $\sigma_{<condition1>}(\sigma_{<condition2>}(R)) = \sigma_{<condition2>}(\sigma_{<condition1>}(R))$
- Cascade of Select operations

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

$\sigma_{(Dno=4 \textbf{ AND } Salary>25000) \text{ OR } (Dno=5 \textbf{ AND } Salary>30000)}(EMPLOYEE)$.

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5

PROJECT OPERATION (UNARY OPERATION)

- It selects a subset of columns from the relation.
- Denoted by $\pi_{<\text{attribute list}>}R$
- It removes duplicate tuples, the result of project is set of tuples

- **Example:**
  - RESULT← π LNAME, FNAME, SALARY (EMPLOYEE)
  - DN← π DNAME, DNUMBER (DEPARTMENT)

EMPLOYEE

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
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
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
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

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

Project Operation

- Project operation is *not* commutative
- $\pi_{\text{<list1>}}(\pi_{\text{<list2>}}(R)) = \pi_{\text{<list1>}}(R)$ as long as <list2> contains the attributes in <list1>

- No of Tuples in the result of projection $\boldsymbol{\pi}_{\text{<list>}}(R)$
  - less or equal to the number of tuples in R
  - If the list of attributes includes a *key* of R, then the no of is *equal* to the no of tuples in R

Relational Algebra Expressions

- We may want to apply several relational algebra operations one after the other
  1. We can write the operations as a single **relational algebra expression** by nesting the operations, or
  2. We can apply one operation at a time and create **intermediate result relations**.

Example: Sequence of Operations

- ***To retrieve the first name, last name, and salary of all employees who work in Department 5***

- Result of sequence of operations:
  - $\pi_{\text{FNAME, LNAME, SALARY}}(\sigma_{\text{DNO=5}}(\text{EMPLOYEE}))$

- Using intermediate relation:
  - $D5 \leftarrow \sigma_{\text{DNO=5}}(\text{EMPLOYEE})$
  - $\text{RESULT} \leftarrow \pi_{\text{FNAME, LNAME, SALARY}}(D5)$

Example of applying multiple operations and RENAME

(a)

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

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

$$D5 \leftarrow \sigma_{\text{DNO=5}}(\text{EMPLOYEE})$$

$$R(\text{First\_name}, \text{Last\_name}, \text{Salary}) \leftarrow \pi_{\text{Fname, Lname, Salary}}(D5)$$

(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

Figure 6.2
Results of a sequence of operations.
(a) $\pi_{\text{Fname, Lname, Salary}}(\sigma_{\text{Dno=5}}(\text{EMPLOYEE}))$.
(b) Using intermediate relations and renaming of attributes.

RENAME OPEARATION

- Rename operator is denoted by ρ (rho)
- Rename operation ρ can be expressed as:
  - $\rho_S(R)$ rename the *relation R* to S
  - $\rho_{(B1, B2, ..., Bn)}(R)$ rename the *attributes* to B1, B2, …..Bn
  - $\rho_{S\ (B1, B2, ..., Bn)}(R)$ rename relation R to S, *and* attributes to B1, B2, …..Bn
- Example:
  - $\rho_{\ RESULT\ (First\_Name, Last\_Name, Salary)}(D5)$

Union (Binary Operation)

- The result of $R \cup S$, 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
- The two relations R and S must be "type compatible" (or Union compatible)
  - R and S must have same number of attributes
  - Each pair of corresponding attributes must have same or compatible domains

UNION 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

$$DEP5\_EMPS \leftarrow \sigma_{DNO=5}(EMPLOYEE)$$
$$RESULT1 \leftarrow \pi_{SSN}(DEP5\_EMPS)$$
$$RESULT2(SSN) \leftarrow \pi_{SUPERSSN}(DEP5\_EMPS)$$
$$RESULT \leftarrow RESULT1 \cup RESULT2$$

EMPLOYEE

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
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
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
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

RESULT1

Ssn
123456789
333445555
666884444
453453453

RESULT2

Ssn
333445555
888665555

RESULT

Ssn
123456789
333445555
666884444
453453453
888665555

INTERSECTION AND SET DIFFERENCE (BINARY OPERATIONS)

