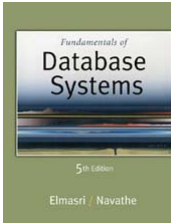


## Chapter 6

### İlişkisel Cebir ve İlişkisel Hesaplama (The Relational Algebra and Relational Calculus)

*Türkçe özet: Erdoğan Dođdu*



Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

## İçerik

- İlişkisel Cebir (Relational Algebra)
  - Bir ilişkili işlemler
  - Küme teorisinden ilişkisel işlemler
  - İkili ilişkisel işlemler
  - Diğer ilişkisel işlemler
  - Örnekler
- İlişkisel Hesaplama (Relational Calculus)
  - Tuple İlişkisel Hesaplama (TRC)
  - Domain İlişkisel Hesaplama (DRC)
- Örnek veritabanı uygulaması (COMPANY)
- QBE (Query By Example) Dili (appendix D)

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 3

## İlişkisel Cebir (Relational Algebra)

- İlişkisel model için bir dizi işlem.
- Kullanıcılara sorgulama (**query**) imkanı verir.
- İşlem sonucu bir veya daha fazla ilişkiden yeni ilişki (relation) oluşturulur.
  - İlişkisel Cebir'de tüm nesneler "ilişki"lerdir.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 4

## İlişkisel Cebir (Relational Algebra)

- **Algebra operations** = Cebir işlemleri
  - Sonuç yeni bir ilişki ve başka bir işlemde kullanılabilir.
- **Relational algebra expression** = İlişkisel cebir ifadesi
  - Bir dizi ilişkili ve ilişkisel cebir işlemlerinden oluşur, sonucu yine bir ilişkidir.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 5

## Algebra (Cebir)

- **Muhammad ibn Musa al-Khwarizmi (800-847 CE)**
  - “Al-jabr” kitabının yazarı (aritmetik değişkenler hakkında)
  - Daha sonra kitap Latinceye çevrilir.
  - Kitabın başlığı “Al-jabr” (Cebir) “Algebra”ya ismini vermiştir.
- **Değişkenlere neden “x” denir?**
  - Al-Khwarizmi değişkenlere “şey” (İngilizce “shay”) diyordu
  - “Shay”: Arapça/Türkçe “şey” (İngilizce’de “thing”).
  - İspanyollar “şey/shay” kelimesini “xay” olarak yazdılar (İspanyolcada “ş/sh” sesi “x” ile yazılır).
  - Zamanla bu kelime “x” olarak kısaltılmıştır.
- **Algoritma (algorithm) kelimesi nereden geliyor?**
  - Algoritma/Algorithm, “al-Khwarizmi”den gelir.
  - Referans: PBS (<http://www.pbs.org/empires/islam/innoalgebra.html>)

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 6

## İlişkisel Cebir (Relational Algebra)

- İşlemler (operations)
  - Bir İlişkili İşlemler (unary operations)
    - SELECT ( $\sigma$  (sigma))
    - PROJECT ( $\pi$  (pi))
    - RENAME ( $\rho$  (rho))
  - Küme Teorisi İşlemleri
    - UNION ( $\cup$ ), INTERSECTION ( $\cap$ ), DIFFERENCE (or MINUS,  $-$ )
    - CARTESIAN PRODUCT ( $\times$ )
  - İkili İşlemler (binary)
    - JOIN (birkaç türü var)
    - DIVISION
  - Diğer İşlemler
    - OUTER JOINS, OUTER UNION
    - AGGREGATE FUNCTIONS (özet bilgi hesaplarlar: örneğin, SUM, COUNT, AVG, MIN, MAX)

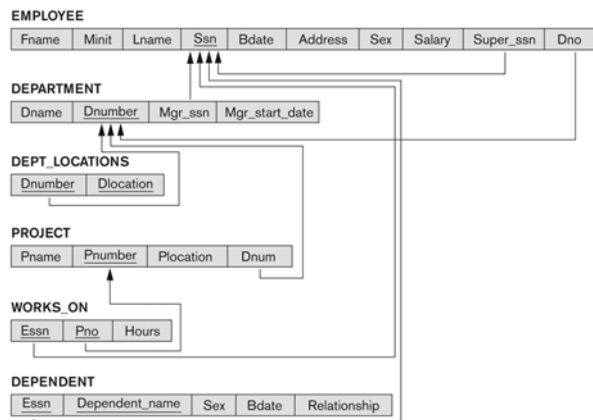
Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 7

## COMPANY Veritabanı

- Örnekler bu veritabanı üzerinden verilecektir.

**Figure 5.7**  
Referential integrity constraints displayed on the COMPANY relational database schema.



Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 8

## SELECT

- İşaret:  $\sigma$  (sigma)
- Filtreleme işlemi yapar, tuple'ların bir alt kümesini seçer.
- Seçme koşulu (selection condition): Filtreleme ifadesi
- Örnek:
  - Bölüm numarası 4 olan EMPLOYEE'ler:

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

- Maaşı \$30,000'dan fazla olan EMPLOYEE'ler:

$\sigma_{SALARY > 30,000} (EMPLOYEE)$

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 9

## SELECT

- SELECT ifadesi:
  - $\sigma_{\langle \text{selection condition} \rangle} (R)$
  - $\sigma$  : select işlemi
  - Selection condition : öznitelikler kullanılarak oluşturulan mantıksal seçme koşulu ifadesi.
  - R : İlişki
  - Seçme koşulunu doğru (**true**) yapan tuple'lar seçilir.
  - Seçme koşulunu yanlış (**false**) yapan tuple'lar seçilmezler.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 10

# SELECT

## ■ SELECT işlemi özellikleri

- SELECT işlemi sonucu bir yeni ilişkidir. Bu ilişkinin şeması seçilen ilişki (R) ile aynıdır (aynı öznitelikler).
- SELECT işlemi yerdeğiştirme özelliğine sahiptir:
  - $\sigma_{\langle \text{condition1} \rangle}(\sigma_{\langle \text{condition2} \rangle}(R)) = \sigma_{\langle \text{condition2} \rangle}(\sigma_{\langle \text{condition1} \rangle}(R))$
- Yerdeğiştirme özelliğinden dolayı bir dizi SELECT işlemi istenilen sırada uygulanabilir (aynı sonuç).
  - $\sigma_{\langle \text{cond1} \rangle}(\sigma_{\langle \text{cond2} \rangle}(\sigma_{\langle \text{cond3} \rangle}(R))) = \sigma_{\langle \text{cond2} \rangle}(\sigma_{\langle \text{cond3} \rangle}(\sigma_{\langle \text{cond1} \rangle}(R)))$
- Bir dizi SELECT işlemi, seçme koşulları VE ile birleştirilerek tek bir SELECT işlemi olarak yazılabilir:
  - $\sigma_{\langle \text{cond1} \rangle}(\sigma_{\langle \text{cond2} \rangle}(\sigma_{\langle \text{cond3} \rangle}(R))) = \sigma_{\langle \text{cond1} \rangle \text{ AND } \langle \text{cond2} \rangle \text{ AND } \langle \text{cond3} \rangle}(R))$
- SELECT sonucu ilişki R'daki tuple sayısı kadar veya daha az tuple içerir.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 11

## Örnek veritabanı durumu

Figure 5.6 One possible database state for the COMPANY relational database schema.

