

Bases de Datos Unidad 1

Instructor: M.C. Luis Basto Díaz
Email: luisbasto@gmail.com

Modelo Relacional

- Definiciones
 - Relación
 - Tabla
 - Dominio
 - Tupla
 - Esquemas
- Álgebra Relacional
- Lenguaje SQL
- Integridad Referencial
- Normalización
 - Primera, segunda y tercera Forma Normal
 - Forma Normal de Boyce - Codd
 - Criterios de Normalización

Álgebra relacional

- Es un lenguaje teórico con operaciones que se aplican a una o más relaciones, con el fin de definir otra relación sin modificar las relaciones originales.
- Codd(1972) propuso ocho operaciones para el álgebra operacional, después se propusieron más.

3

Álgebra relacional

- El álgebra relacional trabaja con una o más relaciones que define otra relación SIN cambiar las relaciones originales.
- Los operandos y resultados son relaciones.
- El resultado de una operación puede llegar a ser entrada de otra operación.
- Se permiten anidar expresiones en álgebra relacional, de la misma forma que se anidan operaciones aritméticas.

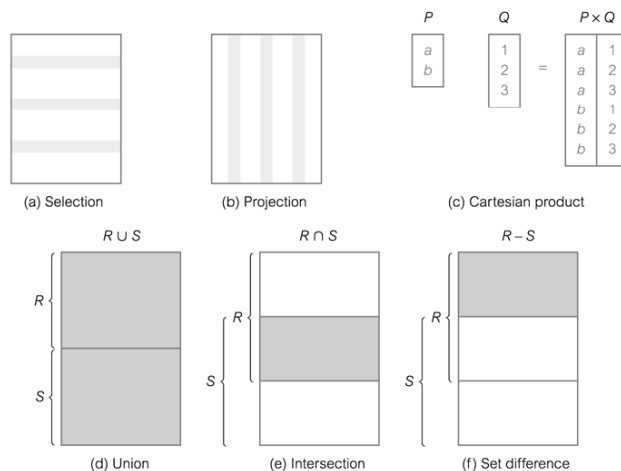
4

Operaciones en el álgebra relacional

- Operaciones básicas
 - Selección
 - Proyección
 - Producto cartesiano
 - Unión
 - Diferencia de conjuntos
- Otras operaciones (basadas en las anteriores)
 - Asociación (join)
 - Intersección
 - División

5

Operaciones en el álgebra relacional



© Pearson Education Limited 1995, 2005

6

The diagram illustrates five types of natural join results between two tables, T and U .

- $T \bowtie U$ (Natural join):** The result is a 2x3 table containing all combinations of rows from T and U .

A	B	C
a	1	x
a	1	y
b	2	x
b	2	y
- $T \bowtie_B U$ (Semijoin):** The result is a 2x2 table containing only the rows from T where the value of attribute B matches a value in U .

A	B
a	1
a	1
- $T \bowtie_{B,C} U$ (Left Outer join):** The result is a 2x3 table containing all rows from T , with C values from U where they match.

A	B	C
a	1	x
a	1	y
b	2	x
b	2	
- $T \bowtie_{B^2} U$ (Natural join):** The result is a 2x3 table containing all combinations of rows from T and U .

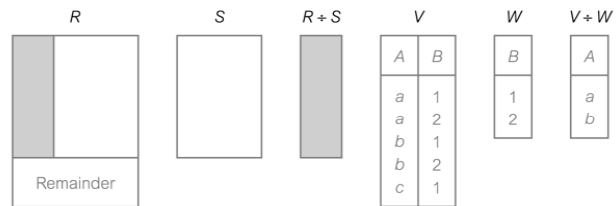
A	B	C
a	1	x
a	1	y
b	2	x
b	2	y
- $T \bowtie_{C^2} U$ (Natural join):** The result is a 2x3 table containing all combinations of rows from T and U .