- INTERSECTION operation: the result of R ∩ S, is a relation that includes all tuples that are in both R and S
- SET DIFFERENCE operation: the result of R – S, is a relation that includes all tuples that are in R but not in S
- Two relations R and S must be “type compatible”

Relational Algebra Operations from Set Theory

- Both $\cup$ and $\cap$ are *commutative* operations
  - $R \cup S = S \cup R$, and $R \cap S = S \cap R$
- Both $\cup$ and $\cap$ can be treated as n-ary operations
  - $R \cup (S \cup T) = (R \cup S) \cup T$
  - $(R \cap S) \cap T = R \cap (S \cap T)$
- Minus operation is not commutative
  - $R - S \neq S - R$

EXAMPLE TO ILLUSTRATE THE RESULT OF UNION, INTERSECT, AND DIFFERENCE

(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

(b)

Fn	Ln
Susan	Yao
Ramesh	Shah
Johnny	Kohler
Barbara	Jones
Amy	Ford
Jimmy	Wang
Ernest	Gilbert
John	Smith
Ricardo	Browne
Francis	Johnson

(c)

Fn	Ln
Susan	Yao
Ramesh	Shah

(d)

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

(e)

Fname	Lname
John	Smith
Ricardo	Browne
Francis	Johnson

Figure 6.4
The set operations UNION, INTERSECTION, and MINUS. (a) Two union-compatible relations. (b) STUDENT ∪ INSTRUCTOR. (c) STUDENT ∩ INSTRUCTOR. (d) STUDENT − INSTRUCTOR. (e) INSTRUCTOR − STUDENT.

CARTESIAN PRODUCT

- The result of Cartesian product of two relations R(A1, A2, . . ., An) x S(B1, B2, . . ., Bm) is given as:

  Result(A1, A2, . . ., An, B1, B2, . . ., Bm)

- Let $|R| = n_R$ and $|S| = n_S$, then $|R \times S| = n_R * n_S$
- R and S may NOT be "type compatible"
- *Cross Product is a meaningful operation only if it is followed by other operations*

Problem:
Retrieve a list of each female employee's dependents

$F \leftarrow \sigma_{SEX='F'}(EMPLOYEE)$
$EN \leftarrow \pi_{FNAME, LNAME, SSN}(F)$
$E\_DP \leftarrow EN \times DEPENDENT$
$A\_DP \leftarrow \sigma_{SSN=ESSN}(\mathbf{E\_DP})$
$R \leftarrow \pi_{FNAME, LNAME, DEPENDENT\_NAME}(A\_DP)$

Figure 6.5
The CARTESIAN PRODUCT (CROSS PRODUCT) operation.

F

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

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship
333445555	Alice	F	1986-04-05	Daughter
333445555	Theodore	M	1983-10-25	Son
333445555	Joy	F	1958-05-03	Spouse
987654321	Abner	M	1942-02-28	Spouse
123456789	Michael	M	1988-01-04	Son
123456789	Alice	F	1988-12-30	Daughter
123456789	Elizabeth	F	1967-05-05	Spouse

JOIN(Binary Operation)

- JOIN denoted by ⋈ *combine related tuples* from various relations
- JOIN combines CARTESIAN PRODECT and SELECT into a single operation
- General form of a join operation on two relations R(A1, A2, . . ., An) and S(B1, B2, . . ., Bm) is:

$$R \underset{\text{<join condition>}}{\bowtie} S$$

EXAMPLE OF JOIN OPERATION

- Retrieve the name of the manager of each department.

EMPLOYEE

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
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
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
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date
Research	5	333445555	1988-05-22
Administration	4	987654321	1995-01-01
Headquarters	1	888665555	1981-06-19

$$\text{DEPT\_MGR} \leftarrow \text{DEPARTMENT} \bowtie_{\text{MGRSSN=SSN}} \text{EMPLOYEE}$$

DEPT_MGR

Dname	Dnumber	Mgr_ssn	...	Fname	Minit	Lname	Ssn	...
Research	5	333445555	...	Franklin	T	Wong	333445555	...
Administration	4	987654321	...	Jennifer	S	Wallace	987654321	...
Headquarters	1	888665555	...	James	E	Borg	888665555	...

