

Lenguajes de consulta (Query Languages)

- **Poder:**
Cuál es la “*Máquina de Turing*” en bases de datos?
- **Estilo:**
procedural / no procedural

maneja conjuntos / maneja tuplas

basado en logica / en álgebra / en lenguaje natural

Algebra Relacional

Lenguaje de consulta del modelo relacional que permite expresar las consultas mediante la composición de operaciones que se aplican sobre relaciones.

Se aplica sobre las tuplas de las relaciones y el resultado de cada operación es una nueva relación, que se puede manipular en una ocasión futura.

Operadores del algebra relacional:

Selección	Unión	Intersección
Proyección	Diferencia	Join(Reunión)
	Producto Cartesiano	Join Natural
		Cociente

Selección σ

Sirve para seleccionar un conjunto de tuplas de una relación que cumplan una condición.

La condición de selección se aplica sobre los atributos de la relación.

$\sigma_{\langle \text{condicion sobre atributos} \rangle} (\langle \text{Nombre Relación} \rangle)$

Condición:

$\langle \text{nombre atributo} \rangle \langle \text{operador de comparación} \rangle \langle \text{valor constante} \rangle$

operador de comparación: { = , < , ≤ , > , ≥ , ≠ }

Las condiciones de selección se combinan con condiciones booleanas: AND, OR, NOT.

Ejemplo de Selección

Empleados

Nombre	Nro.	Depto.	Salario
Federico Azurra	333456	5	7850
Paola Valdés	453321	4	6980
Héctor Torres	889546	5	8456

Seleccionar los empleados que trabajan en el departamento 4.

$\sigma_{\text{Depto.} = 4} (\text{Empleados})$

Nombre	Nro.	Depto.	Salario
Paola Valdés	453321	4	6980

Ejemplo de Selección

Empleados

Nombre	Nro.	Depto.	Salario
Federico Azurra	333456	5	7850
Paola Valdés	453321	4	6980
Héctor Torres	889546	5	8456

Seleccionar los empleados cuyo salario es mayor que 7000.

$\sigma_{\text{Salario} > 7000} (\text{Empleados})$

Nombre	Nro.	Depto.	Salario
Federico Azurra	333456	5	7850
Héctor Torres	889546	5	8456

Ejemplo de Selección

Empleados

Nombre	Nro.	Depto.	Salario
Federico Azurra	333456	5	7850
Paola Valdés	453321	4	6980
Héctor Torres	889546	5	8456

Seleccionar los empleados que trabajan en el departamento 5 cuyo salario es mayor que 8000.

$\sigma_{\text{Depto.} = 5} (\sigma_{\text{Salario} > 8000} (\text{Empleados}))$

$\sigma_{\text{Salario} > 8000} (\sigma_{\text{Depto.} = 5} (\text{Empleados}))$

$\sigma_{\text{Depto.} = 5 \text{ AND Salario} > 8000} (\text{Empleados})$

Nombre	Nro.	Depto.	Salario
Héctor Torres	889546	5	8456

Proyección π

Sirve para seleccionar un conjunto de atributos de una relación.

$$\pi_{\langle \text{atributos} \rangle} (\langle \text{Nombre Relación} \rangle)$$

El resultado es una relación o sea un conjunto de tuplas con los atributos indicados en la proyección.

La aridad, o grado, de la relación resultado es igual al número de atributos por los que se proyecta.

