

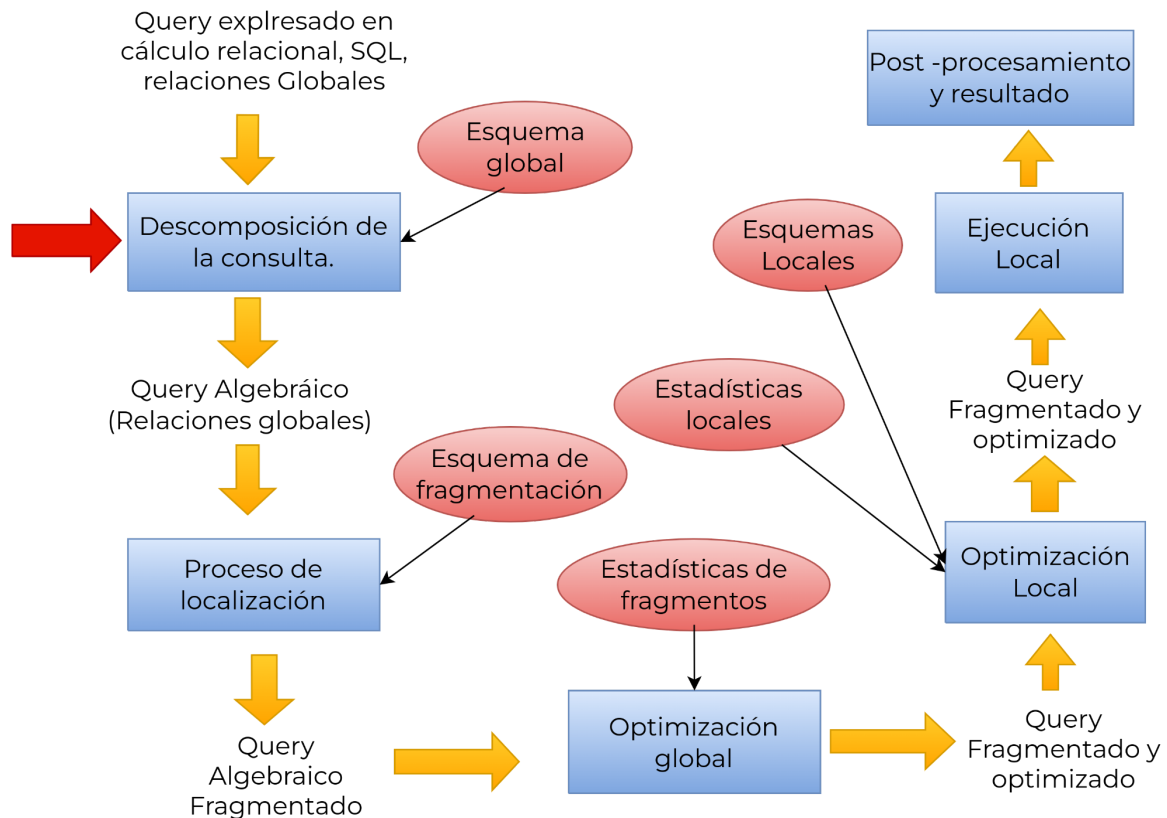


4. DESCOMPOSICIÓN Y LOCALIZACIÓN DE SENTENCIAS DISTRIBUIDAS

Índice

4. DESCOMPOSICIÓN Y LOCALIZACIÓN DE SENTENCIAS DISTRIBUIDAS	1
4.1. Descomposición de sentencias distribuidas	1
Forma Normal Conjuntiva:	2
Forma Normal disyuntiva:	3
Reglas de equivalencia:	3
Ejercicio en clase	3
Ejercicio en clase	4
Ejercicio en clase	5
Ejercicio en clase	5
Ejercicio en clase	6
Regla 1: Conmutatividad de operadores binarios	6
Regla 2: Asociatividad de operadores binarios	6
Regla 4: Selección conmutativa con proyección	7
Ejercicio en clase	8
4.2. Proceso de localización	8
Ejemplo	9
4.2.1. Reducción por fragmentación horizontal primaria	9
Ejercicio en clase	10
Reducción por join	10
Ejercicio en clase	11
4.2.2. Reducción por fragmentación derivada	11
Ejercicio en clase	11
4.2.3. Reducción por fragmentación vertical	12
Ejercicio en clase	12
4.2.4. Fragmentación híbrida	12
Ejercicio en clase	13

4.1. Descomposición de sentencias distribuidas



Tanto la entrada como la salida consideran consultas globales.

