

FICHEROS Y BASES DE DATOS (E44)

3º INGENIERÍA EN INFORMÁTICA

Tema 10.

Álgebra Relacional

- 1.- Introducción.
- 2.- Una Sintaxis para el Álgebra Relacional.
- 3.- Asignación Relacional.
- 4.- Operaciones Tradicionales sobre Conjuntos.
- 5.- Operaciones Relacionales Especiales.
- 6.- ¿Para qué sirve el Álgebra?.
- 7.- Operaciones Adicionales.

(Capítulo 13 del Date)

INTRODUCCIÓN

Manipulación de Datos

- La manipulación de los datos en el modelo relacional se puede abordar de dos modos.
- Mediante el Álgebra Relacional se suministran los operadores que permiten construir una relación que contiene la información buscada.
- El Cálculo Relacional define la notación que permite describir las propiedades que deben cumplir las tuplas de la relación resultante.
- Además, en ambos casos resulta necesario añadir una Operación de Asignación que permite dar nombre a un resultado intermedio.

Panorama General del Álgebra

- El Álgebra Relacional se compone de un conjunto de operadores de alto nivel que operan sobre una o dos relaciones, y dan como resultado una relación.
- En la definición del modelo relacional, Codd definió ocho operadores:
 - Oper. Tradicionales sobre Conjuntos: la Unión, la Intersección, la Diferencia, y el Producto Cartesiano.
 - Oper. Relacionales Especiales: la Reunión, la Restricción, la Proyección, y la División.
- Con posterioridad, otros autores han definido otros operadores algebraicos que se han ido incorporando al modelo relacional original.

INTRODUCCIÓN

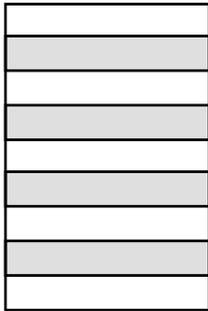
Operadores Originales

- Restricción (Select). Operador unario que genera una relación con las tuplas de la relación operando que cumplen la condición.
- Proyección (Project). Operador unario que genera una relación con los atributos seleccionados de la relación operando.
- Producto (Product). Operador binario que genera una relación cuyas tuplas son todas las posibles combinaciones de las tuplas de las dos relaciones operando.
- Unión (Union). Operador binario que genera una relación compuesta por las tuplas que están en alguna de las relaciones operando.
- Intersección (Intersect). Operador binario que genera una relación que contiene las tuplas que están en las dos relaciones operando.
- Diferencia (Minus). Operador binario que genera una relación en la que aparecen las tuplas que se encuentran en la primera relación y no se encuentran en la segunda.
- Reunión (Join). Operador binario que genera una relación que contiene las combinaciones de las tuplas de ambas relaciones que cumplen una determinada condición.
- División (Divide). Operador binario que genera una relación que contiene todos los valores de los atributos de la primera relación que concuerdan, en el resto de atributos, con todos los valores de la segunda relación.

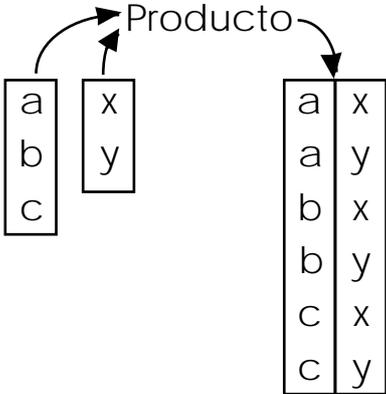
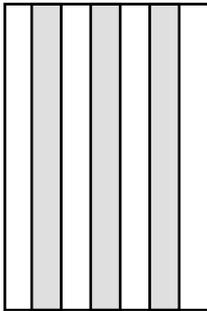
INTRODUCCIÓN

Representación de los Operadores Originales

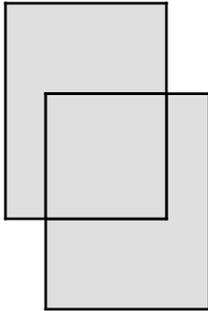
Restricción



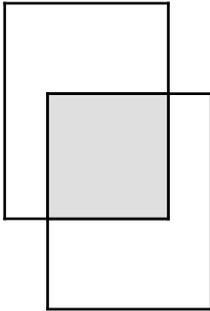
Proyección



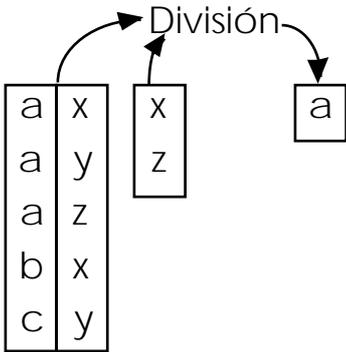
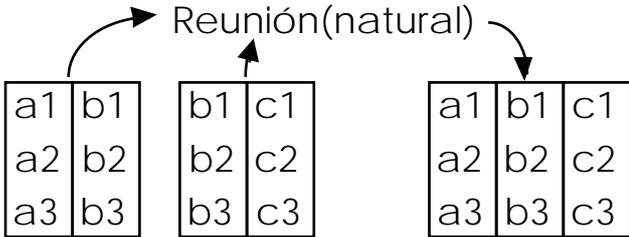
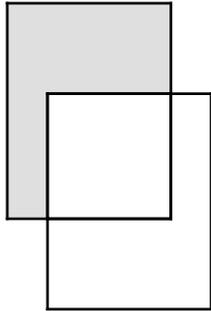
Unión



Intersección



Diferencia



INTRODUCCIÓN

Comentarios sobre el Álgebra

- El operador SELECT no corresponde con la instrucción SELECT de SQL.
- Realmente, ésta incluye la funcionalidad de todos los anteriores operadores, e incluso alguno más.