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

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	20	10.0
333445555	10	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

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 12

## PROJECT

- İşaret:  $\pi$  (pi)
- İstenilen sütunları (öznitelikler) seçer diğerlerini siler.
  - PROJECT dikey seçme yapar (verital partitioning)
    - Herbir tuple için belirtilen öznitelikler tutulur, diğerleri atılır.
- Örnek: Her bir EMPLOYEE'nin adı, soyadı, ve maaşını listele:

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

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 13

## PROJECT

- PROJECT ifadesi:
  - $\pi_{\text{<attribute list>}}(R)$
  - $\pi$  (pi): PROJECT işlemi
  - <attribute list>: R'dan seçilecek öznitelik listesi
  - Sonuçta elde edilen ilişkide aynı olan (duplicate) tuple'lar silinir.
    - Çünkü sonuç da bir ilişkidir, yani bir tuple kümesidir, ve kümeler (set) aynı elemanları, yani aynı öznitelik değerlerine sahip birden fazla tuple'ı, içeremez.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 14

## PROJECT (devam)

- PROJECT işlemi özellikleri:
  - Project işlemi  $\pi_{\langle \text{list} \rangle}(R)$  ile elde edilen sonuç ilişkisindeki tuple sayısı, R ilişkisindeki tuple sayısı ile aynı veya daha azdır.
    - Eğer list'de bir anahtar varsa, R ve sonuç ilişkisindeki tuple sayısı eşittir.
  - PROJECT yerdeğiştirebilir DEĞİLDİR
    - $\pi_{\langle \text{list1} \rangle}(\pi_{\langle \text{list2} \rangle}(R)) = \pi_{\langle \text{list1} \rangle}(R)$  (eğer  $\langle \text{list2} \rangle$ ,  $\langle \text{list1} \rangle$ 'deki öznitelikleri içeriyorsa)

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 15

## SELECT ve PROJECT örnekleri

**Figure 6.1**

Results of SELECT and PROJECT operations. (a)  $\sigma_{(Dno=4 \text{ AND } Salary > 25000) \text{ OR } (Dno=5 \text{ AND } Salary > 30000)}(EMPLOYEE)$ . (b)  $\pi_{Lname, Fname, Salary}(EMPLOYEE)$ . (c)  $\pi_{Sex, Salary}(EMPLOYEE)$ .

(a)

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

(b)

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

(c)

Sex	Salary
M	30000
M	40000
F	25000
F	43000
M	38000
M	25000
M	55000

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 16



## İlişkisel Cebir İfadeleri

- Birden fazla işlem içeren uzun bir ifade olabilir veya,
  - $\pi_{\text{FNAME, LNAME, SALARY}}(\sigma_{\text{DNO}=5}(\text{EMPLOYEE}))$
- Ara işlemler bir ara ilişkiye atanır ve sonraki işlemlerde kullanılır (parçalı ifadeler)
  - **DEP5\_EMPS**  $\leftarrow \sigma_{\text{DNO}=5}(\text{EMPLOYEE})$
  - **RESULT**  $\leftarrow \pi_{\text{FNAME, LNAME, SALARY}}(\text{DEP5_EMPS})$

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 17

## RENAME

- İşaret:  $\rho$  (rho)
- İlişki ve/veya ilişki özniteliklerinin adlarını değiştirmekte kullanılır.
  - Aynı ilişkiye farklı işlemlerde kullanmak üzere,
  - Veya JOIN gibi ikili işlemlerde aynı ilişkiyi iki farklı anlamda kullanmak için.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 18

## RENAME

### ■ RENAME ifadesi:

- $\rho_{S(B1, B2, \dots, Bn)}(R)$ 
  - İlişki adı R'den S'ye değişir, ve
  - Öznitelik adları B1, B1, .....Bn'e değişir.
- $\rho_S(R)$ 
  - Yalnızca ilişki adı R'den S'ye değişir
- $\rho_{(B1, B2, \dots, Bn)}(R)$ 
  - Yalnızca öznitelik adları B1, B1, .....Bn'e değişir.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 19

## RENAME

### ■ Daha kısa kullanımı:

- $RESULT \leftarrow \pi_{FNAME, LNAME, SALARY}(DEP5\_EMPS)$ 
  - RESULT 'daki öznitelik adları  
DEP5\_EMPS'dakilerle (ve aynı zamanda  
EMPLOYEE'dekilerle) aynı.
- $RESULT(F, M, L, S, B, A, SX, SAL, SU, DNO) \leftarrow$   
 $\rho_{RESULT(F,M,L,S,B,A,SX,SAL,SU,DNO)}(DEP5\_EMPS)$ 
  - DEP5\_EMPS'deki 10 özneliğin adları sırasıyla F,  
M, L, S, B, A, SX, SAL, SU, DNO olarak değişti.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 20

## RENAME örneği

(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

**Figure 6.2**

Results of a sequence of operations.

(a)  $\pi_{Fname, Lname, Salary}(\sigma_{Dno=5}(EMPLOYEE))$ .

(b) Using intermediate relations and renaming of attributes.

Slide 6- 21

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

## UNION (Birleşim)

- İşaret:  $\cup$
- $R \cup S$ , R ve/veya S'deki tüm tuple'ları verir.
- Aynı tuple'lardan birisi alınır.
- R ve S ilişkileri tür uyumlu "type compatible" (veya UNION uyumlu) olmalıdır,
  - R ve S aynı sayıda öznitelik içermeli,
  - Karşılıklı öznitelikler tür uyumlu (type compatible) olmalı; aynı veya uyumlu domain'lere sahip olmalılar.
    - $\pi_{Phone}(DEPARTMENT) \cup \pi_{Office\_phone}(EMPLOYEE)$

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 22

## UNION örneği

- 5 nolu bölümde çalışan (RESULT1) veya 5 nolu bölümde çalışan birini yöneten (supervise) EMPLOYEE'lerin (RESULT2) SSN numaraları:
  - $DEP5\_EMPS \leftarrow \sigma_{DNO=5}(EMPLOYEE)$
  - $RESULT1 \leftarrow \pi_{SSN}(DEP5\_EMPS)$
  - $RESULT2(SSN) \leftarrow \pi_{SUPERSSN}(DEP5\_EMPS)$
  - $RESULT \leftarrow RESULT1 \cup RESULT2$
  - UNION işlemi RESULT1, RESULT2, veya her ikisinde yer alan tuple'ları verir.
  - Uzun ifade olarak:
    - $\pi_{SSN}(\sigma_{DNO=5}(EMPLOYEE)) \cup \pi_{SUPERSSN}(EMPLOYEE)$
    - Hangisi daha hızlı çalışır?

