

Bases de Datos
Unidad 3
Procesamiento de consultas

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

Procesamiento de consultas

- Estrategias de procesamiento
- Equivalencia de expresiones
- Optimización usando álgebra relacional

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Procesamiento de consultas

- Son las actividades implicadas en el análisis sintáctico, la validación, la optimización y la ejecución de la consulta. [connolly, Begg]
 - Transforma una consulta escrita en un lenguaje de programación (SQL), en una estrategia de ejecución correcta y eficiente expresada en un lenguaje de bajo nivel (álgebra relacional) y ejecutar la estrategia para extraer los datos requeridos.

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Optimización de consultas

- Es la actividad de seleccionar una estrategia de ejecución eficiente para el procesamiento de una consulta. [connolly, Begg]
 - Su objetivo consiste en elegir aquella que minimice el uso de los recursos.
 - Reducir el tiempo de ejecución de las consultas = tiempo de ejecución de todas las operaciones individuales que componen la consulta.

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Estrategias de procesamiento

```
SELECT *
FROM Staff s, Branch b
WHERE s.branchNo = b.branchNo AND
(s.position = 'Manager' AND b.city = 'London');
```

- (1) $\sigma_{(\text{position}='Manager') \wedge (\text{city}='London')} \wedge$
 $(\text{Staff.branchNo}=\text{Branch.branchNo}) (\text{Staff} \times \text{Branch})$
- (2) $\sigma_{(\text{position}='Manager') \wedge (\text{city}='London')} ($
 $\text{Staff}_{\text{Staff.branchNo}=\text{Branch.branchNo}} \text{Branch})$
- (3) $(\sigma_{\text{position}='Manager'}(\text{Staff}))_{\text{Staff.branchNo}=\text{Branch.branchNo}}$
 $(\sigma_{\text{city}='London'}(\text{Branch}))$

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

• Suposiciones:

- 1000 tuplas en Staff; 50 tuplas en Branch
- 50 administradores (Managers); 5 sucursales en Londres.
- No hay índices o claves de ordenación
- Los resultados de operaciones intermedias son almacenadas en disco.
- El costo de escritura final se ignora
- Las tuplas son accesadas de una en una.

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

- Los costos en accesos a disco son:

$$(1) (1000 + 50) + 2*(1000 * 50) = 101\ 050$$

- 1000 + 50 accesos a disco para leer cada una de las dos relaciones.
- 1000*50 para el producto cruz.
- 1000*50 para leer el producto cruz tomando en cuenta la condición.

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

- Los costos en accesos a disco son:

$$(2) 2*1000 + (1000 + 50) = 3\ 050$$

- 1000 + 50 accesos a disco para leer cada una de las dos relaciones.
- La combinación de las dos relaciones tiene 1000 tuplas, una por empleado (1 empleado trabaja en una sola sucursal).
- 1000 accesos a disco para leer el resultado de la combinación.

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

$$(3) \quad 1000 + 2*50 + 5 + 50 = 1\,155$$

- 1000 accesos a disco para determinar si se trata de un empleado con categoría Manager.
- Se produce una relación con 50 tuplas.
- Se requiere 50 accesos a disco para leer en Branch las sucursales de Londres.
- Se genera una relación con 5 tuplas.
- Se combina la relación Staff y Branch reducidas, se requieren (50 +5) accesos a disco.

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

- El producto cartesiano de las operaciones de combinación es más costoso que la operación de selección.
- La tercera operación reduce de manera significativa el tamaño de las relaciones que están siendo combinadas.

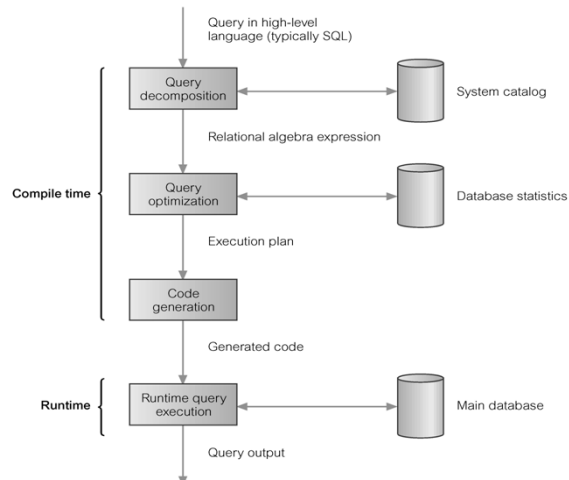
M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Procesamiento de consultas

- Descomposición
 - Análisis sintáctico y validación
- Optimización
- Generación de código y ejecución