Propiedades del Álgebra

- La aplicación de un operador sobre una o dos relaciones da como resultado otra relación.
- Es decir, las relaciones son Cerradas respecto de este Álgebra,
- Por esta razón, se pueden definir Expresiones Relacionales Anidadas, donde el resultado de una operación es el operando de otra.
- Desde el Punto de Vista Conceptual, cada operación genera una relación resultado, que puede ser un resultado intermedio o el resultado de la consulta.
- Realmente no resulta necesario generar cada resultado intermedio para obtener la relación solución.
- Dada una expresión, el SGBD debe decidir que resultado intermedio debe generar.
- En algún caso, puede generar expresiones equivalentes que permitan un procesamiento más eficiente.

SINTAXIS DEL ÁLGEBRA

Cabecera de las Relaciones Resultado

- Hasta el momento se ha comentado el modo en el que actúan los operadores, generando el contenido de la relación resultado.
- Pero también resulta necesario definir cual va a ser la cabecera de dicha relación.
- Si el operador trabaja sobre una relación, la cabecera de la relación resultado es un subconjunto de los atributos de la relación operando.
- Si el operador trabaja sobre dos relaciones, la cabecera de la relación resultado es un subconjunto de la unión de los atributos de las relaciones operando.
- En este segundo caso, puede ocurrir que en las relaciones operando aparezcan atributos con el mismo nombre, de modo que en la unión de las cabecera sólo apareciera uno de ellos.
- Este hecho provocaría varios problemas:
 - Elegir el atributo que aparecería en la unión.
 - En ningún caso se podrían manejar ambos atributos.
- Para evitar estos problemas se añade un nuevo operador, Renombrar (Rename), operador unario que permite modificar el nombre de los atributos de una relación.
- Así, se resolvería el problema mediante la modificación del nombre de uno de los atributos en conflicto, .

SINTAXIS DEL ÁLGEBRA

Gramática BNF

expresión

::= expresión_de_una_relación |
expresión_de_dos_relaciones

expresión_de_una_relación

::= renombrado | restricción | proyección

renombrado

::= término RENAME atributo AS atributo

término

::= relación | (expresión)

restricción

::= término WHERE comparación

proyección

::= término |
término [lista_con_comas_de_atributos]

lista_con_comas_de_atributos

::= atributo |
atributo , lista_con_comas_de_atributos

expresión_de_dos_relaciones

::= proyección operación_binaria expresión

operación_binaria

::= UNION | INTERSECT | MINUS |
TIMES | JOIN | DIVIDEBY

SINTAXIS DEL ÁLGEBRA

Comentarios sobre la Gramática

- Los paréntesis y los corchetes se definen como elementos del lenguaje, sin tener ningún significado adicional.
- Únicamente el carácter "|" marca la separación de las posibles interpretaciones de un símbolo.
- Las categorías "relación" y "atributo" son los Identificadores, es decir, son una categoría terminal de la gramática, y representan el nombre de una relación y de un atributo.
- La categoría comparación representa una operación escalar sencilla de comparación, cuyos operandos son atributos o literales,

comparación

::= escalar comparador escalar

comparador

::= < | <= | = | <> | >= | >

escalar

::= atributo | literal

- El parentizado resulta muy importante en la definición del lenguaje, para asegurar la correcta evaluación de las operaciones.
- La inclusión de ciertas simplificaciones va a permitir la reducción del número total de paréntesis.

SINTAXIS DEL ÁLGEBRA

Simplificaciones de la Gramática BNF

- Cuando se desean renombrar varios atributos de una relación de modo simultáneo sería interesante realizar el renombrado de una única vez.
- La gramática tal y como está definida sólo permite renombrar un atributo cada vez, por lo que sería necesario anidar varias operaciones RENAME.
- Para evitar el anidamiento se debería modificar la gramática del modo siguiente,

renombrado

::= termino RENAME

lista_con_comas_a_renombrar

lista_con_comas_a_renombrar

::= renombrar | renombrar ,

lista_con_comas_a_renombrar

renombrar

::= atributo AS atributo

- Esta simplificación se podría extender a otras operaciones.

Ejemplo de Renombrado Múltiple

```
( ( ( S RENAME SITUACION AS SSITUACION )
  RENAME S# AS SNUM )
  RENAME CIUDAD AS SCIUDAD )
```

```
( S RENAME SITUACION AS SSITUACION ,
  S# AS SNUM , CIUDAD AS SCIUDAD )
```

SINTAXIS DEL ÁLGEBRA

Extensiones de la Gramática BNF

- La gramática definida admite la inclusión de literales en las comparaciones.
- También sería interesante que incluyera la posibilidad de incluir literales de tuplas y de relaciones.
- Para ello se debe extender la gramática,

literal_de_relación

::= { serie_de_tuplas }

serie_de_tuplas

::= tupla | tupla , serie_de_tuplas

tupla

::= (pares_dominio_valor)

pares_dominio_valor

::= par_dominio_valor |
par_dominio_valor , pares_dominio_valor

par_dominio_valor

::= dominio : literal

Herencia de Claves Primarias

- La relación resultado está compuesta de una cabecera y un cuerpo, pero también debe de elegirse una clave primaria.
- Por lo tanto, se deben de definir unas Reglas de Herencia de Clave Primaria, que permitan definir la clave primaria de la relación resultado de cualquier operador.

ASIGNACIÓN RELACIONAL

Definición

- El objetivo de esta operación es "Recordar" alguno de los resultados intermedios de una expresión.
- De este modo se puede modificar el estado de la base de datos.
- Así, la inserción y el borrado de tuplas de una relación se podría realizar mediante la utilización de los operadores UNION y MINUS.

```
s := S UNION { ( S#           : 'S6'           ,
                SNOMBRE      : 'Beltrán'      ,
                SITUACION     : 50             ,
                CIUDAD        : 'Madrid'      ) } ;
```

```
sp := SP MINUS { ( S#           : 'S1'           ,
                  P#           : 'P1'           ,
                  CANT         : 300           ) } ;
```

- Pero la utilización de estos operadores no controla la aparición de errores, como la inserción de tuplas ya existentes o el borrado de tuplas inexistentes.
- Para evitar estos problemas se deberían definir operadores específicos para realizar las inserciones y borrados.
- Estos operadores serían semejantes a las definidos en SQL, INSERT y DELETE.

