CS281 Class Notes Week 5 (03.03.2015-06.03.2015)

Hatice Çözeli

Nurdan Tatar

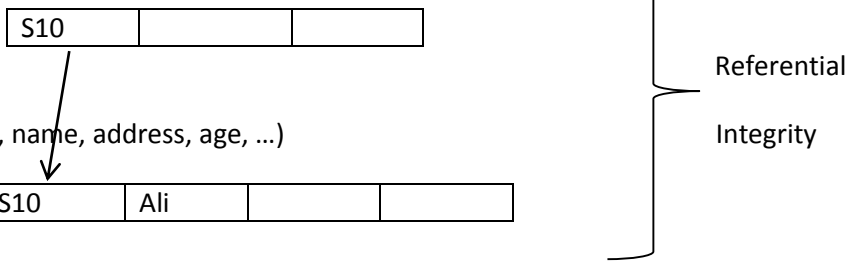
Buket Tuna

03.03.2015

Logical data independence

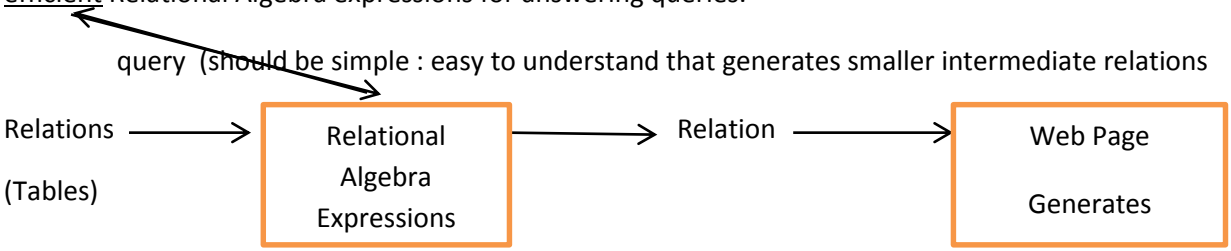
Hierarchies (from Chapter 3-Slide 28)

HourlyEmp(ssn, hwages, hworked)



CHAPTER 4 – Relational Algebra

Relational Algebra is used with the relational model for answering queries. We aim to write efficient Relational Algebra expressions for answering queries.



Operators

(+, /, *, - for numbers)

π : projection

choosing a column

σ : selection

choosing a row

⋈ : join, i.e $\bowtie_{\text{Enroll.sno = Student.sno}}$

∪ : union

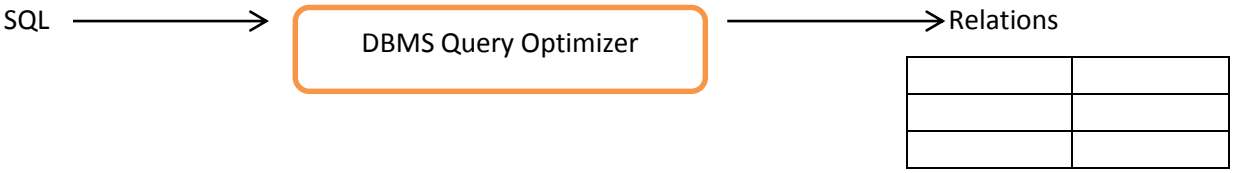
∩ : intersection

* π and σ are unary operators:

a = b + c, where + is a binary operator

a = -b , where – is an unary operator

There are some more operators.



Examples :

student(sno, stname, year, major)

enroll(sno, cno, grade)

course(cno, room)

1) Display student nos :
   $\pi_{sno}(student) = \pi_{sno}student$
2) Display stname :
   $R = \pi_{stname}(student)$
   If Student (sno, stname) =

10	Ali
20	Zeynep
30	Zeynep

,R(sname) =

Ali
Zeynep

~~Zeynep~~

06.03.2015

Set Oriented

∪

∩

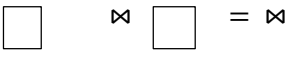
-

X (Cartesian product)

Relation Oriented

π : Projection 

σ : Selection 

⋈ : Join 

÷ : Division

R1(No, Name)

1	A
2	B
3	A

R2(No, Name)

1	A
3	A
4	C

- R = R1 ∪ R2 , R(No, Name)

1	A
2	B
3	A
4	C

*R1 has n1 tuples and R2 has n2 tuples, then R = R1 ∪ R2,

$$\max(n1,n2) \leq |R| \leq (n1+n2)$$

- R = R1 ∩ R2, R(No, Name)