La salida es una consulta expresada en álgebra relacional semánticamente correcta.

Pasos para realizar la descomposición:

- Normalización.
- Análisis
- Eliminación de redundancia.
- Re-escritura.

4.1.1. Normalización

- Consiste en expresar una consulta en su forma normal que permita facilitar su posterior procesamiento, en especial predicados.
- Existen 2 formas normales:

Forma Normal Conjuntiva:

$$(P_{11} \vee P_{12} \vee \dots \vee P_{1n}) \wedge \dots \wedge (P_{m1} \vee P_{m2} \vee \dots \vee P_{mn})$$

Forma Normal disyuntiva:

$$(P_{11} \wedge P_{12} \wedge \dots \wedge P_{1n}) \vee \dots \vee (P_{m1} \wedge P_{m2} \wedge \dots \wedge P_{mn})$$

Reglas de equivalencia:

1. $p_1 \vee p_2 \Leftrightarrow p_2 \vee p_1$
2. $p_1 \wedge p_2 \Leftrightarrow p_2 \wedge p_1$
3. $p_1 \wedge (p_2 \wedge p_3) \Leftrightarrow (p_1 \wedge p_2) \wedge p_3$
4. $p_1 \vee (p_2 \vee p_3) \Leftrightarrow (p_1 \vee p_2) \vee p_3$
5. $p_1 \wedge (p_2 \vee p_3) \Leftrightarrow (p_1 \wedge p_2) \vee (p_1 \wedge p_3)$
6. $p_1 \vee (p_2 \wedge p_3) \Leftrightarrow (p_1 \vee p_2) \wedge (p_1 \vee p_3)$
7. $\neg(p_2 \wedge p_3) \Leftrightarrow \neg p_2 \vee \neg p_3$
8. $\neg(p_2 \vee p_3) \Leftrightarrow \neg p_2 \wedge \neg p_3$
9. $\neg(\neg p_1) \Leftrightarrow p_1$



Ejercicio en clase

Realizar las actividades del siguiente ejercicio. La respuesta no se incluye en estas notas, se sugiere agregar la respuesta en sus apuntes.

- A. Generar una sentencia SQL que muestre los nombres de los empleados del **proyecto_id=10** con duración igual a 12 o 24 meses.
- B. Expresar la sentencia en su Forma Normal Disyuntiva (FND).

4.1.2. Análisis

- Procesa una consulta para determinar si es correcta semánticamente.
- Una sentencia es semánticamente incorrecta si sus componentes no contribuyen para nada en la generación del resultado.
- Para detectar errores de semántica se construye la llamada “gráfica del query” o “gráfica de conexiones”.
 - Cada nodo representa el resultado de una operación la cual produce una “relación” que corresponde al resultado final de la consulta.
 - El conector que lleva al resultado representa a una operación de proyección.
 - Los demás conectores representan una operación de Join o de selección.
 - Existe otra gráfica llamada “Gráfica de Joins” la cual únicamente considera esta operación, empleada especialmente para la fase de optimización.
- La gráfica se emplea para detectar errores de semántica.

- La gráfica es incorrecta si se detecta un nodo sin conexión (se generan varias gráficas).
- Algunos sistemas lo consideran correcto asumiendo un producto cartesiano lo cual se debe evitar.



Ejercicio en clase

Realizar las actividades del siguiente ejercicio. La respuesta no se incluye en estas notas, se sugiere agregar la respuesta en sus apuntes.