OPER. TRADIC. SOBRE CONJUNTOS

Unión, Intersección y Diferencia

- La definición matemática de las operaciones sobre conjuntos resulta muy sencilla, de modo que el conjunto resultante contiene:
 - Unión, los elementos diferentes de ambos conjuntos.
 - Intersección, los elementos que aparecen en ambos conjuntos.
 - Diferencia, los elementos que aparecen en el primer conjunto y no en el segundo.
- En dicha definición no se analiza de que tipo son los elementos de los diferentes conjuntos.
- Pero en el álgebra relacional se desea, que tanto los operandos como el resultado, sean relaciones.
- Por tanto todas las tuplas que aparecen en la relación resultado deben ser Homogéneas, para lo cual las relaciones operando deben de ser Compatibles:
 - El grado de las relaciones operando tiene que coincidir.
 - El dominio de sus atributos con el mismo nombre también debe de coincidir.
- Si existiera algún problema relacionado con el nombre de algún atributo, se podría resolver mediante la utilización del operador RENAME.
- Así, la relación resultado también sería compatible con las relaciones operando.

OPERACIONES TRADICIONALES SOBRE CONJUNTOS

Ejemplos de Unión, Intersección y Diferencia

A

S#	SNOMBRE	SITUACION	CIUDAD
S1	Salazar	20	Londres
S4	Corona	20	Londres

B

S#	SNOMBRE	SITUACION	CIUDAD
S1	Salazar	20	Londres
S2	Jaimés	10	París

A UNION B

S#	SNOMBRE	SITUACION	CIUDAD
S1	Salazar	20	Londres
S2	Jaimés	10	París
S4	Corona	20	Londres

A INTERSECT B

S#	SNOMBRE	SITUACION	CIUDAD
S1	Salazar	20	Londres

A MINUS B

S#	SNOMBRE	SITUACION	CIUDAD
S4	Corona	20	Londres

B MINUS A

S#	SNOMBRE	SITUACION	CIUDAD
S2	Jaimés	10	París

OPER. TRADIC. SOBRE CONJUNTOS

Producto Cartesiano Ampliado

- Desde el punto de vista matemático, el producto cartesiano de dos conjuntos es el conjunto de todos los Pares Ordenados de Elementos, donde el Primer Elemento pertenece al Primer Conjunto y el Segundo pertenece al Segundo.
- Aplicándola sobre relaciones, el Producto es el Conjunto de todos los Pares Ordenados de Tuplas, de modo que la primera Tupla pertenece a la Primera Relación y la Segunda pertenece a la Segunda.
- Nuestro objetivo es que todas las operaciones sobre relaciones generen una relación, por lo que resulta necesario modificar su definición.
- Para ello, cada par se interpreta como una tupla cuyos pares atributo-valor son la unión de los pares atributo-valor de las dos tuplas del par.
- Así, la cabecera de la relación sería la unión de la cabecera de las relaciones operando.
- Si aparece un mismo nombre de atributo en las dos relaciones, se debe utilizar el operador RENAME, para que ambas sean Compatibles.

Propiedades

- La unión, la intersección y el producto son Conmutativas, aunque no la diferencia.
- Lo mismo ocurre con la Asociatividad.

OPER. TRADIC. SOBRE CONJUNTOS

Ejemplo de Producto

A	S#	B	P#	A TIMES B	S#	P#
	S1		P1		S1	P1
	S2		P2		S1	P2
	S3		P3		S1	P3
	S4		P4		S1	P4
	S5		P5		S1	P5
			P6		S1	P6
					S2	P1
					S2	P2
					S2	P3
					S2	P4
					S2	P5
					S2	P6
					S3	P1
					S3	P2
					S3	P3
					S3	P4
					S3	P5
					S3	P6
					S4	P1
					S4	P2
					S4	P3
					S4	P4
					S4	P5
					S4	P6
					S5	P1
					S5	P2
					S5	P3
					S5	P4
					S5	P5
					S5	P6

OPER. RELACIONALES ESPECIALES

Restricción

- La Restricción da como resultado una relación que contiene un subconjunto de las tuplas de la relación operando.
- Siendo A una relación, X e Y un literal o un atributo de la relación A, y theta un operador de comparación escalar simple, la expresión

$A \text{ WHERE } X \text{ theta } Y$, theta $\{<, <=, =, <>, >=, >\}$

da como resultado una relación que contiene las tuplas de A que cumplen que la condición (X theta Y) es verdadera.

- Una forma más correcta de nombrar al operador es como Restricción Theta, ya que se caracteriza por el operador theta.
- La definición únicamente permite evaluar una condición por cada operación restricción, de modo que se deben utilizar otros operadores para aplicar varias restricciones.
- Como el resultado de una condición es un resultado lógico, se podría extender la gramática para que incluya a los operadores lógicos.
- La siguiente tabla muestra la relación existente entre ambas soluciones.

A WHERE c1 AND c2	(A WHERE c1) INTERSECT (A WHERE c2)
A WHERE c1 OR c2	(A WHERE c1) UNION (A WHERE c2)
A WHERE NOT c	A MINUS (A WHERE c)

OPER. RELACIONALES ESPECIALES

Proyección

- La Proyección genera una relación en la que aparecen un subconjunto de los atributos de la relación operando.
- Siendo A una relación y X, Y, ... , Z atributos de A, la expresión,

$$A[X, Y, \dots, Z]$$

genera una relación donde,

- La cabecera son los atributos X, Y, ... , Z.
- El cuerpo se forma por el conjunto de tuplas (X:x, Y:y, ... , Z:z) tal que existe una tupla t en A que tiene el valor x en el atributo X, el valor y en el atributo Y, ... , y el valor z en el atributo Z.
- Como el resultado es una relación,
 - No pueden aparecer tuplas duplicadas, por lo que éstas se eliminan.
 - La lista de atributos no puede contener atributos replicados.
- Se define la Proyección Identidad como la proyección en la que no se especifica la lista de atributos.
- Cuando se especifica la lista de atributos pero no se especifica ningún atributo en ella, se obtiene la Proyección Nula.