Complete Set of Relational Operations

- The set of operations including
  - SELECT σ,
  - PROJECT π ,
  - UNION $\cup$,
  - DIFFERENCE – ,
  - RENAME ρ, and
  - CARTESIAN PRODUCT X

  is called a *complete set* because any relational algebra expression can be expressed using these.

- For example:
  - $R \cap S = (R \cup S) - ((R - S) \cup (S - R))$
  - $R \bowtie_{\text{<join condition>}} S = \sigma_{\text{<join condition>}} (R \times S)$

EXAMPLE TO ILLUSTRATE THE RESULT OF UNION, INTERSECT, AND DIFFERENCE

(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

$$R \cap S = (R \cup S) - ((R - S) \cup (S - R))$$

(b)

Fn	Ln
Susan	Yao
Ramesh	Shah
Johnny	Kohler
Barbara	Jones
Amy	Ford
Jimmy	Wang
Ernest	Gilbert
John	Smith
Ricardo	Browne
Francis	Johnson

(c)

Fn	Ln
Susan	Yao
Ramesh	Shah

(d)

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

(e)

Fname	Lname
John	Smith
Ricardo	Browne
Francis	Johnson

Figure 6.4
The set operations UNION, INTERSECTION, and MINUS. (a) Two union-compatible relations. (b) STUDENT ∪ INSTRUCTOR. (c) STUDENT ∩ INSTRUCTOR. (d) STUDENT – INSTRUCTOR. (e) INSTRUCTOR – STUDENT.

SOME PROPERTIES OF JOIN

- Consider the following JOIN operation:
  - R(A1, A2, . . ., An) $\bowtie_{R.Ai=S.Bj}$ S(B1, B2, . . ., Bm)
  - Result is a relation Q with degree n + m attributes:
    - Q(A1, A2, . . ., An, B1, B2, . . ., Bm), in that order.
  - If R has n_R tuples, and S has n_S tuples, then no of tuples in join result < n_R * n_S .

THETA-JOIN

- The general case of JOIN operation is called a Theta-join: $R \bowtie_{theta} S$
- *Theta* is a boolean expression on the attributes of R and S; for example:
  - R.Ai<S.Bj AND (R.Ak=S.Bl OR R.Ap<S.Bq)
- Theta can have any comparison operators {=,≠,<,≤,>,≥,}

EQUI-JOIN

- EQUIJOIN is a join condition that involves only equality operator = .

- **Example:**
  - DEPT_MGR ← DEPARTMENT $\underset{\text{MGRSSN=SSN}}{\bowtie}$ EMPLOYEE
  - Retrieve a list of each female employee's dependents

    F ← $\sigma_{\text{SEX='F'}}$(EMPLOYEE)

    EN ← $\pi_{\text{FNAME, LNAME, SSN}}$ (F)

    E_DP ← EN $\underset{\text{SSN=ESSN}}{\rhd\!\bowtie}$ DEPENDENT

Issue with Equijoin Operation

DEPT_MGR

Dname	Dnumber	Mgr_ssn	...	Fname	Minit	Lname	Ssn	...
Research	5	333445555	...	Franklin	T	Wong	333445555	...
Administration	4	987654321	...	Jennifer	S	Wallace	987654321	...
Headquarters	1	888665555	...	James	E	Borg	888665555	...

- Superfluous column
- Result of EQUIJOIN always have one or more pairs of attributes that have identical values in every tuple.

NATURAL JOIN OPERATION

- NATURAL JOIN operation (denoted by *) is created to get rid of the superfluous attribute in an EQUIJOIN condition.

- The two join attributes, or each pair of corresponding join attributes must *have the same name* in both relations
  - If this is not the case, a renaming operation is applied first.

NATURAL JOIN OPERATION

- **Example:** To apply a natural join on the DNUMBER attributes of DEPARTMENT and DEPT_LOCATIONS, it is sufficient to write:
  - DEPT_LOCS ← DEPARTMENT * DEPT_LOCATIONS
- Only attribute with the same name is DNUMBER
- An implicit join condition is created based on this attribute:
  DEPARTMENT.DNUMBER=DEPT_LOCATIONS.DNUMBER

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date
Research	5	333445555	1988-05-22
Administration	4	987654321	1995-01-01
Headquarters	1	888665555	1981-06-19