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 23

## UNION örneği

**Figure 6.3**

Result of the  
UNION operation  
 $RESULT \leftarrow RESULT1$   
 $\cup RESULT2$ .

**RESULT1**

Ssn
123456789
333445555
666884444
453453453

**RESULT2**

Ssn
333445555
888665555

**RESULT**

Ssn
123456789
333445555
666884444
453453453
888665555

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 24

## Küme Teorisinden Alınan İlişkisel İşlemler

- UNION  $\cup$
- INTERSECTION  $\cap$ ,
- SET DIFFERENCE  $-$
- Bu işlemler ikili işlemler, iki ilişki içerir, bu ilişkiler “tür uyumlu” olmalılar:
  - $R1(A1, A2, \dots, An)$  ve  $R2(B1, B2, \dots, Bn)$  aşağıdaki koşullar doğruysa tür uyumludur:
    - Aynı sayıda öznitelik içerir (n)
    - Karşılıklı öznitelikler tür uyumlu olmalı
      - $dom(Ai)=dom(Bi), i=1, 2, \dots, n$
  - $(R1 \text{ işlem } R2)$ 'nin sonucu  $R1$ 'deki aynı öznitelik adlarını alır.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 25

## INTERSECTION

- İşaret:  $\cap$
- $R \cap S$  sonucu hem R ve hem de S'de yer alan tuple'ları içerir.
  - R ve S “type compatible” olmalı
  - Sonuç ilişkideki öznitelikler R'deki öznitelik adlarını taşır.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 26

## SET DIFFERENCE (MINUS)

- İşaret: –
- $R - S$  sonucu R’da yer alan fakat S’de yer almayan tuple’lar.
  - R ve S “type compatible” olmalı
  - Sonuç ilişkideki öznitelikler R’deki öznitelik adlarını taşır.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 27

## UNION, INTERSECT, and DIFFERENCE örneği

(a) STUDENT		INSTRUCTOR		(b)	
Fn	Ln	Fname	Lname	Fn	Ln
Susan	Yao	John	Smith	Susan	Yao
Ramesh	Shah	Ricardo	Browne	Ramesh	Shah
Johnny	Kohler	Susan	Yao	Johnny	Kohler
Barbara	Jones	Francis	Johnson	Barbara	Jones
Amy	Ford	Ramesh	Shah	Amy	Ford
Jimmy	Wang			Jimmy	Wang
Ernest	Gilbert			Ernest	Gilbert

(c)		(d)		(e)	
Fn	Ln	Fn	Ln	Fname	Lname
Susan	Yao	Johnny	Kohler	John	Smith
Ramesh	Shah	Barbara	Jones	Ricardo	Browne

(d)		(e)	
Fn	Ln	Fname	Lname
Johnny	Kohler	John	Smith
Barbara	Jones	Ricardo	Browne
Amy	Ford	Francis	Johnson
Jimmy	Wang		
Ernest	Gilbert		

**Figure 6.4**

The set operations UNION, INTERSECTION, and MINUS. (a) Two union-compatible relations. (b)  $STUDENT \cup INSTRUCTOR$ . (c)  $STUDENT \cap INSTRUCTOR$ . (d)  $STUDENT - INSTRUCTOR$ . (e)  $INSTRUCTOR - STUDENT$ .

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 28

## UNION, INTERSECT, and DIFFERENCE özellikleri

- Union ve intersection yer değiştirebilir (*commutative*) işlemlerdir:
  - $R \cup S = S \cup R$ , and  $R \cap S = S \cap R$
- Union ve intersection n-li işlemler olarak kullanılabilir; her iki işlemde (*associative*) işlemler:
  - $R \cup (S \cap T) = (R \cup S) \cap T$
  - $(R \cap S) \cap T = R \cap (S \cap T)$
- Set difference (minus) işlemi yer değiştirir (*commutative*) değildir.
  - $R - S \neq S - R$

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 29

## CARTESIAN (CROSS) PRODUCT

- İşaret:  $\times$
- $R(A_1, A_2, \dots, A_n) \times S(B_1, B_2, \dots, B_m)$  sonucu R ve S'deki tüm tuple'ların karşılıklı her türlü kombinasyonunu içerir.
  - Sonuç ilişkisi (Q)'nin derecesi:  $n + m$  öznitelik:
    - $Q(A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_m)$  (öznitelikler aynı sırada)
  - R'deki tuple sayısı:  $|R| = n_R$
  - S'deki tuple sayısı:  $|S| = n_S$
  - R x S'deki tuple sayısı:  $(n_R * n_S)$  tuple
  - R ve S tür uyumlu olmak zorunda değil.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 30

## CARTESIAN PRODUCT

- CROSS PRODUCT tek başına anlamlı değil, diğer işlemlerle anlamlı olur.
- Örnek (anlamsız):
  - $FEMALE\_EMPS \leftarrow \sigma_{SEX='F'}(EMPLOYEE)$
  - $EMP\_NAMES \leftarrow \pi_{FNAME, LNAME, SSN}(FEMALE\_EMPS)$
  - $EMP\_DEPENDENTS \leftarrow EMP\_NAMES \times DEPENDENT$
- EMP\_DEPENDENTS, EMPNAMES ve DEPENDENT'deki tuple'ların her türlü kombinasyonunu verir.
  - Akraba olsunlar veya olmasınlar..

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 31

## CARTESIAN PRODUCT (cont.)

- Örnek (anlamlı):
  - $FEMALE\_EMPS \leftarrow \sigma_{SEX='F'}(EMPLOYEE)$
  - $EMP\_NAMES \leftarrow \pi_{FNAME, LNAME, SSN}(FEMALE\_EMPS)$
  - $EMP\_DEPENDENTS \leftarrow EMP\_NAMES \times DEPENDENT$
  - $ACTUAL\_DEPS \leftarrow \sigma_{SSN=ESSN}(EMP\_DEPENDENTS)$
  - $RESULT \leftarrow \pi_{FNAME, LNAME, DEPENDENT\_NAME}(ACTUAL\_DEPS)$
- RESULT: Kadın çalışanlar ve onların bağımlıları (dependents)
  - SELECT işlemi ile kartezyen çarpım sonucu filtreleniyor ve yalnızca ilişkili ikililer tutuluyor..

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 32



# CARTESIAN PRODUCT

## Örneği

**Figure 6.5**  
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	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, 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	...

ACTUAL DEPENDENTS							
Fname	Lname	Ssn	Essn	Dependent_name	Sex	Bdate	...
Jennifer	Wallace	987654321	987654321	Abner	M	1942-02-28	...