OPER. RELACIONALES ESPECIALES

Ejemplos de Restricción y Proyección

S

S#	SNOMBRE	SITUACION	CIUDAD
S1	Salazar	20	Londres
S2	Jaimes	10	París
S3	Bernal	30	París
S4	Corona	20	Londres
S5	Aldana	30	Atenas

P

P#	PNOMBRE	COLOR	PESO	CIUDAD
P1	Tuerca	Rojo	12	Londres
P2	Perno	Verde	17	París
P3	Birlo	Azul	17	Roma
P4	Birlo	Rojo	14	Londres
P5	Leva	Azul	12	París
P6	Engrane	Rojo	19	Londres

S WHERE
CIUDAD = 'Londres'

S#	SNOMBRE	SITUACION	CIUDAD
S1	Salazar	20	Londres
S4	Corona	20	Londres

P WHERE PESO < 14

P#	PNOMBRE	COLOR	PESO	CIUDAD
P1	Tuerca	Rojo	12	Londres
P5	Leva	Azul	12	París

S[S#]

S#
S1
S2
S3
S4
S5

P[COLOR,CIUDAD]

COLOR	CIUDAD
Rojo	Londres
Verde	París
Azul	Roma
Azul	París

(S WHERE CIUDAD = 'Londres') [S#]

S#
S1
S4

OPER. RELACIONALES ESPECIALES

Reunión Natural

- La Reunión Natural permite unir las tuplas de dos relaciones, que cumplen que parte de sus atributos coinciden.
- Siendo A y B dos relaciones donde,
 - La cabecera de A es $(X_1, \dots, X_m, Y_1, \dots, Y_n)$.
 - La cabecera de B es $(Y_1, \dots, Y_n, Z_1, \dots, Z_p)$.

entonces se cumple que la expresión,

A JOIN B

genera una relación cuya cabecera es,

$(X_1, \dots, X_m, Y_1, \dots, Y_n, Z_1, \dots, Z_p) \quad (X, Y, Z)$

y el cuerpo se compone por el conjunto de tuplas $(X:x, Y:y, Z:z)$ tal que existe una tupla en A que tiene los valores x en X y los valores y en Y, y existe una tupla en B que tiene los valores y en Y y los valores z en Z.

- Una reunión natural también se puede definir mediante una combinación de operadores, tal y como se muestra a continuación,

$((A \text{ RENAME } Y \text{ AS } YA) \text{ TIMES } B) \text{ WHERE } YA = Y [X, Y, Z]$

- El análisis de la anterior expresión demuestra que la reunión natural cumple las propiedades de conmutatividad y asociatividad.
- Además, si el conjunto de atributos Y es vacío, la reunión natural es equivalente al producto.

OPER. RELACIONALES ESPECIALES

Ejemplo de Reunión Natural

S JOIN P

S#	SNOMBRE	SITUACION	CIUDAD	P#	PNOMBRE	COLOR	PESO
S1	Salazar	20	Londres	P1	Tuerca	Rojo	12
S1	Salazar	20	Londres	P4	Bitlo	Rojo	14
S1	Salazar	20	Londres	P6	Engrane	Rojo	19
S2	Jaimes	10	Paris	P2	Perno	Verde	17
S2	Jaimes	10	Paris	P5	Leva	Azul	12
S3	Bernal	30	Paris	P2	Perno	Verde	17
S3	Bernal	30	Paris	P5	Leva	Azul	12
S4	Corona	20	Londres	P1	Tuerca	Rojo	
S4	Corona	20	Londres	P4	Bitlo	Rojo	
S4	Corona	20	Londres	P6	Engrane	Rojo	

OPER. RELACIONALES ESPECIALES

Reunión Theta

- La Reunión Theta es una versión generalizada de la reunión natural, en donde las tuplas de las relaciones deben cumplir cierta condición para que su unión aparezca en el resultado.
- Siendo A y B dos relación, X un atributo de A, Y un atributo de B, y theta un operador de comparación escalar simple, la expresión

$(A \text{ TIMES } B) \text{ WHERE } X \text{ theta } Y$, theta $\{<, <=, =, >, >=, >\}$

genera una relación cuya cabecera coincide con la del producto de A y B, y el cuerpo es el subconjunto de tuplas de este producto que cumplen la condición $(X \text{ theta } Y)$.

- La principal diferencia con la reunión natural es la necesidad de que no existe ningún atributo de las relaciones cuyo nombre coincida, es decir, las relaciones deben ser compatibles respecto del producto.
- Dicha propiedad se hereda de su definición que se fundamenta en el operador producto, así como la conmutatividad y asociatividad.
- Además, tampoco se produce la eliminación de ninguno de los atributos ya que ninguno de ellos coincide.
- Cuando el comparador es el signo igual se obtiene una Equirreunión.
- Una reunión natural se puede observar como una equirreunión de la que se ha proyectado un subconjunto determinado de atributos.