DEPT_LOCATIONS

Dnumber	Dlocation
1	Houston
4	Stafford
5	Bellaire
5	Sugarland
5	Houston

Example: Natural Join

- Another example: Q ← R(A,B,C,D) * S(C,D,E)
  - The implicit join condition includes *each pair* of attributes with the same name, "AND" together:
    - R.C=S.C AND R.D=.S.D
  - Result keeps only one attribute of each such pair:
    - Q(A,B,C,D,E)

EXAMPLE OF NATURAL JOIN OPERATION

(a)

PROJ_DEPT

Pname	Pnumber	Plocation	Dnum	Dname	Mgr_ssn	Mgr_start_date
ProductX	1	Bellaire	5	Research	333445555	1988-05-22
ProductY	2	Sugarland	5	Research	333445555	1988-05-22
ProductZ	3	Houston	5	Research	333445555	1988-05-22
Computerization	10	Stafford	4	Administration	987654321	1995-01-01
Reorganization	20	Houston	1	Headquarters	888665555	1981-06-19
Newbenefits	30	Stafford	4	Administration	987654321	1995-01-01

(b)

DEPT_LOCS

Dname	Dnumber	Mgr_ssn	Mgr_start_date	Location
Headquarters	1	888665555	1981-06-19	Houston
Administration	4	987654321	1995-01-01	Stafford
Research	5	333445555	1988-05-22	Bellaire
Research	5	333445555	1988-05-22	Sugarland
Research	5	333445555	1988-05-22	Houston

Figure 6.7
Results of two NATURAL JOIN operations.
(a) PROJ_DEPT ← PROJECT * DEPT.
(b) DEPT_LOCS ← DEPARTMENT * DEPT_LOCATIONS.

EXAMPLE

Find SSN of employees who work on all the projects

PROJECT

Pname	Pnumber	Plocation	Dnum
ProductX	1	Bellaire	5
ProductY	2	Sugarland	5
ProductZ	3	Houston	5
Computerization	10	Stafford	4
Reorganization	20	Houston	1
Newbenefits	30	Stafford	4

SSN_PNOS

Essn	Pno
123456789	1
123456789	2
666884444	3
453453453	1
453453453	2
333445555	2
333445555	3
333445555	10
333445555	20
999887777	30
999887777	10
987987987	10
987987987	30
987654321	30
987654321	20
888665555	20

- All_Projects(Pno) ← π_{Pnumber} (Project)
- Ssn_Pnos ← $\pi_{\text{Essn,Pno}}$ (Works_on)
- SSNS(ssn)← Ssn_Pnos ??? All_Projects

DIVISION

DIVISION (BINARY OPERATION)

- The division operation is applied to two relations

  $R(Z) \div S(X)$, where $X \subset Z$.

- Let $Y = Z - X$
  - We have $Z = X \cup Y$ and Y is a set of attributes of R that are not the attributes of S.
  - The result of DIVISION is a relation T(Y)
  - For a tuple t to appear in the result T of the DIVISION, the values in t must appear in R in combination with *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

EXAMPLE OF DIVISION

Find SSN of employees who work on all the projects that *John Smith* works on

- Smith ← $\sigma_{\text{fname='John' and lname='Smith'}}$ (Employee)
- Smith_Pnos ← π_{Pno} (Works_on $\bowtie_{\text{essn=ssn}}$ Smith)
- Ssn_Pnos ← $\pi_{\text{Essn,Pno}}$ (Works_on)
- SSNS(ssn)← Ssn_Pnos ÷ Smith_Pnos

SSN_PNOS

Essn	Pno
123456789	1
123456789	2
666884444	3
453453453	1
453453453	2
333445555	2
333445555	3
333445555	10
333445555	20
999887777	30
999887777	10
987987987	10
987987987	30
987654321	30
987654321	20
888665555	20

SMITH_PNOS

Pno
1
2

SSNS

Ssn
123456789
453453453

Simulation of Division operator

- $R \div S$

- $Temp \leftarrow \pi_B((\pi_B(R) \times S) - R)$
- $T \leftarrow \pi_B(R) - Temp$