RESULT		
Fname	Lname	Dependent_name
Jennifer	Wallace	Abner

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 33

# JOIN

- İşaret: ⋈
- CARTESIAN PRODUCT ve arkasından SELECT işlemi sıklıkla kullanılır (ilişkili tuple'ları bulmak için...)
- JOIN bu iki işlemi birleştiren özel bir işlem
- İlişkisel veritabanlarındaki en önemli işlem...
- $R(A_1, A_2, \dots, A_n)$  ve  $S(B_1, B_2, \dots, B_m)$  için:

$$R \bowtie_{\langle \text{join condition} \rangle} S$$

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 34

## JOIN örneği

- Örnek: Her bölümün yöneticisinin adı
  - DEPARTMENT ve EMPLOYEE tuple'larının kombinasyonlarını al, yalnızca DEPARTMENT manager SSN'i, EMPLOYEE SSN'i ile aynı olan ikilileri tut.
$$\text{DEPT\_MGR} \leftarrow \text{DEPARTMENT} \bowtie_{\text{MGRSSN=SSN}} \text{EMPLOYEE}$$
  - Koşul aşağıdaki gibi de yazılabilir
    - DEPARTMENT.MGRSSN= EMPLOYEE.SSN

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 35

## JOIN örneği

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	...

Figure 6.6

Result of the JOIN operation

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

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 36

## JOIN özellikleri

- $R(A_1, A_2, \dots, A_n) \bowtie_{R.A_i=S.B_j} S(B_1, B_2, \dots, B_m)$ 
  - Sonuç (Q)  $n + m$  öznitelik içerir:
    - $Q(A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_m)$  (aynı sırada).
  - Q tuple'ları bir tuple R'den (r) ve bir tuple S'den (s) içerir ve ayrıca  $r[A_i]=s[B_j]$  olmalıdır (join koşulu doğru olmalı).
  - $n_R$  : R'deki tuple sayısı
  - $n_Q \leq (n_R * n_S)$

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 37

## JOIN özellikleri

- JOIN ifadesi (genel): Teta-join
  - $R \bowtie_{theta} S$
- Theta: join koşulu; herhangi bir mantıksal ifade
- Örnek:
  - $R.A_i < S.B_j$  AND  $(R.A_k = S.B_l$  OR  $R.A_p < S.B_q)$
  - Genellikle birden fazla eşitlik AND'lenmiş olur:
    - $R.A_i = S.B_j$  AND  $R.A_k = S.B_l$  AND  $R.A_p = S.B_q$

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 38

## EQUIJOIN

- Yalnızca eşitlik karşılaştırmaları (=) içeren join.
  - Karşılaştırılan öznitelik adları aynı olmaz zorunda değil.
  - Örnek EQUIJOIN:
    - $R.A_i = S.B_j$  AND  $R.A_k = S.B_l$  AND  $R.A_p = S.B_q$

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 39

## NATURAL JOIN

- İşaret: \*
- EQUIJOIN'deki aynı isimli öznitelik karşılaştırmalarını tekrar yazmamak için.
- İki ilişkideki aynı isimli özniteliklerin eşitliğini karşılaştırır.
  - Eğer öznitelik adları farklı ise önce RENAME ( $\rho$ ) işlemi ile adlar aynı yapılır.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 40

## NATURAL JOIN

- Örnek: DEPARTMENT ve DEPT\_LOCATIONS (DNUMBER üzerinden natural join):
  - $DEPT\_LOCS \leftarrow DEPARTMENT * DEPT\_LOCATIONS$
- Her iki ilişkide DNUMBER aynı adlı tek öznitelik
- DNUMBER üzerinden görünmez bir join koşulu oluşturulur:
 
$$DEPARTMENT.DNUMBER = DEPT\_LOCATIONS.DNUMBER$$
- Örnek:  $Q \leftarrow R(A,B,C,D) * S(C,D,E)$ 
  - Görünmez join koşulu:
    - $R.C = S.C \text{ AND } R.D = S.D$
  - Sonuç ilişkisi (aynı olan özniteliklerden birisi alınır)
    - $Q(A,B,C,D,E)$

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 41

## NATURAL JOIN

örneği

(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)  $PROJ\_DEPT \leftarrow PROJECT * DEPT$ .

(b)  $DEPT\_LOCS \leftarrow DEPARTMENT * DEPT\_LOCATIONS$ .

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 42

## İlişkisel işlemlerin kapalı kümesi (complete set)

- SELECT  $\sigma$ , PROJECT  $\pi$ , UNION  $\cup$ , DIFFERENCE  $-$ , RENAME  $\rho$ , CARTESIAN PRODUCT  $\times$
- Diğer işlemler bu beş işlemle ifade edilebilir.
- Örnek:
  - $R \cap S = (R \cup S) - ((R - S) \cup (S - R))$
  - $R \bowtie_{\langle \text{join condition} \rangle} S = \sigma_{\langle \text{join condition} \rangle} (R \times S)$

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 43

## DIVISION (bölme)

- İşaret:  $\div$ 
  - $R(Z) \div S(X)$ , X, Z'nin alt kümesi.
  - $Y = Z - X$  (ve  $Z = X \cup Y$ )
  - Y: R'de olan S'de olmayan öznitelikler
  - DIVISION sonucu:
    - $T(Y)$ : Z'den X öznitelikleri atılır
    - T, R ilişkisinden yalnızca aşağıdaki t tuple'larını tutar:
      - $t_R[Y] = t$ , ve
      - $t_R[X] = t_s$  (S'deki herbir  $t_s$  için)
  - S'deki bütün tuple değerlerini içeren R tuple'larının **ortak (aynı)** Y özniteliklerini verir.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 44

## DIVISION örneği

(a)		(b)	
SSN_PNOS		SMITH_PNOS	
Essn	Pno	Pno	
123456789	1	1	
123456789	2	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		

SSNS	
Ssn	
123456789	
453453453	

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

T	
B	
b1	
b4	

**Figure 6.8**

The DIVISION operation. (a) Dividing SSN\_PNOS by SMITH\_PNOS. (b)  $T \leftarrow R \div S$ .