OPER. RELACIONALES ESPECIALES

Ejemplo de Reunión Theta

(S RENAME CIUDAD AS SCIUDAD) TIMES (P RENAME CIUDAD AS PCIUDAD)
 WHERE SCIUDAD > PCIUDAD

S#	SNOMBRE	SITUACION	SCIUDAD	P#	PNOMBRE	COLOR	PESO	PCIUDAD
S2	Jaimés	10	París	P1	Tuerca	Rojo	12	Londres
S2	Jaimés	10	París	P4	Birlo	Rojo	14	Londres
S2	Jaimés	10	París	P6	Engrane	Rojo	19	Londres
S3	Bernal	30	París	P1	Tuerca	Rojo	12	Londres
S3	Bernal	30	París	P4	Birlo	Rojo	14	Londres
S3	Bernal	30	París	P6	Engrane	Rojo	19	Londres

OPER. RELACIONALES ESPECIALES

División

- La División permite seleccionar un subconjunto de los valores de las tuplas que cumplen que los atributos asociados se relacionan en la relación con un subconjunto de valores del resto de los atributos.
- Siendo A y B dos relaciones donde,
 - La cabecera de A es $(X_1, \dots, X_m, Y_1, \dots, Y_n)$.
 - La cabecera de B es (Y_1, \dots, Y_n) .

entonces se cumple que la expresión,

A DIVIDEBY B

genera una relación donde,

- La cabecera es (X_1, \dots, X_m) .
- El cuerpo se compone por el conjunto de tuplas (X:x) tal que existe una tupla en A que tiene los valores x en X y los valores y en Y, para cada una de las tuplas de B que tienen los valores y en Y.
- Como el resultado es una relación, se eliminan las tuplas duplicadas que pudieran aparecer.
- Este operador suele ser útil para consultas que incluyan en su enunciado la palabra "todos".
- Se puede definir mediante una combinación de operadores como sigue,

$A[X] \text{ MINUS } ((A[X] \text{ TIMES } B) \text{ MINUS } A)[X]$.

OPER. RELACIONALES ESPECIALES

Ejemplo de División

A	<table border="1"><thead><tr><th>S#</th><th>P#</th></tr></thead><tbody><tr><td>S1</td><td>P1</td></tr><tr><td>S1</td><td>P2</td></tr><tr><td>S1</td><td>P3</td></tr><tr><td>S1</td><td>P4</td></tr><tr><td>S1</td><td>P5</td></tr><tr><td>S1</td><td>P6</td></tr><tr><td>S2</td><td>P1</td></tr><tr><td>S2</td><td>P2</td></tr><tr><td>S3</td><td>P2</td></tr><tr><td>S4</td><td>P2</td></tr><tr><td>S4</td><td>P4</td></tr><tr><td>S4</td><td>P5</td></tr></tbody></table>	S#	P#	S1	P1	S1	P2	S1	P3	S1	P4	S1	P5	S1	P6	S2	P1	S2	P2	S3	P2	S4	P2	S4	P4	S4	P5	B	<table border="1"><thead><tr><th>P#</th></tr></thead><tbody><tr><td>P1</td></tr></tbody></table>	P#	P1	A DIVIDE BY B	<table border="1"><thead><tr><th>S#</th></tr></thead><tbody><tr><td>S1</td></tr><tr><td>S2</td></tr></tbody></table>	S#	S1	S2
S#	P#																																			
S1	P1																																			
S1	P2																																			
S1	P3																																			
S1	P4																																			
S1	P5																																			
S1	P6																																			
S2	P1																																			
S2	P2																																			
S3	P2																																			
S4	P2																																			
S4	P4																																			
S4	P5																																			
P#																																				
P1																																				
S#																																				
S1																																				
S2																																				
		C	<table border="1"><thead><tr><th>P#</th></tr></thead><tbody><tr><td>P2</td></tr><tr><td>P4</td></tr></tbody></table>	P#	P2	P4	A DIVIDE BY C	<table border="1"><thead><tr><th>S#</th></tr></thead><tbody><tr><td>S1</td></tr><tr><td>S4</td></tr></tbody></table>	S#	S1	S4																									
P#																																				
P2																																				
P4																																				
S#																																				
S1																																				
S4																																				
		D	<table border="1"><thead><tr><th>P#</th></tr></thead><tbody><tr><td>P1</td></tr><tr><td>P2</td></tr><tr><td>P3</td></tr><tr><td>P4</td></tr><tr><td>P5</td></tr><tr><td>P6</td></tr></tbody></table>	P#	P1	P2	P3	P4	P5	P6	A DIVIDE BY D	<table border="1"><thead><tr><th>S#</th></tr></thead><tbody><tr><td>S1</td></tr></tbody></table>	S#	S1																						
P#																																				
P1																																				
P2																																				
P3																																				
P4																																				
P5																																				
P6																																				
S#																																				
S1																																				

¿PARA QUÉ SIRVE EL ÁLGEBRA?

Consulta de las Datos

- A lo largo del capítulo se han definido una serie de operadores relacionales que son la base del álgebra relacional.
- Pero no todos ellos son realmente primitivas, ya que algunos de estos operadores pueden definirse a partir de los otros operadores:

- La intersección mediante la utilización de la diferencia,

$$A \text{ INTERSECT } B \quad A \text{ MINUS } (A \text{ MINUS } B).$$

- La reunión natural a partir de la combinación de los operadores de renombrado, producto, restricción y proyección,

$$A \text{ JOIN } B \quad , \quad A \{X, Y\} \quad , \quad B \{Y, Z\}$$

$$(((A \text{ RENAME } Y \text{ AS } YA) \text{ TIMES } B) \text{ WHERE } YA = Y)[X, Y, Z].$$

- La división mediante la utilización de la restricción, el producto y la proyección,

$$A \text{ DIVIDEBY } B \quad , \quad A \{X, Y\} \quad , \quad B \{Y\}$$

$$A[X] \text{ MINUS } ((A[X] \text{ TIMES } B) \text{ MINUS } A)[X].$$

- A pesar de esto, dada la utilidad de estos tres operadores, también se incluyen dentro de los operadores básicos del álgebra relacional.
- Por todo lo dicho parece que el único objetivo del álgebra es la Consulta de Datos, pero esta impresión no es correcta.

¿PARA QUÉ SIRVE EL ÁLGEBRA?