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

Recap of Relational Algebra Operations

Table 6.1
Operations of Relational Algebra

Operation	Purpose	Notation
SELECT	Selects all tuples that satisfy the selection condition from a relation R.	$\sigma_{\text{<selection condition>}}(R)$
PROJECT	Produces a new relation with only some of the attributes of R, and removes duplicate tuples.	$\pi_{\text{<attribute list>}}(R)$
THETA JOIN	Produces all combinations of tuples from R_1 and R_2 that satisfy the join condition.	$R_1 \bowtie_{\text{<join condition>}} R_2$
EQUIJOIN	Produces all the combinations of tuples from R_1 and R_2 that satisfy a join condition with only equality comparisons.	$R_1 \bowtie_{\text{<join condition>}} R_2$, OR $R_1 \bowtie_{(\text{<join attributes 1>}), (\text{<join attributes 2>})} R_2$
NATURAL JOIN	Same as EQUIJOIN except that the join attributes of R_2 are not included in the resulting relation; if the join attributes have the same names, they do not have to be specified at all.	$R_1 *_{\text{<join condition>}} R_2$, OR $R_1 *_{(\text{<join attributes 1>}), (\text{<join attributes 2>})} R_2$ OR $R_1 * R_2$
UNION	Produces a relation that includes all the tuples in R_1 or R_2 or both R_1 and R_2; R_1 and R_2 must be union compatible.	$R_1 \cup R_2$
INTERSECTION	Produces a relation that includes all the tuples in both R_1 and R_2; R_1 and R_2 must be union compatible.	$R_1 \cap R_2$
DIFFERENCE	Produces a relation that includes all the tuples in R_1 that are not in R_2; R_1 and R_2 must be union compatible.	$R_1 - R_2$
CARTESIAN PRODUCT	Produces a relation that has the attributes of R_1 and R_2 and includes as tuples all possible combinations of tuples from R_1 and R_2.	$R_1 \times R_2$
DIVISION	Produces a relation $R(X)$ that includes all tuples $t[X]$ in $R_1(Z)$ that appear in R_1 in combination with every tuple from $R_2(Y)$, where $Z = X \cup Y$.	$R_1(Z) \div R_2(Y)$

Aggregate Functions

- Now we specify mathematical **aggregate functions** on collections of values from the database.

- **Examples:**
  - Retrieve the average or total salary of all employees
  - Retrieve total number of employee tuples

- Functions applied to collections of numeric values include
  - SUM, AVERAGE, MAXIMUM, and MINIMUM.
  - COUNT function is used for counting tuples or values.

Aggregate Function Operation

- Use of the Aggregate Functional operation $\mathcal{F}$
  - $\mathcal{F}_{\text{MAX Salary}}$ (EMPLOYEE)
  - $\mathcal{F}_{\text{MIN Salary}}$ (EMPLOYEE)
  - $\mathcal{F}_{\text{SUM Salary, AVERAGE Salary}}$ (EMPLOYEE)
  - $\mathcal{F}_{\text{COUNT SSN}}$ (EMPLOYEE)
    - Count just counts the number of rows, without removing duplicates
    - COUNT (*) returns the number of rows in the result of the query.
    - COUNT can be used to count values in a column rather than tuples, example COUNT SSN
    - NULL values are **discarded** when aggregate functions are applied to a particular column (attribute).

Using Grouping with Aggregation

- Grouping can be combined with Aggregate Functions

- **Example:**
  - For each department, retrieve the DNO, COUNT SSN, and AVERAGE SALARY
  - DNO $\mathcal{F}$COUNT SSN, AVERAGE Salary (EMPLOYEE)

EXAMPLE: AGGREGATE FUNCTIONS AND GROUPING

EMPLOYEE

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
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
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
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

The aggregate function operation.

(a) $\rho_{R(\text{Dno, No\_of\_employees, Average\_sal})}(_{\text{Dno}}\Im_{\text{COUNT Ssn, AVERAGE Salary}}(\text{EMPLOYEE}))$.

(b) $_{\text{Dno}}\Im_{\text{COUNT Ssn, AVERAGE Salary}}(\text{EMPLOYEE})$.

(c) $\Im_{\text{COUNT Ssn, AVERAGE Salary}}(\text{EMPLOYEE})$.