1	A
3	A

* $0 \leq |R| \leq \min(n1,n2)$

- R = R1-R2, R(No, Name) *difference

2	B

* $0 \leq |R| \leq n1$

* R1 ∩ R2 = R1 – (R1-R2)

- R = R1 X R2 , R(R1.No, R1.Name, R2.No, R2.Name)

1	A	1	A
1	A	3	A
1	A	4	C
2	B	1	A
2	B	3	A
2	B	4	C
3	A	1	A
3	A	3	A
3	A	4	C

* R1 X R2 = R2 X R1

R1 ∪ R2 = R2 ∪ R1

R1-R2 ≠ R2-R1

Projection π : Relation Schemas

Employee(ENo, Name, Status)

1	Peter	A
2	Mary	B
3	Jane	A
10	Ann	C
11	Peter	A
12	Mary	A

π_{Name}(Employee)

Peter
Mary
Jane
Ann

~~Peter~~

~~Mary~~

* $\pi_{ENo, Name}$ →

$\pi_{ENo, Name, Status}$ →

$0 \leq |R| \leq |R|$

Selection σ :

R = $\sigma_{Status = A}$(Employee) , R(ENo, Name, Status)

1	Peter	A
3	Jane	A
11	Peter	A
12	Mary	A

* $0 \leq |R| \leq |R|$

- $\sigma_{ENo>10 \text{ v } (Status=B \text{ } \Lambda Name=\text{"Jane"})}$

Find employee name with Status A:

$\pi_{Name}(\sigma_{Status=A}(Employee))$ or

$\sigma_{Status=A}(\pi_{Name}(Employee))$

Find all ENo and Name with Status A or Status B:

$\pi_{ENo, Name}(\sigma_{Status=A \text{ v } Status=B}(Employee))$

$\pi_{ENo, Name}(\sigma_{Status=A}(Employee) \cup \sigma_{Status=B}(Employee))$

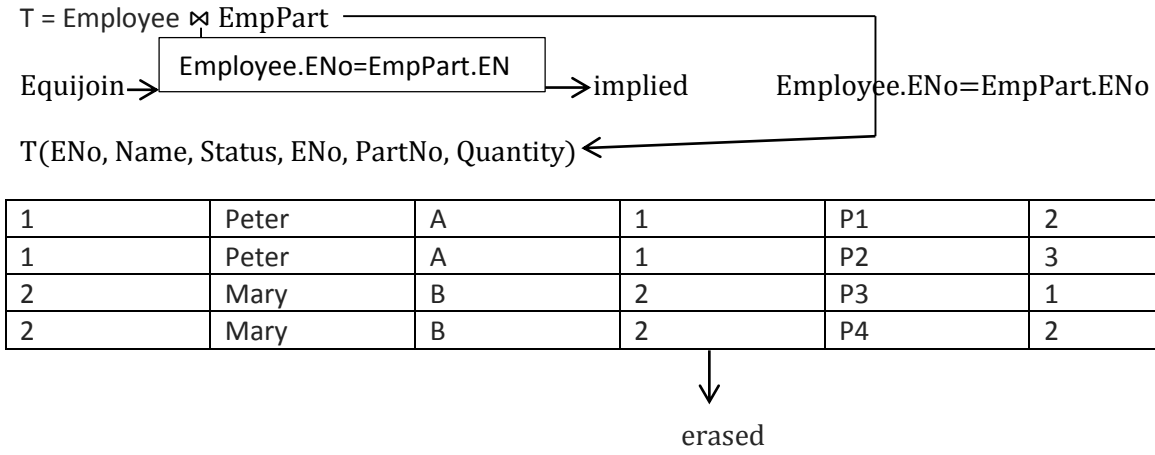
Join ⋈ :

Equijoin : Condition is explicit

Natural Join : Condition is implied

EmpPart(ENo, PartNo, Quantity)

1	P1	2
1	P2	3
2	P3	1
2	P4	2



Query : Find PartNo and quantities supplied by Employee with Status B.

- $\pi_{PartNo,Quantity}(\sigma_{Status=B}(Employee) \bowtie EmpPArt)$

T1(ENo,Name,Status)

2	Mary	B

T2(ENo, Name, Status, PartNo, Quantity)

2	Mary	B	P3	1
2	Mary	B	P4	2

- $\pi_{PartNo,Quantity}(\sigma_{Status=B}(Employee \bowtie EmpPart))$

$\pi(\sigma_{Status=A}(Employee \times EmpPart))$