Utilidad del Álgebra

- El objetivo fundamental del álgebra es ayudar a Escribir Expresiones, como las siguientes:
 - Definir el Alcance de una Consulta.
 - Definir el Alcance de una Recuperación.
 - Definir el Alcance de una Actualización.
 - Definir los Datos de una Vista.
 - Definir los Datos de una Instantánea.
 - Definir Privilegios de Acceso.
 - Definir Requerimientos de Estabilidad.
 - Definir Restricciones de Estabilidad.
- De un modo formal, una expresión se define como una Representación Simbólica de Alto Nivel de la Intención del Usuario.
- Por ello, se pueden aplicar sobre ellas reglas de transformación, de gran utilidad en el proceso de optimización,

$$((S \text{ JOIN } SP) \text{ WHERE } P\# = 'P2') [SNOMBRE]$$
$$(S \text{ JOIN } (SP \text{ WHERE } P\# = 'P2')) [SNOMBRE].$$

- El álgebra también se utiliza como Patrón de Referencia para conocer la capacidad expresiva de un lenguaje.
- Así, un lenguaje es Relacionalmente Completo si puede expresar cualquier relación definida mediante una expresión del álgebra.

OPERACIONES ADICIONALES

Introducción

- En los últimos tiempos se han definido una serie de operadores con el objeto de ampliar el conjunto de operadores relacionales original.
- El objetivo de estos operadores es corregir alguna de las defectos del álgebra relacional.
- El álgebra relacional definida a través de los operadores originales, no incluye capacidad de cálculo.
- El operador Ampliación opera sobre escalares y/o los valores de los atributos en las tuplas, y el resultado se almacena en un nuevo atributo.
- El operador Resumen agrupa las tuplas de una relación mediante un criterio determinado, y opera sobre los valores de un atributo, dando lugar a un resultado para cada grupo.
- También se ha extendido alguno de los operadores clásicos.
- La División Generalizada elimina la restricción de la división en la que la cabecera de la relación divisor debía ser un subconjunto de la cabecera de la relación dividendo.
- La Reunión Externa incluye en el resultado las tuplas de las relaciones que no cumplen la condición, poniendo a valor nulo los atributos de la otra relación.
- También existen Versiones Externas de la Unión Intersección y Diferencia, en las que no se requiere la compatibilidad de las relaciones.

OPERACIONES ADICIONALES

Ampliación

- Ampliación (Extend). Operador unario que genera una relación con un atributo más que la relación operando en la que se incluye el resultado de una operación entre escalares y/o los valores de los atributos de cada tupla.

- La sintaxis de este operador es la siguiente,

EXTEND término ADD cálculo AS atributo

donde *cálculo* es cualquier operación entre escalares y/o valores de atributos de *término*, y *atributo* no coincide con ningún atributo de *término*.

- Un ejemplo podría ser el siguiente,

EXTEND P ADD (PESO * 454) AS PESOGRS .

- Como el resultado del operador es una relación podría aparecer como operando en otro operador, como sigue,

(EXTEND P ADD (PESO * 454) AS PESOGRS)
WHERE PESOGRS > 10000

- Para evitar el anidamiento de operadores, cuando se desea realizar varias ampliaciones sobre la misma relación, se podría modificar la sintaxis anterior como sigue,

EXTEND término ADD cálcl1 AS atrib1 , cálcl2 AS atrib2

- Un ejemplo de esta sintaxis es la siguiente,

(EXTEND P ADD 'Peso en gramos' AS EXPLICACIÓN,
(PESO * 454) AS PESOGRS)

OPERACIONES ADICIONALES

Gramática BNF de Ampliación

ampliación

::= EXTEND término ADD cálculo AS atributo

cálculo

::= atributo | literal |
función (lista_de_cálculos) |
(cálculo operador cálculo)

lista_de_cálculos

::= cálculo | cálculo , lista_de_cálculos

operador

::= + | - | * | / | \

función

::= sqr | sqrt | log | ln | exp

ampliación_multiple

::= EXTEND término ADD pares_cálculos

pares_cálculos

::= cálculo AS atributo |
cálculo AS atributo , (pares_cálculos)

OPERACIONES ADICIONALES

Resumen

- Resumen (Summarize). Operador unario que genera una relación en la que aparecen los atributos por los que se agrupa y un atributo adicional en el que aparece el resultado de aplicar la operación seleccionada sobre los valores de un atributo en cada grupo.
- La sintaxis de este operador es la siguiente,

```
SUMMARIZE término GROUPBY (lista_con_comas_de_atributos)  
      ADD cálculo_de_agregados AS atributo
```

donde *cálculo_de_agregados* es una función definida sobre un conjunto de valores, y que se aplica a uno de los atributos de *término*.

- La *lista_con_comas_de_atributos* puede ser una lista vacía en cuyo caso la función se aplica sobre todas las tuplas de *término*.
- El operador SUMMARIZE aplicado sobre una relación vacía genera una relación vacía.
- Este funcionamiento es erróneo, ya que COUNT y SUM debería dar cero, mientras que AVG, MAX y MIN debería dar un resultado indefinido.
- En SQL el funcionamiento es bastante extraño, generando resultados incorrectos.
- Para poder aplicar diversas funciones sobre un mismo agrupamiento se debe modificar la sintaxis como sigue,

```
SUMMARIZE término GROUPBY (lista_con_comas_de_atributos)  
      ADD cálculo_de_agreg1 AS atrib1 ,  
      cálculo_de_agreg2 AS atrib2 .
```

OPERACIONES ADICIONALES

Ejemplos de Resumen

```
SUMMARIZE SP GROUPBY ( P# ) ADD SUM ( CANT ) AS CANTTOTAL )
```

```
SUMMARIZE SP GROUPBY ( ) ADD SUM ( CANT ) AS GRANTOTAL )
```

```
SUMMARIZE SP GROUPBY ( P# ) ADD SUM ( CANT ) AS CANTTOTAL ),  
AVG ( CANT ) AS CANTMEDIO )
```

```
( ( SUMMARIZE ( P WHERE COLOR = 'Rojo' ) GROUPBY ( CIUDAD )  
ADD COUNT AS N ) WHERE N > 5 ) [CIUDAD]
```

```
( ( SUMMARIZE ( P WHERE COLOR = 'Rojo' ) GROUPBY ( CIUDAD )  
ADD COUNT AS N ) WHERE N < 5 ) [CIUDAD]  
UNION ( P [CIUDAD] MINUS ( P WHERE COLOR = 'Rojo' ) [CIUDAD] )
```