A	B	C
a	1	x
a	1	y
b	2	x
b	2	y

(g) Natural join

(h) Semijoin

(i) Left Outer join



(j) Divis on (shaded area)

Example of division

© Pearson Education Limited 1995, 2005

StaffNo	fName	lName	Position	sex	DOB	salary	branchNo
SL21	john	white	manager	M	01-Oct-45	30000	B005
SG37	Ann	Beech	Assitant	F	10-Nov-60	12000	B003
SG14	David	Ford	Supervisor	M	24-Mar-67	18000	B003
SA9	Mary	Howe	Assitant	F	19-Feb-56	9000	B007
SG5	Susan	Brand	Manager	F	03-Jun-58	24000	B003
SL41	Julie	Lee	Assitant	F	13-Jun-62	9000	B005

Operaciones unarias

- Selección

- Se aplica a una única relación R y define otra relación que contiene únicamente aquellas tuplas de R que satisfacen la condición (predicado) especificada.

- $\sigma_{\text{predicado}}(R)$

- Ejemplo: Enumerar todos los salarios del personal cuyo salario sea superior a \$10,000.

$\sigma_{\text{salario} > 10000}(\text{Staff})$

9

StaffNo	fName	lName	Position	sex	DOB	salary	branchNo
SL21	john	white	manager	M	01-Oct-45	30000	B005
SG37	Ann	Beech	Assitant	F	10-Nov-60	12000	B003
SG14	David	Ford	Supervisor	M	24-Mar-67	18000	B003
SG5	Susan	Brand	Manager	F	03-Jun-58	24000	B003

10

Operaciones unarias

- Proyección

- La operación de proyección se aplica a una única relación R y define otra relación que contiene un subconjunto vertical de R, extrayendo los valores de los atributos especificados y eliminando los duplicados.

- $\Pi_{a_1, \dots, a_n}(R)$

- Ejemplo: Generar una lista de salarios par todo el personal, mostrando solamente los detalles referidos a los atributos staffNo, fName, lName y Salary

$$\Pi_{\text{staffNo}, \text{fName}, \text{lName}, \text{Salary}}(\text{Staff})$$

11

StaffNo	fName	lName	salary
SL21	john	white	30000
SG37	Ann	Beech	12000
SG14	David	Ford	18000
SA9	Mary	Howe	9000
SG5	Susan	Brand	24000
SL41	Julie	Lee	9000

12

Operaciones de conjunto

- Las operaciones de selección y proyección extraen información de una única relación.
- Se puede combinar información de diversas relaciones.
- Operaciones de conjuntos son: Conjuntos de unión, diferencia de conjuntos, intersección y producto cartesiano.

13

Operaciones de conjunto

- Unión
 - La unión de dos relaciones R y S define una relación que contiene todas las tuplas R, de S o tanto de R como de S, eliminándose las tuplas duplicadas. R y S deben ser compatibles con respecto a la unión.
- Si R y S tienen tuplas I y J, respectivamente, su unión se obtiene concatenándolas en una única relación con un máximo de $(i + j)$ tuplas.

14

Operaciones de conjunto

- La unión es posible sólo si los esquemas de las dos relaciones se corresponden, si tienen el mismo número de atributos y cada pareja de atributos correspondientes tienen el mismo dominio.
- Ejemplo: Enumerar todas las ciudades en las que exista una sucursal o un inmueble en alquiler.

15

$\Pi_{\text{city}}(\text{Branch}) \cup \Pi_{\text{city}}(\text{PropertyForrent})$

- Primero se utiliza la operación de proyección para proyectar las relaciones Branch y PropertyForRent sobre el atributo city, eliminando los duplicados según sea necesario.

16

City
London
Aberdeen
Glasgow
Bristol

17

Diferencia de conjuntos

- Define una relación compuesta por las tuplas que se encuentran en la Relación R, pero no en S. R y S deben ser compatibles con respecto a la unión.
- Ejemplo: enumerar todas las ciudades en las que exista una sucursal, pero no haya inmuebles en alquiler.

$$\Pi_{\text{city}}(\text{Branch}) - \Pi_{\text{city}}(\text{PropertyForrent})$$

18

City
Bristol

19

Intersección

- Define una nueva relación compuesta por el conjunto de todas las tuplas que existen tanto en R como en S. R y S deben ser compatibles con respecto a la unión.
- Ejemplo: enumerar todas las ciudades en la que exista tanto una sucursal como al menos un inmueble en el alquiler.

