

Tema 3: Dinámica Relacional

- Operar con la base de datos. Las operaciones sobre la BD producen nuevos estados en la BD.
- Los lenguajes relacionales son de especificación. Se distinguen:
 - Procedimentales o Algebraicos:** Especifica los pasos, hasta llegar a la meta, mediante operaciones que sufre la BD.
 - No-procedimentales:** Indica la meta, pero no como la alcanzaremos, como serie el estado final de la BD.
- Los operandos en cualquier operación algebraica-relacional son relaciones, y el resultado es siempre una relación. No se modifican las originales, se crea otra nueva tabla.
- Operadores unarios AR Básica:** Recibe una relación y devuelve otra relación.
 - Selección:** Mismas columnas, pero solo las filas que cumplan la condición. Se designa con un sigma. grado original=grado final y cardinalidad original>=cardinalidad final.

$\sigma_{\text{Condición}}(\text{Relación})$ Ejem: $\sigma_{\text{autor} = \text{'Dumas'}}(\text{Libros})$

- Proyección:** Tiene menos columnas siempre, pero las mismas filas, se indican los atributos que formaran la nueva relación. grado original > grado final y cardinalidad original=cardinalidad fin.

$\pi_{\text{atrib1, atrib2, ...}}(\text{Relación})$ Ejem: $\pi_{\text{titulo, autor}}(\text{Libros})$ $\begin{matrix} \text{tit} & \text{autor} \\ \hline \end{matrix}$

- Renombrado:** Le doy una etiqueta a una operación, para ahorrar repetir la expresión.

$\rho_A(\sigma_{\text{autor} = \text{'Dumas'}}(\text{Libros}))$ ó $A \equiv \sigma_{\text{autor} = \text{'Dumas'}}(\text{Libros})$

- Operadores binarios AR Básica:** Para las 3 primeras las relaciones deben ser compatibles, mismo grado y definición de los atributos. Recibe dos relaciones compatibles y devuelve otra relación.

- Unión:** Tiene todas las filas que tenían ambos operandos. grado 1=grado 2=grado final y cardinalidad final <= cardinalidad1 + cardinalidad2

$\text{Relación 1} \cup \text{Relación 2}$ Deben tener las mismas columnas.

- Intersección:** Las filas comunes en ambas tablas. grado A=grado B= grado final y cardinalidad final <= cardinalidad del que menos filas tiene.

- $A \cap B = A - (A - B)$

$\text{Relación 1} \cap \text{Relación 2}$ Deben tener las mismas columnas.

- Diferencia:** Todas las filas que están en A pero que no estén también en B. Grados iguales y cardinalidad final <= cardinalidad de la relación de la izquierda.

$\text{Relación A} - \text{Relación B}$ Deben tener las mismas columnas.

- Producto cartesiano:** Tiene todos los atributos de A y B, y en contenido todas las filas de A concatenadas con las filas de B, todas las posibilidades de juntar ambas filas. grado final=grado A+grado B y cardinalidad final= cardinalidad A*cardinalidad B

$\text{Relación A} \times \text{Relación B}$ $\begin{matrix} \text{c} & \text{d} & \text{e} & \text{f} \\ \text{b}_1 & \text{b}_2 & \text{g}_1 & \text{g}_2 \\ \hline \end{matrix} \times \begin{matrix} \text{c} & \text{d} & \text{e} & \text{f} \\ \text{b}_3 & \text{b}_4 & \text{g}_3 & \text{g}_4 \\ \hline \end{matrix} = \begin{matrix} \text{c} & \text{d} & \text{e} & \text{f} \\ \text{b}_1 & \text{b}_3 & \text{g}_1 & \text{g}_3 \\ \text{b}_1 & \text{b}_4 & \text{g}_1 & \text{g}_4 \\ \text{b}_2 & \text{b}_3 & \text{g}_2 & \text{g}_3 \\ \text{b}_2 & \text{b}_4 & \text{g}_2 & \text{g}_4 \\ \hline \end{matrix}$

- Combinación:** Subconjunto del producto cartesiano que cumple una cierta condición que relaciona un atributo de la relación A con otro de la relación B. grado final=grado A+grado B y cardinalidad final<= cardinalidad A*cardinalidad B

$\text{Relación A} \theta_{\text{atrib A} \text{ ? } \text{atrib B}} \text{Relación B}$

- Combinación Natural:** Consiste en una combinación en la que la condición es la igualdad de dos atributos, de distinta relación. Solo se mantiene uno de los dos atributos en la resultado.

$\text{Relación A} *_{\text{atrib}} \text{Relación B}$ El atrib es común en ambas relaciones.

- Los **operadores derivados** pueden ser sustituidos por una secuencia de otras operaciones, sin embargo los **operadores primitivos** no se pueden sustituir por una expresión equivalente.
 - Primitivos: Selección, Proyección, Producto Cartesiano, Diferencia y Unión.
 - Derivados: Intersección, Combinación y Combinación Natural.
- Operadores de Álgebra Relacional extendida:**
 - Agrupación:** Se obtienen los diferentes valores del atributo de criterio. Se añadirán atributos con funciones de agregación que se añaden como proyecciones o selecciones, pueden ser: *Count(x) por cada repetición y luego cuenta las x!*
 - Count() conteo, Sum() sumatorio, Avg() media, Min() mínimo, Max() máximo. Todas esperan parámetros.