Considerar el siguiente enunciado y su sentencia SQL:

Mostrar los nombres y roles de los empleados con puesto 'Lider' que han trabajado en el proyecto 'NoSQL' por más de 3 años:

```
select  e.nombre, ae.rol
from e, ae, pu, pr
where  pu.puesto_id = e.puesto_id
and e.emp_id = ae.emp_id
and  ae_proy_id = pr.proy_id
and  pu.nombre = 'lider'
and  pr.nombre = 'nosql'
and  ae.duracion > 3
```

A. Construir la gráfica de conexiones.

4.1.3. Eliminación de redundancia

- Permite eliminar predicados redundantes en una sentencia SQL.
- Para realizar estas eliminaciones se suelen aplicar las siguientes reglas básicas

1. $p \wedge p \Leftrightarrow p$
2. $p \vee p \Leftrightarrow p$
3. $p \wedge \text{true} \Leftrightarrow p$
4. $p \vee \text{false} \Leftrightarrow p$
5. $p \wedge \text{false} \Leftrightarrow \text{false}$
6. $p \vee \text{true} \Leftrightarrow \text{true}$
7. $p \wedge \neg p \Leftrightarrow \text{false}$
8. $p \vee \neg p \Leftrightarrow \text{true}$
9. $p_1 \wedge (p_1 \vee p_2) \Leftrightarrow p_1$
10. $p_1 \vee (p_1 \wedge p_2) \Leftrightarrow p_1$



Ejercicio en clase

Realizar las actividades del siguiente ejercicio. La respuesta no se incluye en estas notas, se sugiere agregar la respuesta en sus apuntes.

Considerar la siguiente sentencia SQL.

```
select nombre
from empleado e
where (
    nombre <> 'juan'
    and (nombre = 'juan' or nombre = 'paco')
    and nombre <> 'paco'
) or nombre = 'julio'
```

A. Aplicando las reglas básicas de eliminación de redundancia, generar una sentencia SQL equivalente.

4.1.4. Re-escritura

- Representa el último paso de la descomposición de una consulta.
- La sentencia se describe en álgebra relacional.
- Se emplea un “árbol de operadores” en el que sus hojas representan a relaciones (tablas) que existen en la BD.
 - Para construir el árbol se crea un nodo hoja por cada tabla que participa en la sentencia.
- Los demás nodos contienen relaciones intermedias.
- La secuencia de ejecución de operaciones inicia en las hojas hasta llegar a la raíz de donde se obtiene el resultado.
- El nodo raíz representa una operación de proyección o selección.



Ejercicio en clase

Realizar las actividades del siguiente ejercicio. La respuesta no se incluye en estas notas, se sugiere agregar la respuesta en sus apuntes.

- A. Generar una sentencia SQL que muestre los nombres de los empleados con nombre distinto a 'juan' que han trabajado en proyectos de 'migración' con duración de 12 a 24 meses.
- B. Construir una propuesta de árbol de operadores.

4.1.5. Reglas de transformación

- Permiten la construcción de diversos árboles de operadores.



- Permiten mejorar o simplificar el procesamiento de la consulta.

Considerar las siguientes 3 relaciones:

R con sus atributos $A = \{A_1, A_2, \dots, A_n\}$

S con sus atributos $B = \{B_1, B_2, \dots, B_n\}$

T con sus atributos $C = \{C_1, C_2, \dots, C_n\}$



Ejercicio en clase

Realizar las actividades del siguiente ejercicio. La respuesta no se incluye en estas notas, se sugiere agregar la respuesta en sus apuntes.

Para las reglas que se indican, reescribirlas empleando SQL.

Regla 1: Conmutatividad de operadores binarios

A. Producto cartesiano de 2 relaciones.

$$R \times S \Leftrightarrow S \times R$$

B. Join entre 2 relaciones

$$R \bowtie S \Leftrightarrow S \bowtie R$$

C. Aplica también para Unión, intersección, no para diferencia y semi-Join

Regla 2: Asociatividad de operadores binarios

A. El producto y el Join son asociativos.

$$(R \times S) \times T \Leftrightarrow R \times (S \times T)$$

$$(R \bowtie S) \bowtie T \Leftrightarrow R \bowtie (S \bowtie T)$$

Regla 3: Idempotencia de operadores unarios

$$\pi_A(\pi_A(R)) = \pi_A(R)$$

- Escribir el equivalente en SQL

Selección subsecuente aplicada a los atributos de la misma relación

$$\sigma_{p1(A1)}(\sigma_{p2(A2)}(R)) = \sigma_{p1(A1) \wedge p2(A2)}(R)$$

Notación: $\sigma_{p1(A1)}$ Selección de un predicado p1 aplicado a un atributo A1

- Escribir el equivalente en SQL

Regla 4: Selección conmutativa con proyección

Selección y proyección sobre la misma relación pueden ser conmutadas de la siguiente forma:

$$\pi_{A1, \dots, An}(\sigma_{p(Ap)}(R)) \Leftrightarrow \pi_{A1, \dots, An}(\sigma_{p(Ap)}(\pi_{A1, \dots, An, Ap}(R)))$$

- Escribir el equivalente en SQL

Si la operación de Proyección contiene otros atributos Ap, se requiere de otra proyección (lado derecho), de otra forma, la proyección del lado derecho ya no es requerida.