Ejemplo de Proyección

Empleados

Nombre	Sexo	Depto.	Salario
Federico Azurra	M	5	7500
Paola Valdés	F	4	6980
Héctor Torres	M	5	7500

Listar el nombre y el salario de todos los empleados.

$$\pi_{\text{Nombre, Salario}} (\text{Empleados})$$

Nombre	Salario
Federico Azurra	7500
Paola Valdés	6980
Héctor Torres	7500

Ejemplo de Proyección

Empleados

Nombre	Sexo	Depto.	Salario
Federico Azurra	M	5	7500
Paola Valdés	F	4	6980
Héctor Torres	M	5	7500

Listar el sexo y el salario de todos los empleados.

$$\pi_{\text{Sexo, Salario}} (\text{Empleados})$$

Sexo	Salario
M	7500
F	6980
M	7500

Secuencia de Operaciones

Empleados

Nombre	Sexo	Depto.	Salario
Federico Azurra	M	5	7500
Paola Valdés	F	4	6980
Héctor Torres	M	5	7500

Listar el nombre y el salario de los empleados que trabajan en el departamento 5.

$$\pi_{\text{Nombre, Salario}} (\sigma_{\text{Depto.} = 5} (\text{Empleados}))$$

Nombre	Salario
Federico Azurra	7500
Héctor Torres	7500

Relaciones Compatibles

Decimos que dos relaciones son **compatibles** cuando sus esquemas cumplen las siguientes condiciones:

- Tienen la misma aridad **n**
- El i-ésimo atributo de cada relación tiene asociado el mismo dominio para todo $i \in [1, n]$.

$$\text{dom}(A_i) = \text{dom}(B_i) \text{ para } 1 \leq i \leq n$$

Unión \cup

Precondición:

Las relaciones deben ser compatibles.

Resultado:

$r \cup s$ produce una relación con las tuplas que están en **r**, o en **s**, o en ambas.

Ejemplo:

Mecánicos

Nombre	Salario	Empresa
A. Blanco	6500	Fast
J. Martin	5850	Giro

Choferes

Nombre	Salario	Empresa
A. Rey	4500	Veloz
S. Santos	4550	Fast

Mecánicos \cup Choferes

Nombre	Salario	Empresa
A. Blanco	6500	Fast
J. Martin	5850	Giro
A. Rey	4500	Veloz
S. Santos	4550	Fast

Ejemplo de Unión

Mecánicos

Nombre	Salario	Empresa
A.Blanco	6500	Fast
J.Martin	5850	Giro

Choferes

Nombre	Salario	Empresa
A.Rey	4500	Veloz
S.Santos	4550	Fast

Listar los nombres de los choferes y mecánicos que trabajan para la empresa Fast.

$\pi_{\text{Nombre}} (\sigma_{\text{Empresa} = \text{"Fast"}} (\text{Mecánicos} \cup \text{Choferes}))$

Nombre	Nombre	Salario	Empresa
A.Blanco	A.Blanco	6500	Fast
S.Santos	S.Santos	4550	Fast

Nombre	Salario	Empresa
A.Blanco	6500	Fast
J.Martin	5850	Giro
A.Rey	4500	Veloz
S.Santos	4550	Fast

Diferencia –

Precondición:

Las relaciones deben ser compatibles.

Resultado:

$r - s$ produce una relación con las tuplas que están en r pero que no están en s .

Ejemplo:

Mecánicos

Nombre	Salario	Empresa
A.Blanco	6500	Fast
J.Martin	5850	Giro

Choferes

Nombre	Salario	Empresa
J.Martin	5850	Giro
S.Santos	4550	Fast

Mecánicos - Choferes

Nombre	Salario	Empresa
A.Blanco	6500	Fast

Producto Cartesiano x

Precondición:

No tiene.

Resultado:

$R \times S$ produce una relación con los atributos de R seguidos de los atributos de S y donde cada tupla de R se combina con cada tupla de S .

$R(A_1, A_2, \dots, A_n)$, si R tiene k tuplas
 $S(B_1, B_2, \dots, B_j)$, si S tiene m tuplas

$R \times S (A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_j)$,
 $R \times S$ tiene $k \times m$ tuplas.

Ejemplo de Producto Cartesiano

Mecánicos

Nombre	Salario	CodEmp
A.Blanco	6500	1
J.Martin	5850	2

Empresas

NombreE	CodEmp	Telef
Giro	1	480 2424
Fast	2	905 6875

Listar el nombre de los mecánicos de la empresa Fast.