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Descomposición de consultas



M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Descomposición de consultas

- Su objetivo es transformar consultas de alto nivel en álgebra relacional (AR) y revisar que la consulta es sintáctica y semánticamente correcta.
- Las etapas son:
 - Análisis
 - Normalización
 - Análisis semántico
 - Simplificación
 - Reestructuración de la consulta

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

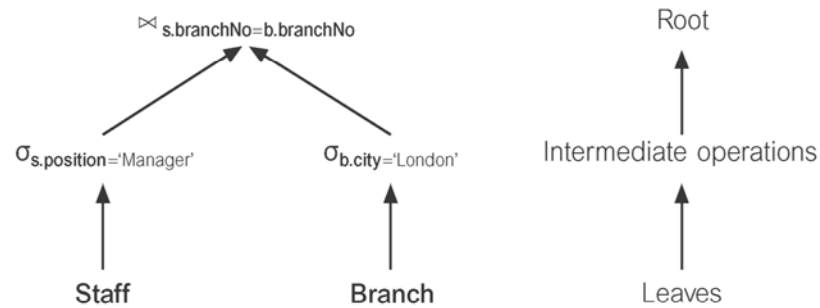
Descomposición de consultas

- **Análisis**
 - Analiza lexic y sintácticamente la consulta usando técnicas de un compiladora
 - Verifica relaciones y la existencia de los atributos
 - Verifica que las operaciones son apropiadas para el tipo de objeto.

```
SELECT staff_no  
FROM Staff  
WHERE position > 10;
```

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Descomposición de consultas



M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Descomposición de consultas

- Normalización
 - Convierte la consulta en una forma normalizada que puede manipularse más fácilmente.
 - Pueden aplicarse reglas de transformación
 - Forma normal conjuntiva
 - $(position = 'Manager' \vee salary > 20000) \wedge (branchNo = 'B003')$
 - Forma normal disyuntiva
 - $(position = 'Manager' \wedge branchNo = 'B003') \vee (salary > 20000 \wedge branchNo = 'B003')$

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

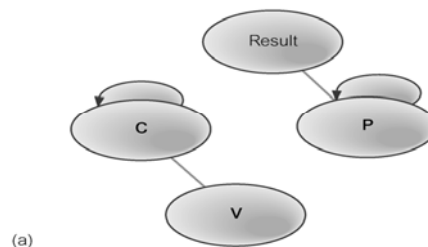
Descomposición de consultas

- Análisis semántico
 - Rechaza las consultas normalizadas que estén incorrectamente formuladas o contradictorias.
 - Consulta incorrecta
 - Si los componentes no contribuyen a la generación del resultado.
 - Consulta contradictoria
 - Si el predicado no puede ser satisfecho por ninguna tupla.

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Descomposición de consultas

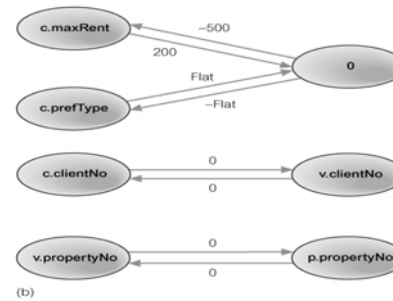
```
SELECT p.propertyNo, p.street
FROM Client c, Viewing v, PropertyForRent p
WHERE c.clientNo = v.clientNo AND
      c.maxRent >= 500 AND
      c.prefType = 'Flat' AND p.ownerNo = 'CO93';
```



M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Descomposición de consultas

```
SELECT p.propertyNo, p.street
FROM Client c, Viewing v, PropertyForRent p
WHERE c.maxRent > 500 AND
      c.clientNo = v.clientNo AND
      v.propertyNo = p.propertyNo AND
      c.prefType = 'Flat' AND c.maxRent < 200;
```



M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Descomposición de consultas

- Simplificación
 - Sus objetivos son:
 - Detectar cualificaciones redundantes,
 - eliminar subexpresiones comunes y
 - transformar la consulta en otra consulta que sea semánticamente equivalente pero que se pueda calcular más fácil y eficientemente.

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Método heurístico de optimización

- Se utilizan reglas de transformación para convertir una expresión del álgebra relacional (AR) en otra forma equivalente que se sepa que es más eficiente.
- Recordar que:
 - Es más eficiente realizar la operación de selección sobre una relación antes de utilizar dicha relación en una combinación, en lugar de primero efectuar la combinación y luego la selección.