Regla 5: Selección conmutativa con operadores binarios

A. Selección y producto pueden ser conmutados.

$$\sigma_{p(Ai)}(R \bowtie S) \Leftrightarrow (\sigma_{p(Ai)}(R)) \bowtie S$$

- Escribir el equivalente en SQL

B. Selección y Join

$$\sigma_{p(Ai)}(R \bowtie_{p(Aj, Bk)} S) \Leftrightarrow (\sigma_{p(Ai)}(R)) \bowtie_{p(Aj, Bk)} S$$

- Escribir el equivalente en SQL

C. Selección y unión

$$\sigma_{p(Ai)}(R \cup T) \Leftrightarrow (\sigma_{p(Ai)}(R)) \cup \sigma_{p(Ai)}(T)$$

D. Aplica similar para selección y diferencia.

Modelo Relacional para los ejemplos de análisis.

Regla 6: Proyección conmutativa con operadores binarios

$$\pi_C(R \cup S) \Leftrightarrow \pi_{A'}(R) \cup \pi_{B'}(S)$$

$$\pi_C(R \bowtie S) \Leftrightarrow \pi_{A'}(R) \bowtie \pi_{B'}(S)$$

$$\pi_C(R \bowtie_{p(A_j, B_k)} S) \Leftrightarrow \pi_{A'}(R) \bowtie_{p(A_j, B_k)} \pi_{B'}(S)$$

$$C = A' \cup B'$$

$$A' \subseteq A$$

$$B' \subseteq B$$

La aplicación de estas 6 reglas puede generar múltiples árboles de operadores, por lo tanto, múltiples soluciones para ejecutar una consulta. Algunos más eficientes que otros.

Estas reglas pueden aplicarse en 4 formas principales:

1. Separación de operadores unarios para simplificar expresiones
2. Operaciones unarias sobre la misma relación se deben agrupar para realizar un solo acceso a disco.
3. Operadores unarios pueden ser combinados con operadores binarios con la finalidad de que operadores unarios como Selección y proyección se ejecuten primero.
4. Operadores binarios son ordenados.