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 45

## İlişkisel Cebir (özet)

**Table 6.1**  
Operations of Relational Algebra

Operation	Purpose	Notation
SELECT	Selects all tuples that satisfy the selection condition from a relation $R$ .	$\sigma_{\langle \text{selection condition} \rangle}(R)$
PROJECT	Produces a new relation with only some of the attributes of $R$ , and removes duplicate tuples.	$\pi_{\langle \text{attribute list} \rangle}(R)$
THETA JOIN	Produces all combinations of tuples from $R_1$ and $R_2$ that satisfy the join condition.	$R_1 \bowtie_{\langle \text{join condition} \rangle} R_2$
EQUIJOIN	Produces all the combinations of tuples from $R_1$ and $R_2$ that satisfy a join condition with only equality comparisons.	$R_1 \bowtie_{\langle \text{join condition} \rangle} R_2$ OR $R_1 \bowtie_{(\langle \text{join attributes } 1 \rangle), (\langle \text{join attributes } 2 \rangle)} R_2$
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.	$R_1 \natural_{\langle \text{join condition} \rangle} R_2$ OR $R_1 \natural_{(\langle \text{join attributes } 1 \rangle), (\langle \text{join attributes } 2 \rangle)} R_2$
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 $Z = X \cup Y$ .	$R_1(Z) \div R_2(Y)$

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 46

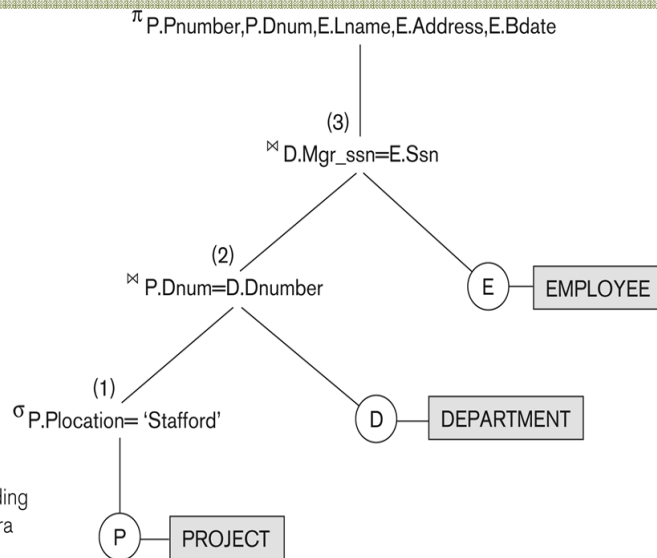
## Sorgu Ağacı

- Sorgunun veri yapısı ile gösterimi
- Düğümler: işlemler (select, project, join, ...)
- Yapraklar: (temel) ilişkiler
- Ağaç yapısı sorgunun karmaşıklığı konusunda bir fikir verir.
- Sorgu İyileştirme (query optimization)
  - Sorgunun/sorgu ağacının işlemleri hızlandıracak şekilde yeniden düzenlenmesi (**kitap 15.bölüm**)

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 47

## Örnek Sorgu Ağacı



**Figure 6.9**  
Query tree corresponding  
to the relational algebra  
expression for Q2.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 48



## Diğer işlemler: Özetleme İşlevleri ve Gruplama

- Özet bilgi örneği:
  - EMPLOYEE'lerin **ortalama** maaşları veya **toplam** sayıları.
  - İstatiksel amaçlı kullanılırlar...
- Temel özetleme işlevleri
  - SUM, AVERAGE, MAXIMUM, MINIMUM
  - COUNT: tuple veya değer saymakta kullanılır

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 49

## Özetleme İşlevi

- İşaret:  $\mathcal{F}$
- Kullanım:
  - $\mathcal{F}_{\text{MAX Salary}}(\text{EMPLOYEE})$ 
    - EMPLOYEE'deki en yüksek maaş değeri
  - $\mathcal{F}_{\text{MIN Salary}}(\text{EMPLOYEE})$ 
    - EMPLOYEE'deki en düşük maaş değeri
  - $\mathcal{F}_{\text{SUM Salary}}(\text{EMPLOYEE})$ 
    - EMPLOYEE'deki maaşların toplam değeri
  - $\mathcal{F}_{\text{COUNT SSN, AVERAGE Salary}}(\text{EMPLOYEE})$ 
    - EMPLOYEE'deki toplam çalışan sayısı ve ortalama maaşları

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 50

## Gruplayarak özetleme

- Örnek: Her bir bölüm için, bölüm no, bölümde çalışan sayısı, ve bölümde çalışanların ortalama maaşları
  - DNO, COUNT SSN, ve AVERAGE SALARY
  - $\rho_{R(Dno, No\_of\_employees, Average\_sal)} (Dno \mathcal{S} COUNT SSN, AVERAGE Salary (EMPLOYEE))$
  - Gruplama öznitelikleri sola, özetleme işlevleri sağa
  - Yukarıda:
    - Çalışanlar DNO (department number)'a göre gruplanır,
    - Her bir grup için çalışanlar (SSN'leri) sayılır
    - Her grup için maaş bilgilerinin ortalaması bulunur

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 51

## Örnek

**Figure 6.10**

The aggregate function operation.

- (a)  $\rho_{R(Dno, No\_of\_employees, Average\_sal)} (Dno \mathcal{S} COUNT SSN, AVERAGE Salary (EMPLOYEE))$ .  
 (b)  $Dno \mathcal{S} COUNT SSN, AVERAGE Salary (EMPLOYEE)$ .  
 (c)  $\mathcal{S} COUNT SSN, AVERAGE Salary (EMPLOYEE)$ .

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

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 52

## Örnek

**Figure 8.6**

Results of GROUP BY and HAVING. (a) Q24. (b) Q26.

(a)

Fname	Minit	Lname	Ssn	...	Salary	Super_ssn	Dno
John	B	Smith	123456789		30000	333445555	5
Franklin	T	Wong	333445555		40000	888665555	5
Ramesh	K	Narayan	666884444		38000	333445555	5
Joyce	A	English	453453453	...	25000	333445555	5
Alicia	J	Zelaya	999887777		25000	987654321	4
Jennifer	S	Wallace	987654321		43000	888665555	4
Ahmad	V	Jabbar	987987987		25000	987654321	4
James	E	Bong	888665555		55000	NULL	1

Grouping EMPLOYEE tuples by the value of Dno

Dno	Count (*)	Avg (Salary)
5	4	33250
4	3	31000
1	1	55000

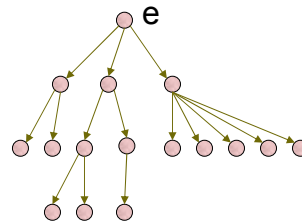
Result of Q24

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 53

## Diğer işlemler (Özyineli Kapama)

- Özyineli kapama (recursive closure) işlemi
  - Özyineli ilişkiye (recursive relationship) uygulanır.
  - Örnek: Bir EMPLOYEE e'nin "supervise" ettiği tüm EMPLOYEE'ler ve onların "supervise" ettiği tüm EMPLOYEE'leri bulmak.



Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 54

## Diğer işlemler (Özyineli Kapama)

- Özyineli kapama bir döngü mekanizması olmadan gerçekleştirilemez.
  - Ancak SQL3 standardı özyineli kapama (recursive closure) içermektedir.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 55

## Diğer işlemler (Özyineli Kapama)

(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  $\cup$  RESULT2)

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 56

## Diğer işlemler (OUTER JOIN)

- NATURAL JOIN ve EQUIJOIN'de ilişkili olmayan tuple'lar sonuç ilişkiye alınmamaktadır.
  - Null değerli join öznitelikleri de sonuca girmemektedir.
  - Bu bilgi kaybına yol açabilmektedir.
  - OUTER JOIN işlemleri ile ilişkili olsun olmasın, tüm R, tüm S, veya her iki ilişkideki tüm tuple'lar sonuç ilişkisine alınabilmektedir.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 57

## Diğer işlemler (OUTER JOIN)

- Left outer join ( $R \bowtie_{\text{left}} S$ )
  - Soldaki ilişkideki (R) bütün tuple'lar sonuç ilişkisine alınır
  - Eğer karşılık gelen (matching) S tuple'ı yoksa, S'den gelen öznitelikler null değerleri ile doldurulur.
- Right outer join ( $R \bowtie_{\text{right}} S$ )
  - Sağdaki ilişkideki (S) bütün tuple'lar sonuç ilişkisine alınır
  - Eğer karşılık gelen (matching) R tuple'ı yoksa, R'den gelen öznitelikler null değerleri ile doldurulur.
- Full outer join ( $R \bowtie_{\text{full}} S$ ) her iki ilişkiden tüm tuple'ları alır; karşılığı olmayan (non-matching) tuple'lara karşılık null değerleri iki tarafa eklenir.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 58

## Diğer işlemler (OUTER JOIN)

### 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

**Figure 6.12**  
The result of a  
LEFT OUTER JOIN  
operation.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 59

## Diğer işlemler (OUTER JOIN)

- OUTER JOIN ile tür uyumlu olmayan iki ilişkinin birleşimi (union) yapılmaktadır.
- Örneğin  $R(X, Y)$  ve  $S(X, Z)$  ilişkileri yarım uyumludur (**partially compatible**); bazı öznitelikleri ( $X$  öznitelikleri) tür uyumludur.
- Bu durumda tür uyumlu öznitelikler sonuç ilişkisinde bir kere yer alır, her iki ilişkiden tür uyumlu olmayan öznitelikler de sonuçta yer alır:  $T(X, Y, Z)$

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 60

## Diğer işlemler (OUTER JOIN)

- Örnek:
  - STUDENT(Name, SSN, Department, Advisor)
  - INSTRUCTOR(Name, SSN, Department, Rank).
  - STUDENT  $\bowtie$  INSTRUCTOR
  - Name, SSN, Department öznitelikleri üzerinden karşılaştırılırlar.
  - Eğer öğrenci aynı zamanda instructor (öğretim elemanı) ise hem Advisor hem de Rank bilgileri eklenir, aksi halde bunlardan birisi null olarak atanır.
  - Sonuç ilişkisi:
    - **STUDENT\_OR\_INSTRUCTOR**  
(Name, SSN, Department, Advisor, Rank)
    - Bu ilişki bütün öğrenci ve instructor'ların birleşimidir (union)

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 61

## Örnekler: Yordamsal çözüm ile

- **Q1: Retrieve the name and address of all employees who work for the 'Research' department** ('Research' bölümü çalışanlarının ad ve adresi)  
$$\text{RESEARCH\_DEPT} \leftarrow \sigma_{\text{DNAME}='Research'}(\text{DEPARTMENT})$$
$$\text{RESEARCH\_EMPS} \leftarrow (\text{RESEARCH\_DEPT} \bowtie_{\text{DNUMBER}=\text{DNOEMPLOYEE}} \text{EMPLOYEE})$$
$$\text{RESULT} \leftarrow \pi_{\text{FNAME}, \text{LNAME}, \text{ADDRESS}}(\text{RESEARCH\_EMPS})$$
- **Q6: Retrieve the names of employees who have no dependents** (dependent'i olmayan çalışanların adları)  
$$\text{ALL\_EMPS} \leftarrow \pi_{\text{SSN}}(\text{EMPLOYEE})$$
$$\text{EMPS\_WITH\_DEPS}(\text{SSN}) \leftarrow \pi_{\text{ESSN}}(\text{DEPENDENT})$$
$$\text{EMPS\_WITHOUT\_DEPS} \leftarrow (\text{ALL\_EMPS} - \text{EMPS\_WITH\_DEPS})$$
$$\text{RESULT} \leftarrow \pi_{\text{LNAME}, \text{FNAME}}(\text{EMPS\_WITHOUT\_DEPS} * \text{EMPLOYEE})$$

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 62

## Örnekler: Tek ifade ile

Yukarıdaki sorgular tek ifade ile de yazılabilir:

- **Q1: Retrieve the name and address of all employees who work for the 'Research' department.** ('Research' bölümü çalışanlarının ad ve adresi)

$$\pi_{\text{Fname, Lname, Address}} (\sigma_{\text{Dname='Research'}} (\text{DEPARTMENT} \bowtie_{\text{Dnumber=Dno}} (\text{EMPLOYEE})))$$

- **Q6: Retrieve the names of employees who have no dependents.**

$$\pi_{\text{Lname, Fname}} ((\pi_{\text{Ssn}} (\text{EMPLOYEE}) - \rho_{\text{Ssn}} (\pi_{\text{Essn}} (\text{DEPENDENT}))) * \text{EMPLOYEE})$$

## İlişkisel Hesaplama (Relational Calculus - RC)

- RC ifadeleri yeni bir ilişki tanımlı yapmada kullanılır,
- Yeni ilişki, değişkenler kullanılarak tanımlanır:
  - *Tuple değişkenleri* ile VT ilişkilerinin tuple'ları kullanılarak yeni ilişki tanımlanabilir (**Tuple Relational Calculus - TRC**)
  - *Sütun değişkenleri* ile VT ilişkilerinin öznitelikleri kullanılarak yeni ilişki tanımlanabilir (**Domain Relational Calculus - DRC**).
- Hesaplama (calculus) ifadeleri sonuç ilişkisinin nasıl elde edileceğini (işlemler sırası) değil, sonuç ilişkisinin ne içereceğini (tanımsal - declarative) tanımlar.
  - RA ve RC arasındaki temel fark budur.



## RC vs. RA

- RC: yordamsal olmayan veya tanımsal bir dil
  - **nonprocedural** or **declarative** language.
- RA: Sonucu elde etmek için gerekli işlemleri tanımlar; yordamsal bir dildir.
  - **procedural**

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 65

## Tuple Relational Calculus (TRC)

- Tuple değişkenleri tanımlanarak yazılır.
- Tuple değişkenleri bir VT ilişkisi üzerinden tanımlanır;
  - Değişken değerleri o ilişkiden gelir.
- Basit TRC sorgusu:
$$\{t \mid \text{COND}(t)\}$$
  - $t$ : tuple değişkeni
  - $\text{COND}(t)$ :  $t$ 'yi kullanan bir koşul.
  - Sonuç ilişkisi:  $\text{COND}(t)$ 'yi sağlayan tuple'lar.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 66