Gramática BNF de Resumen

resumen

```
::= SUMMARIZE término GROUPBY  
lista_con_comas_de_atributos ADD  
cálculo_de_agregados AS atributo
```

cálculo_de_agregados

```
::= operación_de_agregados ( atributo )
```

operación_de_agregados

```
::= SUM | COUNT | AVG | MAX | MIN
```

resumen_múltiple

```
::= SUMMARIZE término GROUPBY  
lista_con_comas_de_atributos ADD  
pares_resumen
```

pares_resumen

```
::= cálculo_de_agregados AS atributo |  
cálculo_de_agregados AS atributo ,  
pares_resumen
```

OPERACIONES ADICIONALES

División Generalizada

- La División Generalizada (Generalized Divide) es una extensión del operador división clásico que se puede aplicar sobre cualquier pareja de relaciones.
- Siendo A y B dos relaciones donde,
 - La cabecera de A es $(X_1, \dots, X_m, Y_1, \dots, Y_n)$.
 - La cabecera de B es $(Y_1, \dots, Y_n, Z_1, \dots, Z_p)$.

entonces se cumple que la expresión,

A DIVIDEBY B

genera una relación donde,

- La cabecera es $(X_1, \dots, X_m, Z_1, \dots, Z_p)$.
- El cuerpo se compone por el conjunto de tuplas $(X:x, Z:z)$ tal que existe una tupla en A que tiene los valores x en X y los valores y en Y, para todas las tuplas de B que tienen los valores y en Y y los valores z en Z.
- En función de la cardinalidad de los conjuntos de atributos de X, Y y Z se obtiene,
 - Si Z es vacío, la división clásica entre A y B.
 - Si X es vacío, la división clásica entre B y A.
 - Si Y es vacío, el producto de A y B.
- Esta operación se puede definir como,

$(A \text{ JOIN } B)[X,Z] \text{ MINUS } ((A[X] \text{ TIMES } B) \text{ MINUS } (A \text{ JOIN } B))[X,Z]$.

OPERACIONES ADICIONALES

Ejemplos de División Generalizada

A	<table border="1"><thead><tr><th>S#</th><th>P#</th></tr></thead><tbody><tr><td>S1</td><td>P1</td></tr><tr><td>S1</td><td>P2</td></tr><tr><td>S1</td><td>P3</td></tr><tr><td>S2</td><td>P1</td></tr><tr><td>S2</td><td>P2</td></tr><tr><td>S3</td><td>P2</td></tr><tr><td>S4</td><td>P2</td></tr><tr><td>S4</td><td>P3</td></tr><tr><td>S4</td><td>P4</td></tr></tbody></table>	S#	P#	S1	P1	S1	P2	S1	P3	S2	P1	S2	P2	S3	P2	S4	P2	S4	P3	S4	P4
S#	P#																				
S1	P1																				
S1	P2																				
S1	P3																				
S2	P1																				
S2	P2																				
S3	P2																				
S4	P2																				
S4	P3																				
S4	P4																				

B	<table border="1"><thead><tr><th>P#</th><th>J#</th></tr></thead><tbody><tr><td>P1</td><td>J1</td></tr><tr><td>P2</td><td>J1</td></tr><tr><td>P3</td><td>J1</td></tr><tr><td>P2</td><td>J2</td></tr><tr><td>P3</td><td>J2</td></tr><tr><td>P1</td><td>J3</td></tr><tr><td>P2</td><td>J3</td></tr><tr><td>P4</td><td>J3</td></tr><tr><td>P4</td><td>J4</td></tr></tbody></table>	P#	J#	P1	J1	P2	J1	P3	J1	P2	J2	P3	J2	P1	J3	P2	J3	P4	J3	P4	J4
P#	J#																				
P1	J1																				
P2	J1																				
P3	J1																				
P2	J2																				
P3	J2																				
P1	J3																				
P2	J3																				
P4	J3																				
P4	J4																				

A DIVIDEBY B	<table border="1"><thead><tr><th>S#</th><th>J#</th></tr></thead><tbody><tr><td>S1</td><td>J1</td></tr><tr><td>S1</td><td>J2</td></tr><tr><td>S4</td><td>J2</td></tr><tr><td>S4</td><td>J4</td></tr></tbody></table>	S#	J#	S1	J1	S1	J2	S4	J2	S4	J4
S#	J#										
S1	J1										
S1	J2										
S4	J2										
S4	J4										

A	<table border="1"><thead><tr><th>S#</th><th>P#</th></tr></thead><tbody><tr><td>S1</td><td>P1</td></tr><tr><td>S1</td><td>P2</td></tr><tr><td>S1</td><td>P3</td></tr><tr><td>S2</td><td>P1</td></tr><tr><td>S2</td><td>P2</td></tr><tr><td>S3</td><td>P2</td></tr><tr><td>S4</td><td>P2</td></tr><tr><td>S4</td><td>P3</td></tr><tr><td>S4</td><td>P4</td></tr></tbody></table>	S#	P#	S1	P1	S1	P2	S1	P3	S2	P1	S2	P2	S3	P2	S4	P2	S4	P3	S4	P4
S#	P#																				
S1	P1																				
S1	P2																				
S1	P3																				
S2	P1																				
S2	P2																				
S3	P2																				
S4	P2																				
S4	P3																				
S4	P4																				

B	<table border="1"><thead><tr><th>P#</th></tr></thead><tbody><tr><td>P1</td></tr><tr><td>P2</td></tr><tr><td>P3</td></tr></tbody></table>	P#	P1	P2	P3
P#					
P1					
P2					
P3					