$\pi_{\text{Nombre}} (\sigma_{\text{CodEmp} = \text{Numero AND NombreE} = \text{"Fast"}} (\text{Mecánicos X Empresas}))$

Mecánicos X Empresas

Nombre	Salario	CodEmp	NombreE	Numero	Telef
A.Blanco	6500	1	Giro	1	480 2424
A.Blanco	6500	1	Fast	2	905 6875
J.Martin	5850	2	Giro	1	480 2424
J.Martin	5850	2	Fast	2	905 6875

Join (Reunión) \bowtie <condición>

Precondición:

No tiene.

Resultado:

$R \bowtie S$ produce una relación con los atributos de R seguidos de los atributos de S y donde cada tupla de R se combina con cada tupla de S donde se cumple la condición.

$R(A_1, A_2, \dots, A_n), S(B_1, B_2, \dots, B_j)$

$R \bowtie_{\langle \text{condición} \rangle} S (A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_j) =$
 $\sigma_{\langle \text{condición} \rangle} (R \times S)$

Ejemplo de Join (Reunión)

Mecánicos

Nombre	Salario	CodEmp
A.Blanco	6500	1
J.Martin	5850	2

Empresas

NombreE	Numero	Telef
Giro	1	480 2424
Fast	2	905 6875

Listar el nombre de los mecánicos de la empresa Fast.

$\pi_{\text{Nombre}} (\text{Mecánicos} \bowtie_{\text{CodEmp} = \text{Numero AND NombreE} = \text{"Fast"}} \text{Empresas})$

$\pi_{\text{Nombre}} (\sigma_{\text{CodEmp} = \text{Numero AND NombreE} = \text{"Fast"}} (\text{Mecánicos X Empresas}))$

Mecánicos \bowtie Empresas

Nombre	Salario	CodEmp	NombreE	Numero	Telef
J.Martin	5850	2	Fast	2	905 6875

Join Natural *

Precondición:

No tiene.

Resultado:

Es un join en que la condición es que los atributos que tengan igual nombre en las dos tablas tengan igual valor y sobre el cual los atributos de igual nombre se proyectan una sola vez.

Sean $R(A,B,C)$ y $S(X,Y,Z,A)$

$$R * S = \pi_{A,B,C,X,Y,Z} (R \bowtie_{R.A = S.A} S)$$

Ejemplo de Join Natural

Mecanicos

Nombre	Salario	CodEmp
A.Blanco	6500	1
J.Martin	5850	2

Empresas

NombreE	CodEmp	Telef
Giro	1	480 2424
Fast	2	905 6875

Listar el nombre de los mecánicos de la empresa Fast.

$$\pi_{\text{Nombre}} (\sigma_{\text{NombreE} = \text{"Fast"}} (\text{Mecanicos} * \text{Empresas}))$$

$$\pi_{\text{Nombre}} (\text{Mecanicos} \bowtie_{\text{CodEmp} = \text{CodEmp AND NombreE} = \text{"Fast"}} \text{Empresas})$$

Mecanicos * Empresas

Nombre	Salario	CodEmp	NombreE	Telef
J.Martin	5850	2	Fast	905 6875

Intersección \cap

Precondición:

Las relaciones deben ser compatibles.

Resultado:

$R \cap S$ produce una relación con las tuplas que están en R y en S a la vez.

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

Ejemplo de Intersección

Estudiantes

Nombre	Apellido
Adrian	Blanco
Juan	Martin
Estela	Ferraro
Julian	Estevez

Profesores

Nombre	Apellido
Americo	Suarez
Julian	Estevez
Estela	Ferraro

Listar los estudiantes que son también profesores.

Estudiantes \cap Profesores

EstudiantesProfesores

Nombre	Apellido
Julian	Estevez
Estela	Ferraro

Cociente %

Precondición:

Los atributos de S incluidos en los atributos de R .