R

(a)

Dno	No_of_employees	Average_sal
5	4	33250
4	3	31000
1	1	55000

(b)

Dno	Count_ssn	Average_salary
5	4	33250
4	3	31000
1	1	55000

(c)

Count_ssn	Average_salary
8	35125

EXAMPLES OF QUERIES IN RA

- **Q1: Retrieve the name and address of all employees who work for the 'Research' department.**

  RESEARCH_DEPT ← $\sigma_{\text{DNAME='Research'}}$ (DEPARTMENT)

  RESEARCH_EMPS ← (RESEARCH_DEPT $\bowtie_{\text{DNUMBER= DNO}}$ EMPLOYEE)

  RESULT ← $\pi_{\text{FNAME, LNAME, ADDRESS}}$ (RESEARCH_EMPS)

EMPLOYEE

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
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
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
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date
Research	5	333445555	1988-05-22
Administration	4	987654321	1995-01-01
Headquarters	1	888665555	1981-06-19

DEPT_LOCATIONS

Dnumber	Dlocation
1	Houston
4	Stafford
5	Bellaire
5	Sugarland
5	Houston

Examples of Queries in RA

- **Q2: Retrieve the names of employees who have no dependents.**

  ALL_EMPS ← π SSN(EMPLOYEE)

  EMPS_WITH_DEPS(SSN) ← π ESSN(DEPENDENT)

  EMPS_WITHOUT_DEPS ← (ALL_EMPS - EMPS_WITH_DEPS)

  RESULT ← π LNAME, FNAME (EMPS_WITHOUT_DEPS * EMPLOYEE)

EMPLOYEE

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
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-0					
Ahmad	V	Jabbar	987987987	1969-0					
James	E	Borg	888665555	1937-1					

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship
333445555	Alice	F	1986-04-05	Daughter
333445555	Theodore	M	1983-10-25	Son
333445555	Joy	F	1958-05-03	Spouse
987654321	Abner	M	1942-02-28	Spouse
123456789	Michael	M	1988-01-04	Son
123456789	Alice	F	1988-12-30	Daughter
123456789	Elizabeth	F	1967-05-05	Spouse

Examples of Queries in RA

- **Q3: Retrieve the names of all employees with two or more dependents.**

  $T1(Ssn, No\_of\_dependents) \leftarrow Essn\ \mathcal{F}_{COUNT\ Dependent\_name}(DEPENDENT)$

  $T2 \leftarrow \sigma_{No\_of\_dependents > 1}(T1)$

  $RESULT \leftarrow \pi_{LNAME, FNAME}(T2 * EMPLOYEE)$

EMPLOYEE

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
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
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
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555						

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship
333445555	Alice	F	1986-04-05	Daughter
333445555	Theodore	M	1983-10-25	Son
333445555	Joy	F	1958-05-03	Spouse
987654321	Abner	M	1942-02-28	Spouse
123456789	Michael	M	1988-01-04	Son
123456789	Alice	F	1988-12-30	Daughter
123456789	Elizabeth	F	1967-05-05	Spouse

Examples of Queries in RA

Q4: Find the names of employees who work on *all* the projects controlled by department number 5.

$T1(Pno) \leftarrow \pi_{Pnumber} (\sigma_{Dnum=5} (Project))$

$T2 \leftarrow \pi_{Essn, Pno} (Work\_On)$

$T3 \leftarrow (T2 \div T1)$

$RESULT \leftarrow \pi_{LNAME, FNAME} (T3 * EMPLOYEE)$

PROJECT

Pname	Pnumber	Plocation	Dnum
ProductX	1	Bellaire	5
ProductY	2	Sugarland	5
ProductZ	3	Houston	5
Computerization	10	Stafford	4
Reorganization	20	Houston	1
Newbenefits	30	Stafford	4

Essn	Pno
123456789	1
123456789	2
666884444	3
453453453	1
453453453	2
333445555	2
333445555	3
333445555	10
333445555	20
999887777	30
999887777	10
987987987	10
987987987	30
987654321	30
987654321	20
888665555	20

Outer Join Operation

- In INNER JOIN, tuples without a *matching* are eliminated from the join result
  - Tuples with null are also eliminated
  - This amounts to loss of information.