A DIVIDEBY B	<table border="1"><thead><tr><th>S#</th></tr></thead><tbody><tr><td>S1</td></tr></tbody></table>	S#	S1
S#			
S1			

A	<table border="1"><thead><tr><th>P#</th></tr></thead><tbody><tr><td>P1</td></tr><tr><td>P2</td></tr><tr><td>P3</td></tr></tbody></table>	P#	P1	P2	P3
P#					
P1					
P2					
P3					

B	<table border="1"><thead><tr><th>P#</th><th>J#</th></tr></thead><tbody><tr><td>P1</td><td>J1</td></tr><tr><td>P2</td><td>J1</td></tr><tr><td>P3</td><td>J1</td></tr><tr><td>P2</td><td>J2</td></tr><tr><td>P3</td><td>J2</td></tr><tr><td>P1</td><td>J3</td></tr><tr><td>P2</td><td>J3</td></tr><tr><td>P4</td><td>J3</td></tr><tr><td>P4</td><td>J4</td></tr></tbody></table>	P#	J#	P1	J1	P2	J1	P3	J1	P2	J2	P3	J2	P1	J3	P2	J3	P4	J3	P4	J4
P#	J#																				
P1	J1																				
P2	J1																				
P3	J1																				
P2	J2																				
P3	J2																				
P1	J3																				
P2	J3																				
P4	J3																				
P4	J4																				

A DIVIDEBY B	<table border="1"><thead><tr><th>J#</th></tr></thead><tbody><tr><td>J1</td></tr></tbody></table>	J#	J1
J#			
J1			

OPERACIONES ADICIONALES

Ejemplos de División Generalizada

$(A \text{ JOIN } B)[S\#.J\#] \text{ MINUS } ((A[S\#] \text{ TIMES } B) \text{ MINUS } (A \text{ JOIN } B))[S\#.J\#]$

S#	P#	J#
S1	P1	J1
S1	P1	J3
S1	P2	J1
S1	P2	J2
S1	P2	J3
S1	P3	J1
S1	P3	J2
S2	P1	J1
S2	P1	J3
S2	P2	J1
S2	P2	J2
S2	P2	J3
S3	P2	J1
S3	P2	J2
S3	P2	J3
S4	P2	J1
S4	P2	J2
S4	P2	J3
S4	P3	J1
S4	P3	J2
S4	P4	J3
S4	P4	J4

S#	J#
S1	J1
S1	J2
S1	J3
S2	J1
S2	J2
S2	J3
S3	J1
S3	J2
S3	J3
S4	J1
S4	J2
S4	J3
S4	J4

S#	J#	P#
S1	J1	(P1,P2,P3)
S1	J2	(P2,P3)
S1	J3	(P1,P2,P4)
S1	J4	(P4)
S2	J1	(P1,P2,P3)
S2	J2	(P2,P3)
S2	J3	(P1,P2,P4)
S2	J4	(P4)
S3	J1	(P1,P2,P3)
S3	J2	(P2,P3)
S3	J3	(P1,P2,P4)
S3	J4	(P4)
S4	J1	(P1,P2,P3)
S4	J2	(P2,P3)
S4	J3	(P1,P2,P4)
S4	J4	(P4)

OPERACIONES ADICIONALES

Reunión Externa

- La Reunión Externa es una generalización de la reunión clásica, que genera al menos una tupla en la relación resultado para toda tupla de las relaciones operando.
- Si alguna tupla de una relación operando no aparece en la relación resultado de la reunión clásica, aparece en una tupla de la reunión externa con valores nulos en los atributos de la otra relación.
- Este operador no es una operación primitiva, aunque suele requerirse en muchos casos, por lo que Codd ha incluido la reunión externa theta como operador básico del álgebra.
- En pseudo-SQL se podría enunciar la reunión externa natural como sigue,

```
SELECT S.*, SP.P#, SP.CANT
FROM S, SP
WHERE S.S# = SP.S#
UNION
SELECT S.*, NULL, NULL
FROM S
WHERE NOT EXISTS
( SELECT *
FROM SP
WHERE SP.S# = S.S# )
```

OPERACIONES ADICIONALES

Problemas de la Reunión Externa

- Este operador presenta diferentes problemas, como los que siguen:
 - La equivalencia entre la reunión natural y la equireunión, la primera es una proyección de la segunda, no se mantiene en el caso de la reunión externa.
 - Por esta razón la inclusión de esta operación en los lenguajes existentes, basados en las proyecciones, suele ser complejo.
 - La aparición de nulos en el resultado puede asociar a valor desconocido o bien a que la propiedad no se aplica.
 - Los nulos pueden aparecer en las claves primarias de la relación resultado, de modo que el resultado no puede definirse como una relación base
- La reunión externa natural es teóricamente más importante que la reunión externa theta, pero Codd sólo incluye la segunda en el álgebra.
- Además dado que la primera no se puede derivar de la segunda, la excluye de modo global del álgebra.

OPERACIONES ADICIONALES

Otros Operadores

- El planteamiento para definir la Unión Externa, la Intersección Externa y la Diferencia Externa, es común a todas ellas,
 - Se añaden los atributos no comunes a cada una de las relaciones operando, de modo que se convierten en compatibles.
 - En cada tupla se asigna el valor nulo para cada uno de los atributos adicionales.
 - Posteriormente se implementa la operación.
- Como en la reunión externa, la unión externa presenta serios problemas de interpretación de los nulos.
- La intersección externa siempre genera una relación vacía, excepto en el caso de que las relaciones operando sean compatibles.
- Si las relaciones operando no son compatibles, la diferencia externa siempre genera como resultado la primera relación operando.
- Por todo lo anterior, sólo la unión externa se incluye en el modelo relacional.
- Existen otros conjuntos de generalizaciones de los operadores clásicos, que se denominan de un modo global como Operaciones Quizás.
- Se asocian a una nueva lógica de tres valores: Verdadero, Falso y Desconocido.
- Desconocido aparece cuando se comparan dos valores nulos.