## Tuple Relational Calculus (TRC)

- Örnek: Find the first and last names of all employees whose salary is above \$50,000 (maaşı 50K'den fazla olan çalışanlar)  
 $\{t.FNAME, t.LNAME \mid EMPLOYEE(t) \text{ AND } t.SALARY > 50000\}$
- $EMPLOYEE(t)$ : t'nin değerleri  $EMPLOYEE$  tuple'larından gelir (**range relation**).
- $t.SALARY > 50000$  ile t tuple'ları filtrelenir
  - (SELECTION  $\sigma_{SALARY > 50000}$ )
- Sonuç t tuple'larının FNAME ve LNAME sütunları alınır
  - (PROJECTION  $\pi_{FNAME, LNAME}$ )
- Yukarıdaki işlemler belirtilen sırada yapılmak zorunda değil! TRC sadece maaşı 50K üzeri olan  $EMPLOYEE$  tuple'larının FNAME ve LNAME bilgilerinin sonuç ilişkisini oluşturacağını tanımlamaktadır.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 67

## The Existential and Universal Quantifiers (Varoluşsal ve Evrensel Niceleyiciler)

- Universal quantifier ( $\forall$ ): Hepsi
- Existential quantifier ( $\exists$ ): Bazısı
- Tuple değişkeni t
  - Sınırlı (bound): Eğer ( $\forall t$ ) veya ( $\exists t$ ) teriminde yer alıyorsa; veya
  - Serbest (free).
- F formülü, ve t tuple değişkeni için,
  - ( $\exists t$ )(F) formülü
    - Doğru: Eğer F formülünü doğru yapan en az bir t tuple'ı varsa
    - Yanlış: F'i doğru yapan hiç t tuple'ı yoksa.
  - ( $\forall t$ )(F) formülü
    - Doğru: Eğer bütün t tuple'ları F formülünü doğru yapıyorsa,
    - Yanlış: F'i yanlış yapan bir t tuple'ı varsa.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 68

## Örnek: Varoluşsal Niceleyici Sorgusu

- 'Research' bölümü çalışanlarının ad ve adresleri.  
Sorgunun TRC ile ifadesi:  
$$\{t.FNAME, t.LNAME, t.ADDRESS \mid EMPLOYEE(t) \text{ and } (\exists d) (DEPARTMENT(d) \text{ and } d.DNAME='Research' \text{ and } d.DNUMBER=t.DNO) \}$$
- $t$ : *free tuple variables* (serbest tuple değişkeni)
- $T$  için koşullar denetlenir, eğer ifade herhangi bir  $t$  tuple'ı için doğruysa,  $t$ 'nin  $FNAME$ ,  $LNAME$ , ve  $ADDRESS$  bilgileri sonuca eklenir.
- $EMPLOYEE(t)$  ve  $DEPARTMENT(d)$  koşulları  $t$  ve  $d$ 'nin alabileceği değerleri belirler.
- $d.DNAME = 'Research'$  (filtreleme/seçme koşulu)
  - **SELECT işlemi**
- $d.DNUMBER = t.DNO$  (JOIN koşulu)

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 69

## Örnek: Evrensel Niceleyici Sorgusu

- 5 nolu bölümün kontrol ettiği tüm projelerde çalışan çalışanların adları:  
$$\{e.LNAME, e.FNAME \mid EMPLOYEE(e) \text{ and } ((\forall x)(\text{not}(\text{PROJECT}(x)) \text{ or } \text{not}(x.DNUM=5)) \text{ OR } (\exists w)(\text{WORKS\_ON}(w) \text{ and } w.ESSN=e.SSN \text{ and } x.PNUMBER=w.PNO))))\}$$
- Exclude from the universal quantification all tuples that we are not interested in by making the condition true *for all such tuples*.
  - The first tuples to exclude (by making them evaluate automatically to true) are those that are not in the relation  $R$  of interest.
- In query above, using the expression  $\text{not}(\text{PROJECT}(x))$  inside the universally quantified formula evaluates to true all tuples  $x$  that are not in the  $\text{PROJECT}$  relation.
  - Then we exclude the tuples we are not interested in from  $R$  itself. The expression  $\text{not}(x.DNUM=5)$  evaluates to true all tuples  $x$  that are in the project relation but are not controlled by department 5.
- Finally, we specify a condition that must hold on all the remaining tuples in  $R$ .  
$$(\exists w)(\text{WORKS\_ON}(w) \text{ and } w.ESSN=e.SSN \text{ and } x.PNUMBER=w.PNO)$$

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 70

## Languages Based on Tuple Relational Calculus

- The language **SQL** is based on tuple calculus. It uses the basic block structure to express the queries in tuple calculus:
  - **SELECT** <list of attributes>
  - **FROM** <list of relations>
  - **WHERE** <conditions>
- **SELECT** clause mentions the attributes being projected, the **FROM** clause mentions the relations needed in the query, and the **WHERE** clause mentions the selection as well as the join conditions.
  - SQL syntax is expanded further to accommodate other operations. (See Chapter 8).

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 71

## Languages Based on Tuple Relational Calculus (Contd.)

- Another language which is based on tuple calculus is **QUEL** which actually uses the range variables as in tuple calculus. Its syntax includes:
  - **RANGE OF** <variable name> **IS** <relation name>
- Then it uses
  - **RETRIEVE** <list of attributes from range variables>
  - **WHERE** <conditions>
- This language was proposed in the relational DBMS **INGRES**. (system is currently still supported by Computer Associates – but the QUEL language is no longer there).

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 72

## The Domain Relational Calculus

- Another variation of relational calculus called the domain relational calculus, or simply, domain calculus is equivalent to tuple calculus and to relational algebra.
- The language called QBE (Query-By-Example) that is related to domain calculus was developed almost concurrently to SQL at IBM Research, Yorktown Heights, New York.
  - Domain calculus was thought of as a way to explain what QBE does.
- Domain calculus differs from tuple calculus in the type of variables used in formulas:
  - Rather than having variables range over tuples, the variables range over single values from domains of attributes.
- To form a relation of degree  $n$  for a query result, we must have  $n$  of these domain variables— one for each attribute.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 73

## The Domain Relational Calculus (Contd.)

- An expression of the domain calculus is of the form  
 $\{ x_1, x_2, \dots, x_n \mid$   
 $\text{COND}(x_1, x_2, \dots, x_n, x_{n+1}, x_{n+2}, \dots, x_{n+m})\}$ 
  - where  $x_1, x_2, \dots, x_n, x_{n+1}, x_{n+2}, \dots, x_{n+m}$  are domain variables that range over domains (of attributes)
  - and COND is a condition or formula of the domain relational calculus.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 74

## Example Query Using Domain Calculus

Retrieve the birthdate and address of the employee whose name is 'John B. Smith'.