- OUTER joins operations are used when we want to keep
  - all the tuples in R in the join result , or
  - all tuples in S in the join result, or
  - all tuples in both relations R and S in the join result

LEFT OUTER JOIN

- List the employees name and the department name that they manage. If they don't manage one, then indicate this with a null value.
- Temp ← (Employee $\overset{\text{Ssn=Mgr\_Ssn}}{\bowtie}$ Department)
- Result ← $\pi_{\text{Fname, Minit, Lname, Dname}}$(Temp)

RESULT

Fname	Minit	Lname	Dname
John	B	Smith	NULL
Franklin	T	Wong	Research
Alicia	J	Zelaya	NULL
Jennifer	S	Wallace	Administration
Ramesh	K	Narayan	NULL
Joyce	A	English	NULL
Ahmad	V	Jabbar	NULL
James	E	Borg	Headquarters

Outer Join Operation

- **Left outer join**: keeps every tuple in R, denoted as R ⟕ S
  - if no matching tuple is found in S, then the attributes of S in the join result are filled with null values.

- **Right outer join:** keeps every tuple in S in the result of R ⟖ S.

- **Full outer join:** keeps all tuples in both the left and the right relations. It is denoted by ⟗

Full Outer Join vs Cartesian Product

EMPLOYEE

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
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
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
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date
Research	5	333445555	1988-05-22
Administration	4	987654321	1995-01-01
Headquarters	1	888665555	1981-06-19

??

Employee $_{\text{Ssn=Mgr\_Ssn}}$ Department

RESULT

Fname	Minit	Lname	Dname
John	B	Smith	NULL
Franklin	T	Wong	Research
Alicia	J	Zelaya	NULL
Jennifer	S	Wallace	Administration
Ramesh	K	Narayan	NULL
Joyce	A	English	NULL
Ahmad	V	Jabbar	NULL
James	E	Borg	Headquarters

OUTER UNION OPERATION

- It takes the union of tuples in two relations R(X, Y) and S(X, Z) that are **partially compatible**,
  - Only some of their attributes, say X, are type compatible.
  - The attributes that are type compatible are represented only once in the result
  - The attributes that are not type compatible from either relation are also kept in the result relation T(X, Y, Z).

Outer Union Example

- An outer union can be applied to two relations **STUDENT**(Name, SSN, Department, Advisor) and **INSTRUCTOR**(Name, SSN, Department, Rank).

  - Result relation:
    STUDENT_OR_INSTRUCTOR (Name, SSN, Department, Advisor, Rank)
  - Tuples are matched based on the values of the shared attributes— ***Name, SSN, Department.***
  - If a student is also an instructor, both Advisor and Rank will have a value; otherwise, one of these two attributes will be null.
  - It is same as a FULL OUTER JOIN on the common attributes.

Recursive Closure Operation

- This can't be specified in general using Relational Algebra
- **Example:** Retrieve all SUPERVISEES of an EMPLOYEE **e** at all levels — that is,
  - all employees **e`** directly supervised by **e**;
  - all employees **e``** directly supervised by each employee **e`**;
  - all employees **e```**directly supervised by each employee **e``**;
  - and so on.
- We can retrieve employees at each level and then take their union, however, we cannot specify a query such as
  - "retrieve the supervisees of 'James Borg' at all levels" without utilizing a looping mechanism.
- The SQL3 standard includes syntax for recursive closure.

Recursive Closure Operation

(Borg's SSN is 888665555)

	(SSN)	(SUPERSSN)
SUPERVISION	SSN1	SSN2
	123456789	333445555
	333445555	888665555
	999887777	987654321
	987654321	888665555
	666884444	333445555
	453453453	333445555
	987987987	987654321

RESULT 1	SSN
	333445555
	987654321

(Supervised by Borg)