Proyección π Selección σ Criterio (Relación)
 - División:** Da lugar a las columnas de A que están concatenadas con los atributo/s indicados de la relación B. Y las filas se quedan las que tienen en la columna común los mismos valores. *Basicamente, te quedas con la columnas que no están indicadas en B, y las filas que contenían en esa columna los mismos valores que indica B*
 $A \div B$

$$A \div B \equiv \pi_{\text{conj}(A) - \text{conj}(B)} A - \pi_{\text{conj}(A) - \text{conj}(B)} ((\pi_{\text{conj}(A) - \text{conj}(B)} A \times B) - A)$$
 - Semi-Combinación:** Combinación natural, con un atributo que ambas relaciones tienen en común, en la que la relación final solo se queda los atributos de una de las relaciones, aparecerán solo las filas que tienen algún valor también en la otra relación.
 - Izquierda \bowtie *Relación A $\bowtie_{\text{atrib. común}}$ Relación B*
 - Derecha \bowtie
 - Anti-Combinación:** Es la opuesta a la semi-combinación, nos quedamos con las filas que no tienen ningún valor en común con la columna que comparten ambas. Nos podemos quedar con los atributos de la relación de la derecha o con los de la izquierda.

Izquierda ∇ Relación A $\nabla_{\text{atrib. común}}$ Relación B
Derecha ∇
Las que no coinciden los valores en ambas, y cogemos los de izq o dch.
 - Combinación externa:** Combinación natural en la que también aparecen las filas que no tienen pareja en la otra relación, como no aparecía en ella en la final los atributos que faltan toman valor null. Podemos quedarnos solo los atributos de la derecha, de la izquierda o con todos. Si nos quedamos con un lado aparecen todos las filas de ese lado añadidos los atributos de la segunda relación con su valor si lo tenía asociado al criterio, si no null.

Izquierda \Join Relación A $\Join_{\text{atrib. común}}$ Relación B
Derecha \Join
Ambos \Join
Los de ese lado con las columnas del otro y los valores asociados o null.
 - Orden:** Devuelve una lista ordenada en orden ascendente (menos a mas) o descendente según la definición de la operación y criterio, se pueden aplicar funciones de agregación:
 - first (el primer valor de la lista), last (el ultimo valor de la lista) y rank(valor) el de la posición valor.

Ascendente \Uparrow
Descendente \Downarrow
funciones \Uparrow atrib (Relación)
agregadas
- Operador de Álgebra Relacional completa:** Asignar y modificar.
 - Asignación:** Define operaciones de actualización sobre la base de datos. $Rel \leftarrow Operación$
 - Borrar filas:** Mediante la resta de una relación y una parte de ella.
 - Borrar relación:** Asignar vacío a la relación. $Rel \leftarrow \emptyset$
 - Añadir una fila:** Unión de la relación y una tupla con unos valores definidos.
 - Insert masivo:** Unión de una relación de con otra de la que hemos elegido las columnas que hacen que ambas sean compatibles, tengan los mismos atributos.
 - Actualizar datos:** Unión de la parte modificada con la relación sin esa parte.

◦ Dinámica del SQL: DML

▸ Modos de manipulación:

- Interactivo, mediante operaciones directas en SQL.
- SQL embebido, mediante un lenguaje anfitrión, como C o Java.
- Módulos, llamadas a procedimientos desde procesos externos.

▸ Operaciones de actualización:

- Inserción de tuplas: Insert
- Borrado de tuplas: Delete
- Modificación de tuplas: Update

▸ Operaciones de recuperación:

- Consulta o Query: Select

◦ Lenguaje de control: LCD

▸ Transacción: Operación atómica sobre la base de datos. Conjunto de instrucciones de actualización.

▸ Son:

- Commit [work], realizar, perpetra los cambios y son permanentes, si no se hace se perderán los datos.
- Rollback [work] [to [savepoint] <savepoint>], deshacer, copia los datos de la BD a mi segmento privado, por lo que se restaura lo original y se pierde progreso.
- Savepoint <savepoint>, para que el rollback solo borre hasta ese punto.

◦ Sintaxis de la QUERY: Todo se escribe en la misma sentencia, y lo que esta entre [] es opcional

▸ [WITH <símbolo> AS <subquery>, ...]

atrib no oblig

1) • Precláusula o renombrado, para representar una subquery con un solo símbolo, para que quede mas limpio y se entienda mejor. Se pueden encadenar, separando con , .

7) • SELECT [ALL|DISTINCT] <lista de selección>

- ALL, todas pudiéndose repetir, o DISTINCT, los valores únicos. Obligatorio estar seguido de las columnas que queremos consultar, * para todo el area de trabajo, atributos, pseudo-columnas (ROWNUM fila que ocupa esa posición y table. ROWID dirección física de la columna), constantes o variables ligadas, funciones aritméticas(+,-), de strings(||, substr), de codificación(case, nvl), de conversión (to_char,...), del sistema(sysdate, user), de agregación o compiladas(creadas por el user). Se puede hacer renombrado.

2) • FROM <cláusula de origen>

- Indica la tabla o combinación de ellas (mediante otro SELECT, X CROSS JOIN Y, X JOIN Y [ON condición], X JOIN Y USING columnas con el mismo nombre, UNIÓN o {LEFT|RIGHT|FULL} JOIN [USING ...| ON ...]) de la que obtener los datos. Si no se quiere ninguna tabla se pone DUAL, es una tabla fantasma.

3) • [WHERE <condición>]

- Selección, ponemos la condición y aparecerán las que la cumplan. Mirar diapos.

4) • [GROUP BY <expresión> [HAVING <condcn>]]

- Ordena según expresión, y en having están las condiciones colectivas.

5) • [{UNION|UNION ALL| MINUS| INTERSECT} <query>]

- Union sin repetición, union con repetición, resta, intersección y query con la que realizar la operación, para hacer operaciones de más términos hay que ir encadenando query's que elijan la misma operación.

6) • [ORDER BY <expresión> [ASC|DESC]] ;

- Lista ordenada en base a la operación, por defecto lo hace ascendente. Se acaba la sentencia con un ; .