$$\Pi_{\text{city}}(\text{Branch}) \cap \Pi_{\text{city}}(\text{PropertyForrent})$$

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

20

City
London
Aberdeen
Glasgow

21

Producto cartesiano

- Esta operación define una relación que es la concatenación de cada tupla de la relación R con cada tupla de la relación S.
- Ejemplo: enumerar los nombres y comentarios de todos los clientes que hayan visto un inmueble en alquiler.

$$\begin{aligned} & \Pi_{\text{clienteNo}, \text{fName}, \text{lName}}(\text{Client}) && \times \\ & \Pi_{\text{clienteNo}, \text{propertyNo}, \text{Comment}}(\text{viewing}) \\ & \sigma_{\text{client.clientNo} = \text{viewing.clientNo}} (\Pi_{\text{clienteNo}, \text{fName}, \text{lName}}(\text{Client}) \times \\ & \Pi_{\text{clienteNo}, \text{propertyNo}, \text{Comment}}(\text{viewing})) \end{aligned}$$

22

Client.ClientNo	fName	lName	Viewing.clientNo	propertyNo	Comment

23

Descomposición de operaciones

- Las operaciones complejas del álgebra relacional pueden descomponerse en operaciones más simples.
- TempViewing(clientNo,propertyNo,comment) \leftarrow
 $\Pi_{\text{clienteNo,propertyNo,Comment}}(\text{viewing})$
- TempClient(clientNo,fname,lname) \leftarrow
 $\Pi_{\text{clienteNo,fName,lName}}(\text{Client})$
- Comment(clientNo,fname,lname,vclientNo,propertyNo,comment) \leftarrow TempClient X TempViewing
- Result $\leftarrow \sigma_{\text{clientNo} = \text{clientNo}}$

24

Ejemplos

1. Encontrar todos los empleados cuya edad sea mayor a 45 años
2. Devolver el apellido y el nombre de los empleados que sean mayor de 45 años.
3. Listar todos los empleados que tengan nombre "Raquel" o apellido "Pinol".
4. Listar los nombres de los departamentos en las que el departamento tenga algún proyecto.

25

Ejemplos

- Seleccionar todos los autos que sean modelo arriba del '90.
- Seleccionar el id del propietario que tenga auto de marca 'Volcho'.
- Seleccionar Nombre, RFC y Curp del propietario con Id = 5.
- Seleccionar el estado del propietario con Id = 4.
- Seleccionar todos los propietarios cuyo ciudad sea 'Merida'.
- Seleccionar todas las ciudades del estado "Nuevo León"

26

Producto cartesiano y selección

- Usar la operación de selección para extraer aquellas tuplas donde *Client.clientNo* = *Viewing.clientNo*.

$\sigma_{Client.clientNo = Viewing.clientNo}((\Pi_{clientNo, fName, lName}(Client)) \times (\Pi_{clientNo, propertyNo, comment}(Viewing)))$

Client.clientNo	fName	lName	Viewing.clientNo	propertyNo	comment
-----------------	-------	-------	------------------	------------	---------

- El producto cartesiano y la selección se puede reducir a una sola operación llamada combinación (join).

27

Operaciones Join

- Join es derivado del producto cartesiano
- Es equivalente a realizar una selección sobre el producto cartesiano de dos relaciones.
- Es una de las operaciones más difíciles de implementar en un SMBD.
- Existen varias formas de operación del join.

28

Operaciones Join

- Theta Join
- Equijoin (tipo particular de Theta Join)
- Natural Join
- Outer Join
- Semijoin

29

Theta join (θ -join)

- $R \bowtie_P S$
 - Define una relación que contiene tuplas del producto cartesiano de R y S que satisfacen el predicado P.
 - El predicado P tiene la forma $R.a_i \theta S.b_i$, donde θ puede ser uno de los operadores de comparación ($<, \leq, >, \geq, =, \neq$).