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Reglas de transformación en AR

- Conmutatividad de selecciones

$$\sigma_p(\sigma_q(R)) = \sigma_q(\sigma_p(R))$$

- Ejemplo:

$$\sigma_{\text{branchNo}='B003'}(\sigma_{\text{salary}>15000}(\text{Staff})) = \sigma_{\text{salary}>15000}(\sigma_{\text{branchNo}='B003'}(\text{Staff}))$$

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Reglas de transformación en AR

- En una secuencia de operaciones de Proyección, sólo la última se requiere.

$$\Pi_L \Pi_M \dots \Pi_N(R) = \Pi_L(R)$$

- Ejemplo:

$$\Pi_{\text{IName}} \Pi_{\text{branchNo, IName}}(\text{Staff}) = \Pi_{\text{IName}}(\text{Staff})$$

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Reglas de transformación en AR

- Conmutatividad de Selección y Proyección.
- Si un predicado p involucra solo atributos en la lista de proyección, las operaciones de Selección y Proyección son conmutativas.

$$\Pi_{A_i, \dots, A_m}(\sigma_p(R)) = \sigma_p(\Pi_{A_i, \dots, A_m}(R))$$

donde $p \in \{A_1, A_2, \dots, A_m\}$

- Ejemplo:

$$\Pi_{\text{fName, IName}}(\sigma_{\text{IName}='Beech'}(\text{Staff})) = \sigma_{\text{IName}='Beech'}(\Pi_{\text{fName, IName}}(\text{Staff}))$$

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Reglas de transformación en AR

- Conmutatividad de operaciones Theta Join y producto cartesiano.

$$R \bowtie_p S = S \bowtie_p R$$

$$R \times S = S \times R$$

- ◆ **Aplica también a operaciones Equijoin y Natural join.:**

$$\text{Staff} \bowtie_{\text{staff.branchNo}=\text{branch.branchNo}} \text{Branch} =$$

$$\text{Branch} \bowtie_{\text{staff.branchNo}=\text{branch.branchNo}} \text{Staff}$$

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Reglas de transformación en AR

- Conmutatividad de operaciones Theta join y producto cartesiano.
- Si el predicado de la Selección involucra solo atributos de una de las dos relaciones unidas, la Selección y Join (o el producto cartesiano) conmutan.

$$\sigma_p(R \bowtie_r S) = (\sigma_p(R)) \bowtie_r S$$

$$\sigma_p(R \times S) = (\sigma_p(R)) \times S$$

$$\text{donde } p \in \{A_1, A_2, \dots, A_n\}$$

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Reglas de transformación en AR

- Si el predicado de Selección es predicado conjuntivo teniendo la forma $(p \wedge q)$, donde p solo involucra atributos de R , y q solo atributos de S , las operaciones de Selección y Theta Join conmutan así.

$$\sigma_{p \wedge q}(R \bowtie_r S) = (\sigma_p(R)) \bowtie_r (\sigma_q(S))$$

$$\sigma_{p \wedge q}(R \times S) = (\sigma_p(R)) \times (\sigma_q(S))$$

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Reglas de transformación en AR

- Ejemplo

$$\sigma_{\text{position}='Manager' \wedge \text{city}='London'}(\text{Staff} \bowtie_{\text{Staff.branchNo}=\text{Branch.branchNo}} \text{Branch}) =$$

$$(\sigma_{\text{position}='Manager'}(\text{Staff})) \bowtie_{\text{Staff.branchNo}=\text{Branch.branchNo}} (\sigma_{\text{city}='London'}(\text{Branch}))$$

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Reglas de transformación en AR

- Conmutatividad de operaciones Theta join y producto cartesiano.
- Si la lista de Proyección es de la forma $L = L_1 \cup L_2$, donde L_1 solo tiene atributos de R , y L_2 solo tiene atributos de S , la condición dada de la operación Join solo contiene atributos de L , la Proyección y Theta Join son conmutativas.

$$\Pi_{L_1 \cup L_2}(R \bowtie_r S) = (\Pi_{L_1}(R)) \bowtie_r (\Pi_{L_2}(S))$$

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Reglas de transformación en AR

- Si la condición del Join contiene atributos adicionales que no estén en L ($M = M_1 \cup M_2$ donde M_1 solo tiene atributos de R , y M_2 sólo tiene atributos de S , entonces, una operación de Proyección final es requerida:

$$\Pi_{L_1 \cup L_2}(R \bowtie_r S) = \Pi_{L_1 \cup L_2}((\Pi_{L_1 \cup M_1}(R)) \bowtie_r (\Pi_{L_2 \cup M_2}(S)))$$

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Reglas de transformación en AR