Resultado:

Un conjunto de tuplas t tal que para toda tupla u de S , $t.u$ es una tupla de R .

R

A	B	C	D
a	b	c	d
a	b	e	f
b	c	e	f
e	d	c	d
e	d	e	f
a	b	d	e

S

C	D
c	d
e	f

$R \% S$

A	B
a	b
e	d

Cociente %

$R \% S$, donde $R(Z)$, $S(X)$ tal que $X \subseteq Z$ Sea $Y = Z - X \Rightarrow$

$$R \% S = \pi_Y(R) - \pi_Y(\pi_Y(R) \times S) - R$$

$\pi_Y(R)$

A	B
a	b
b	c
e	d

$$(\pi_Y(R) \times S) - R$$

A	B	C	D
b	c	c	d
b	c	c	d

$$\pi_Y(R) \times S$$

A	B	C	D
a	b	c	d
b	c	c	d
e	d	c	d
a	b	e	f
b	c	e	f
e	d	e	f

=

R

A	B	C	D
a	b	c	d
a	b	e	f
b	c	e	f
e	d	c	d
e	d	e	f
a	b	d	e

Ejemplo de Cociente

Empleados	
Nombre	NroE
Silva	213
Perez	214
Ramos	215
Soca	216

Trabaja_en	
NroE	NroP
213	1
213	2
214	1
215	1
215	2
216	2

Obtener los numeros de los empleados que trabajan en todos los proyectos donde trabaja Silva.

$PS = TRABAJA_EN * \sigma_{Nombre = "Silva"}(EMPLEADOS)$

$NPS = \pi_{NroP}(PS)$

$TOE = TRABAJA_EN - \pi_{NroP, NroE}(PS)$

Resultado = $\pi_{NroE}(TOE \div NPS)$

Algunos problemas del álgebra

JEFES

Empleado	Jefe
M.M.	R.R.
F.C.	R.R.
U.T.	R.R.
R.R.	G.C.
P.P.	J.G.
J.G.	G.C.
G.C.	J.N.

1) Cuántos son los jefes de M.M?

2) Obtener los nombres de los jefes directos o indirectos de M.M.

Obtener los jefes directos de M.M.

$R = \pi_{Jefe}(\sigma_{Empleado = "M.M."}(JEFES))$

Obtener los jefes de los jefes de M.M.

$S = \pi_{JEFES.Jefe}(R \bowtie_{R.Jefe = Empleado} JEFES)$

Obtener sólo los jefes y los jefes de los jefes de M.M.

$R \cup S$

Observación

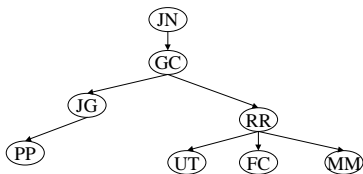
- Si se conoce la cantidad de "niveles" necesarios para obtener todos los jefes de M.M., entonces la consulta se puede resolver de una forma análoga a la vista anteriormente.
- En la forma original en que se planteó la consulta, no es posible resolverla sólo con álgebra relacional.
- La consulta es una

Clausura Transitiva

Clausura Transitiva

Las clausuras transitivas son consultas que típicamente involucran algún tipo de recorrido en un grafo.

El grafo representado en la tabla JEFES es el siguiente



Un Algoritmo

Solución = Álgebra + Lenguaje de Programación

Una solución posible es la siguiente:

r,q: Tabla;

q:= JEFES;

$r := \sigma_{Empleado = "M.M."}(JEFES);$

While haya nuevos resultados (r)

$r := (r \bowtie_{r.Empleado = q.jefe} q) \cup r$

EndWhile

Conclusiones

El álgebra relacional tiene un poder de expresión limitado ya que no permite escribir consultas que impliquen recursiones.

Escribir programas en lenguajes tradicionales (Cobol,C,etc.) que incluyan instrucciones en álgebra

Proveer extensiones al álgebra. Para eso se incluyen operadores recursivos en los lenguajes de consulta.