30

Theta join (θ -join)

- Esta operación se puede reescribir utilizando el producto cartesiano de la siguiente manera:

$$R \bowtie_P S = \sigma_P(R \times S)$$

- El grado de la operación theta join es la suma de los grados de las relaciones R y S.
- Si el predicado P contiene solo la igualdad (=), se usa el término Equijoin.

31

Equijoin

- Listar los nombres y comentarios de todos los clientes que hayan visitado un inmueble en alquiler.

$(\Pi_{\text{clientNo}, \text{fName}, \text{lName}} (\text{Client}))$

$\bowtie \text{Client.clientNo} = \text{Viewing.clientNo}$

$(\Pi_{\text{clientNo}, \text{propertyNo}, \text{comment}} (\text{Viewing}))$

32

Natural Join

- $R \bowtie S$
 - Es un equijoin entre las dos relaciones R y S sobre todos los atributos comunes x.
 - Se elimina una de las dos apariciones de cada atributo común.
 - Lleva a acabo un equijoin sobre todos los atributos de las dos relaciones que tengan el mismo nombre.
 - El grado de esta operación es la suma de los grados de las relaciones R y S, menos el número de atributos de x.

33

Natural Join

- Listar los nombres y comentarios de todos los clientes que hayan visitado un inmueble en alquiler.

$(\Pi_{\text{clientNo}, \text{fName}, \text{lName}} (\text{Client})) \bowtie$
 $(\Pi_{\text{clientNo}, \text{propertyNo}, \text{comment}} (\text{Viewing}))$

34

Outer Join

- Al combinar dos relaciones, una tupla de una relación no tiene ninguna tupla correspondiente en la otra.
- Se utiliza cuando se desea que en el resultado aparezcan las tuplas de una de las relaciones aún cuando no existan valores correspondientes en la otra relación.
- A los valores no existentes en la segunda relación se les asigna un valor nulo.

35

Outer Join

- $R \bowtie S$
 - Se incluyen en la relación resultante las tuplas de R que no tengan valores correspondientes en los atributos comunes de S. (Combinación externa izquierda, Left outer Join)
 - Tuplas de S que no tengan valores correspondiente de en los atributos comunes de R (Combinación externa derecha, Right outer Join).
 - Conserva todas las tuplas de ambas relaciones. (Combinación externa completa, Full outer Join).

36

Outer Join

- Producir un reporte de estado sobre las visitas a los inmuebles.

$\Pi_{\text{propertyNo, street, city}} (\text{PropertyForRent}) \bowtie \text{Viewing}$

37

Semijoin

- $R \Join_p S$
 - Define una relación que contiene las tuplas de R que participan en la combinación de R con S.
 - Realiza una combinación de las dos relaciones y luego la proyecta sobre los atributos de la primera relación.

38

Semijoin

- Enumerar los detalles completos de todos los empleados que trabajen en la sucursal de Glasgow.

Staff \bowtie Staff.branchNo=Branch.branchNo ($\sigma_{\text{city}='Glasgow'}(\text{Branch})$)

39

División

- $R \div S$
 - Suponga que la relación R está definida sobre el conjunto de atributos A y que la relación S está definida sobre el conjunto B, de modo que $B \subseteq A$.
 - Sea $C = A - B$, el conjunto de atributos de R que no son de S.
 - La división: Define una relación sobre los atributos C que está compuesta por el conjunto de tuplas de R que se corresponden con la combinación de todas las tuplas de S.

40

División

- Se puede expresar la división con las operaciones básicas:

$$T1 \leftarrow \Pi_C(R)$$

$$T2 \leftarrow \Pi_C((S \times T1) - R)$$

$$T \leftarrow T1 - T2$$

41

División

- Identificar todos los clientes que hayan visto todos los inmuebles con tres habitaciones.
 - Seleccionamos todos los inmuebles con tres habitaciones: $R1 \leftarrow \sigma_{rooms = 3} (PropertyForRent)$
 - Proyectamos para obtener el número de los inmuebles de la selección anterior: $R2 \leftarrow \Pi_{propertyNo}(R1)$.
 - Utilizamos la división: $\Pi_{clientNo, propertyNo}(Viewing) \div R2$

42