RESULT 2	SSN
	123456789
	999887777
	666884444
	453453453
	987987987

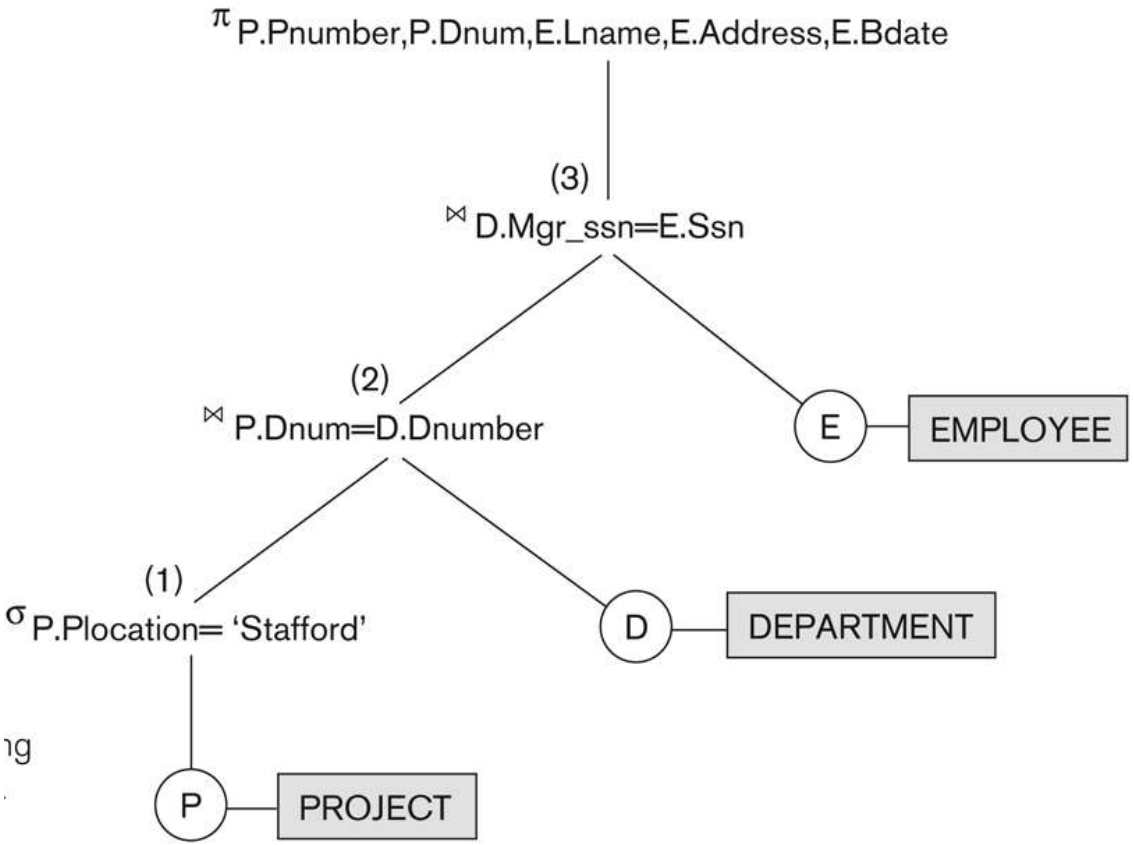
(Supervised by Borg's subordinates)

RESULT	SSN
	123456789
	999887777
	666884444
	453453453
	987987987
	333445555
	987654321

(RESULT1 ∪ RESULT2)

Example of Query Tree

Query: *For every project located in 'Stafford', list the project number, the controlling department number, and the department manager's last name, address, and birth date.*



Query Tree

- An internal data structure to represent a query
- Standard technique to estimate the work done in executing the query, and the optimization of execution

- Nodes stand for operations like selection, projection, join, renaming, division, ….
- Leaf nodes represent base relations

- A tree gives a good visual feel of the complexity of the query and the operations involved
- Algebraic Query Optimization consists of rewriting the query or modifying the query tree into an equivalent tree.

Relational Algebra Operators

- Relational Algebra consists of several groups of operations
  - **Unary Relational Operations**
    - SELECT (symbol: **σ** (sigma))
    - PROJECT (symbol: **π** (pi))
    - RENAME (symbol: **ρ** (rho))
  - **Relational Algebra Operations From Set Theory**
    - UNION (∪), INTERSECTION (∩), DIFFERENCE (–)
    - CARTESIAN PRODUCT (**x**)
  - **Binary Relational Operations**
    - JOIN (several variations of JOIN exist)
    - DIVISION
  - **Additional Relational Operations**
    - OUTER JOINS, OUTER UNION
    - AGGREGATE FUNCTIONS (These compute summary of information: for example, SUM, COUNT, AVG, MIN, MAX)

Chapter Summary

- Relational Algebra
  - Unary Relational Operations
  - Relational Algebra Operations From Set Theory
  - Binary Relational Operations
  - Additional Relational Operations
  - Examples of Queries in Relational Algebra