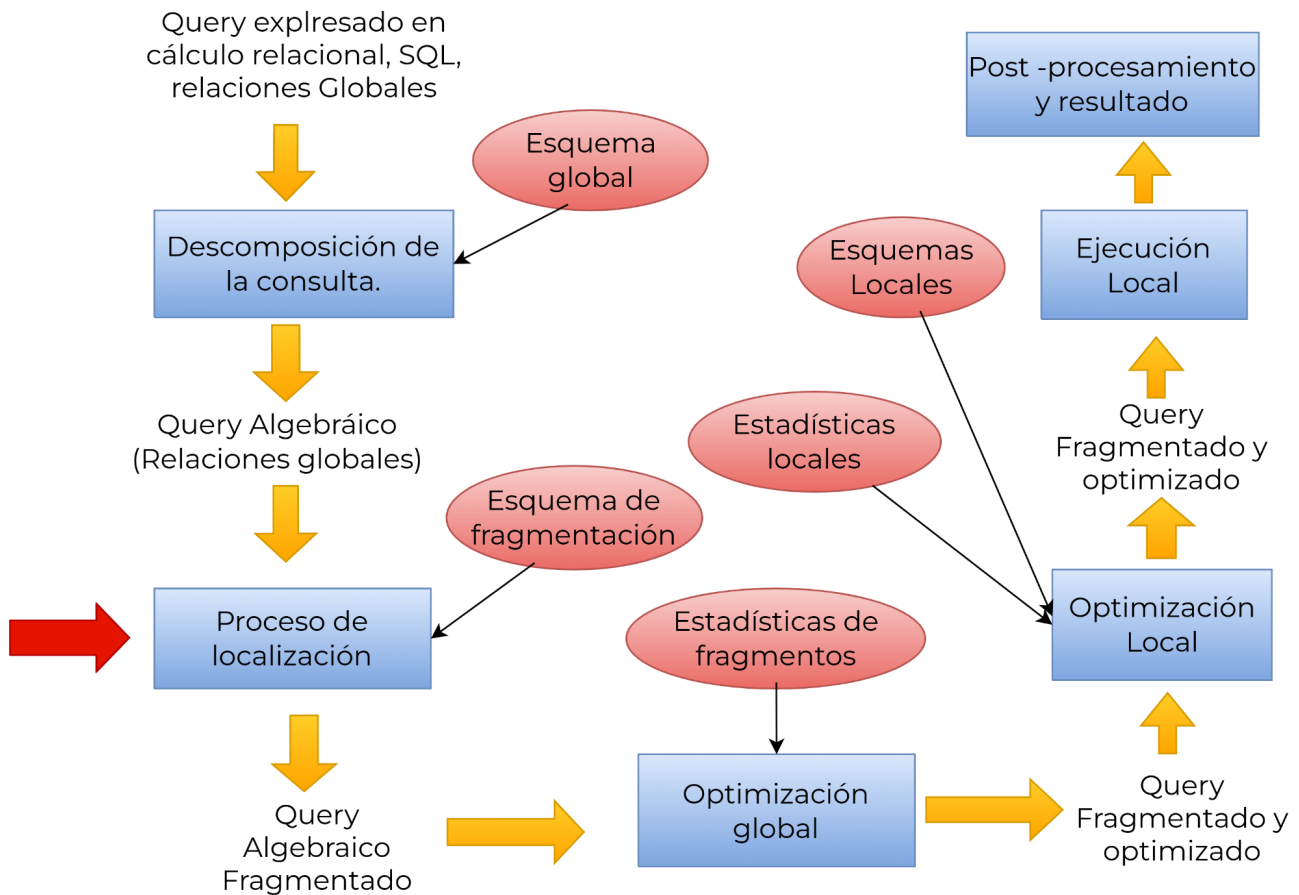


Ejercicio en clase

Realizar las actividades del siguiente ejercicio. La respuesta no se incluye en estas notas, se sugiere agregar la respuesta en sus apuntes.

Considerando el árbol obtenido en el ejercicio 4, aplicar las reglas de equivalencia y proponer una solución más eficiente.

4.2. Proceso de localización



- Hasta este punto no se ha considerado la distribución de los datos.
- En esta etapa se realiza la transformación de la consulta algebraica global en una consulta algebraica expresada en términos de fragmentos haciendo uso del esquema de fragmentación.
- La forma más simple es sustituyendo a cada relación global por la expresión de reconstrucción la cual incluye la definición de los fragmentos de cada tabla. Al resultado de este proceso se le conoce como Programa de localización.

Ejemplo

$$E \rightarrow E_1 \cup E_2 \cup E_3$$

- Esta técnica funciona, pero no es óptima.
- Existen diversas técnicas de localización para optimizar las cuales se aplican para cada tipo de fragmentación.

4.2.1. Reducción por fragmentación horizontal primaria

Existen 2 técnicas para realizar reducción:

- Reducción con selección
- Reducción con join

Reducción con selección.

- Se aplica la propiedad de distributividad con uniones eliminando expresiones que generan conjuntos vacíos.

$$\sigma_{p1(A1)}(EMP_1 \cup EMP_2 \cup EMP_3) = \sigma_{p1(A1)}(EMP_1) \cup \sigma_{p1(A1)}(EMP_2) \cup \sigma_{p1(A1)}(EMP_3)$$



Ejercicio en clase

Realizar las actividades del siguiente ejercicio. La respuesta no se incluye en estas notas, se sugiere agregar la respuesta en sus apuntes.