■ Query :

$\{uv \mid (\exists q) (\exists r) (\exists s) (\exists t) (\exists w) (\exists x) (\exists y) (\exists z)$   
 $(\text{EMPLOYEE}(qrstuvwxyz) \text{ and } q='John' \text{ and } r='B' \text{ and } s='Smith')\}$

- **Abbreviated notation** **EMPLOYEE(qrstuvwxyz)** uses the variables without the separating commas: **EMPLOYEE(q,r,s,t,u,v,w,x,y,z)**
- Ten variables for the employee relation are needed, one to range over the domain of each attribute in order.
  - Of the ten variables q, r, s, . . . , z, only u and v are free.
- Specify the *requested attributes*, BDATE and ADDRESS, by the free domain variables u for BDATE and v for ADDRESS.
- Specify the condition for selecting a tuple following the bar ( | )—
  - namely, that the sequence of values assigned to the variables qrstuvwxyz be a tuple of the employee relation and that the values for q (FNAME), r (MINIT), and s (LNAME) be 'John', 'B', and 'Smith', respectively.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 75

## QBE: A Query Language Based on Domain Calculus (Appendix C)

- This language is based on the idea of giving an example of a query using “example elements” which are nothing but domain variables.
- Notation: An example element stands for a domain variable and is specified as an example value preceded by the underscore character.
- P. (called **P dot**) operator (for “print”) is placed in those columns which are requested for the result of the query.
- A user may initially start giving actual values as examples, but later can get used to providing a minimum number of variables as example elements.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 76

## QBE: A Query Language Based on Domain Calculus (Appendix C)

- The language is very user-friendly, because it uses minimal syntax.
- QBE was fully developed further with facilities for grouping, aggregation, updating etc. and is shown to be equivalent to SQL.
- The language is available under QMF (Query Management Facility) of DB2 of IBM and has been used in various ways by other products like ACCESS of Microsoft, and PARADOX.
- For details, **see Appendix C** in the text.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 77

## QBE Examples

- QBE initially presents a relational schema as a “blank schema” in which the user fills in the query as an example:

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 78

## Example Schema as a QBE Query Interface

### EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
-------	-------	-------	-----	-------	---------	-----	--------	-----------	-----

### DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date
-------	---------	---------	----------------

### DEPT\_LOCATIONS

Dnumber	Dlocation
---------	-----------

### PROJECT

Pname	Pnumber	Plocation	Dnum
-------	---------	-----------	------

### WORKS\_ON

Essn	Pno	Hours
------	-----	-------

### DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship
------	----------------	-----	-------	--------------

**Figure C.1**

The relational schema of Figure 5.5 as it may be displayed by QBE.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 79

## QBE Examples

- The following domain calculus query can be successively minimized by the user as shown:

- Query :

$\{uv \mid (\exists q) (\exists r) (\exists s) (\exists t) (\exists w) (\exists x) (\exists y) (\exists z)$   
 $(\text{EMPLOYEE}(qrstuvwxyz) \text{ and } q=\text{'John'} \text{ and } r=\text{'B'} \text{ and } s=\text{'Smith'})\}$

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 80



## Four Successive ways to specify a QBE Query

(a) EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	_123456789	P_9/1/60	P_100 Main, Houston, TX	_M	_25000	_123456789	_3

(b) EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith		P_9/1/60	P_100 Main, Houston, TX				

(c) EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith		P_X	P_Y				

(d) EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith		P.	P.				

**Figure C.2**

Four ways to specify the query Q0 in QBE.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 81

## QBE Examples

- Specifying complex conditions in QBE:
- A technique called the “condition box” is used in QBE to state more involved Boolean expressions as conditions.
- The C.4(a) gives employees who work on either project 1 or 2, whereas the query in C.4(b) gives those who work on both the projects.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 82

## Complex Conditions with and without a condition box as a part of QBE Query

(a)

Essn	Pno	Hours
P.		> 20

**WORKS\_ON**

(b)

Essn	Pno	Hours
P.	_PX	_HX

**CONDITIONS**

_HX > 20 and (PX = 1 or PX = 2)
---------------------------------

**WORKS\_ON**

(c)

Essn	Pno	Hours
P.	1	> 20
P.	2	> 20

**Figure C.3**

Specifying complex conditions in QBE. (a) The query Q0A. (b) The query Q0B with a condition box. (c) The query Q0B without a condition box.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 83

## Handling AND conditions in a QBE Query

**Figure C.4**

Specifying EMPLOYEES who work on both projects. (a) Incorrect specification of an AND condition. (b) Correct specification.

**WORKS\_ON**

**(a)**

Essn	Pno	Hours
P_ES	1	
P_ES	2	

**WORKS\_ON**

**(b)**

Essn	Pno	Hours
P_EX	1	
P_EY	2	

**CONDITIONS**

_EX = _EY
-----------

**Figure C.4**

Specifying EMPLOYEES who work on both projects. (a) Incorrect specification of an AND condition. (b) Correct specification.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 84

## JOIN in QBE : Examples

- The join is simply accomplished by using the same example element (variable with underscore) in the columns being joined from different (or same as in C.5 (b)) relation.
- Note that the Result is set up as an independent table to show variables from multiple relations placed in the result.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 85

## Performing Join with common example elements and use of a RESULT relation

**Figure C.5**

Illustrating JOIN and result relations in QBE. (a) The query Q1. (b) The query Q8.

### (a) EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
_FN		_LN			_Addr				_DX

### DEPARTMENT

Dname	Dnumber	Mgrssn	Mgr_start_date
Research	_DX		

RESULT			
P.	_FN	_LN	_Addr

### (b) EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
_E1		_E2						_Xssn	
_S1		_S2	_Xssn						

RESULT				
P.	_E1	_E2	_S1	_S2

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 86

## AGGREGATION in QBE

- Aggregation is accomplished by using .CNT for count, .MAX, .MIN, .AVG for the corresponding aggregation functions
- Grouping is accomplished by .G operator.
- Condition Box may use conditions on groups (similar to HAVING clause in SQL – see Section 8.5.8)

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 87

## AGGREGATION in QBE : Examples

(a) EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
							P.CNT.		

(b) EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
							P.CNT.ALL		

(c) EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
			P.CNT.ALL				PAVG.ALL		P.G.

(d) PROJECT

Pname	Pnumber	Plocation	Dnum
P.	_PX		

WORKS\_ON

Essn	Pno	Hours
P.CNT.EX	G._PX	

CONDITIONS

CNT_EX > 2
------------

**Figure C.6**  
Functions and grouping in QBE. (a) The query Q23.  
(b) The query Q23A. (c) The query Q24. (d) The query Q26.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 88

## NEGATION in QBE : Example

**Figure C.7**

Illustrating negation by the query Q6.

### EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
P.		P.	_SX						

### DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship
_SX				

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 89

## UPDATING in QBE : Examples

### (a) EMPLOYEE

	Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
I.	Richard	K	Marini	653298653	30-Dec-52	98 Oak Forest, Katy, TX	M	37000	987654321	4

### (b) EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
D.			653298653						

### (c) EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
	John		Smith				U_S*1.1		U.4

**Figure C.8**

Modifying the database in QBE. (a) Insertion. (b) Deletion. (c) Update in QBE.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 6- 90

## 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
- Relational Calculus
  - Tuple Relational Calculus
  - Domain Relational Calculus
- Overview of the QBE language (appendix C)