Suponer el siguiente esquema de fragmentación y consulta a procesar.

$$\begin{aligned} EMP_1 &= \sigma_{emp-id \leq 100}(E) \\ EMP_2 &= \sigma_{emp-id > 100 \text{ and } emp-id \leq 200}(E) \\ EMP_3 &= \sigma_{emp-id > 200}(E) \end{aligned}$$

```
select *
from emp
where id =150
```

Obtener:

- Expresión de reconstrucción
- Expresión algebraica
- Consulta localizada.
- Expresión de reconstrucción:

Reducción por join

- Empleado al existir operaciones de Join en una consulta.
- La simplificación se realiza aplicando la propiedad de distributividad con uniones eliminando expresiones que generan conjuntos vacíos.

$$(R_1 \cup R_2) \bowtie S = (R_1 \bowtie S) \cup (R_2 \bowtie S)$$



Ejercicio en clase

Realizar las actividades del siguiente ejercicio. La respuesta no se incluye en estas notas, se sugiere agregar la respuesta en sus apuntes.

Suponer el siguiente esquema de fragmentación y consulta a procesar.

$$EMP_1 = \sigma_{emp-id \leq 100}(E)$$

$$EMP_2 = \sigma_{emp-id > 100 \text{ and } emp-id \leq 200}(E)$$

$$EMP_3 = \sigma_{emp-id > 200}(E)$$

$$AE_1 = \sigma_{emp-id \leq 200}(AE)$$

$$AE_2 = \sigma_{emp-id > 200}(AE)$$

```
select *
from e, ae
where e.emp_id = ae.emp_id
```

- A. Obtener la consulta algebraica
- B. Obtener la consulta localizada.
- C. Obtener la consulta reducida

4.2.2. Reducción por fragmentación derivada

- Las operaciones Join son importantes por ser frecuentes y costosas.
- Se optimiza empleando fragmentación horizontal primaria cuando las relaciones se fragmentan empleando la condición del join.
- En una fragmentación horizontal derivada típicamente si R se fragmenta C/R a S, los fragmentos de R y S tienen el mismo valor del predicado empleado para fragmentar se encuentran en el mismo sitio.
- Fragmentación horizontal derivada debe ser empleada en relaciones 1:M.



Ejercicio en clase

Realizar las actividades del siguiente ejercicio. La respuesta no se incluye en estas notas, se sugiere agregar la respuesta en sus apuntes.

Suponer el siguiente esquema de fragmentación y consulta a procesar. Obtener:

$$EMP_1 = \sigma_{puesto-id=10}(E)$$

$$EMP_2 = \sigma_{puesto-id \neq 10}(E)$$

$$AE_1 = AE \bowtie_{emp-id} EMP_1$$

$$AE_2 = AE \bowtie_{emp-id} EMP_2$$

```
select *
from e,ae
where e.emp_id=ae.emp_id
and puesto_id =5
```

- A. Consulta algebraica
- B. Consulta localizada.
- C. Consulta reducida

4.2.3. Reducción por fragmentación vertical



Ejercicio en clase

Realizar las actividades del siguiente ejercicio. La respuesta no se incluye en estas notas, se sugiere agregar la respuesta en sus apuntes.

Suponer el siguiente esquema de fragmentación y consulta. Obtener:

$$EMP_1 = \pi_{emp-id, nombre}(E)$$

$$EMP_2 = \pi_{emp-id, titulo}(E)$$

```
select nombre
from emp;
```

- A. Consulta algebraica
- B. Consulta localizada
- C. Consulta reducida.

4.2.4. Fragmentación híbrida

En este caso se aplica una combinación de las técnicas vistas anteriormente

Reglas principales:

- Eliminar relaciones vacías en fragmentaciones horizontales.
- Eliminar relaciones vacías en fragmentaciones verticales
- Distribuir joins sobre uniones
- Eliminar joins sin uso.



Ejercicio en clase

Realizar las actividades del siguiente ejercicio. La respuesta no se incluye en estas notas, se sugiere agregar la respuesta en sus apuntes.

Suponer el siguiente esquema de fragmentación y consulta. Obtener:

$$E_1 = \sigma_{emp-id \leq 500}(\pi_{emp-id, e.nombre}(E))$$

$$E_2 = \sigma_{emp-id > 500}(\pi_{emp-id, e.nombre}(E))$$

$$E_3 = \pi_{emp-id, puesto-id}(E)$$

```
select e.nombre  
from emp  
where emp_id = 600
```

- A. Consulta algebraica
- B. Consulta localizada
- C. Consulta reducida.