- Ejemplo:

$$\Pi_{\text{position, city, branchNo}}(\text{Staff} \bowtie_{\text{Staff.branchNo=Branch.branchNo}} \text{Branch}) =$$

$$(\Pi_{\text{position, branchNo}}(\text{Staff}) \bowtie_{\text{Staff.branchNo=Branch.branchNo}} (\Pi_{\text{city, branchNo}}(\text{Branch})))$$

- Con la última regla:

$$\Pi_{\text{position, city}}(\text{Staff} \bowtie_{\text{Staff.branchNo=Branch.branchNo}} \text{Branch}) =$$

$$\Pi_{\text{position, city}}((\Pi_{\text{position, branchNo}}(\text{Staff}) \bowtie_{\text{Staff.branchNo=Branch.branchNo}} (\Pi_{\text{city, branchNo}}(\text{Branch}))))$$

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Reglas de transformación en AR

- Conmutatividad de Unión e Intersección:

$$R \cup S = S \cup R$$

$$R \cap S = S \cap R$$

- No ocurre lo mismo en la diferencia

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Reglas de transformación en AR

- Conmutatividad de Selección y operaciones de conjunto (Unión, Intersección y diferencia):

$$\sigma_p(R \cup S) = \sigma_p(S) \cup \sigma_p(R)$$

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

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

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Reglas de transformación en AR

- Conmutatividad de Proyección y Unión:

$$\Pi_L(R \cup S) = \Pi_L(S) \cup \Pi_L(R)$$

- Asociatividad de Unión e Intersección (pero no de diferencia)

$$(R \cup S) \cup T = S \cup (R \cup T)$$

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

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Reglas de transformación en AR

- Conmutatividad de Proyección y Unión:

$$\Pi_L(R \cup S) = \Pi_L(S) \cup \Pi_L(R)$$

- Asociatividad de Unión e Intersección (pero no de diferencia)

$$(R \cup S) \cup T = S \cup (R \cup T)$$

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

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Reglas de transformación en AR

- Asociatividad de Theta Join y Producto Cartesiano.
- El producto cartesiano y el Natural Join son siempre asociativos.

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

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

- Si la condición q involucra atributos de S y T entonces Theta join es asociativo:

$$(R \bowtie_p S) \bowtie_{q \wedge r} T = R \bowtie_{p \wedge r} (S \bowtie_q T)$$

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Reglas de transformación en AR

- Ejemplo:

(Staff \bowtie Staff.staffNo=PropertyForRent.staffNo
PropertyForRent)

\bowtie ownerNo=Owner.ownerNo \wedge staff.IName=Owner.IName Owner =

Staff \bowtie staff.staffNo=PropertyForRent.staffNo \wedge staff.IName=IName
(PropertyForRent \bowtie ownerNo Owner)

M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

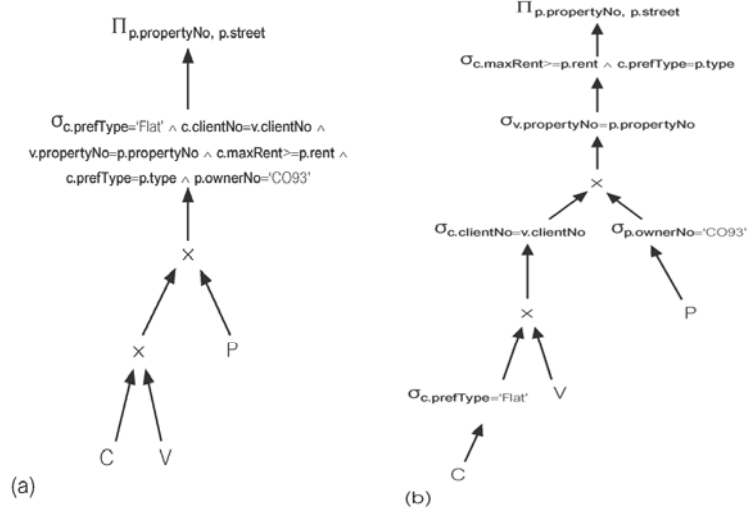
Uso de reglas de transformación

- Para los inquilinos prospectos que estén buscando apartamentos, localizar los inmuebles que satisfacen sus requisitos y son propiedad del propietario C093.

```
SELECT p.propertyNo, p.street
FROM Client c, Viewing v, PropertyForRent p
WHERE c.prefType = 'Flat' AND
      c.clientNo = v.clientNo AND
      v.propertyNo = p.propertyNo AND
      c.maxRent >= p.rent AND
      c.prefType = p.type AND
      p.ownerNo = 'C093';
```

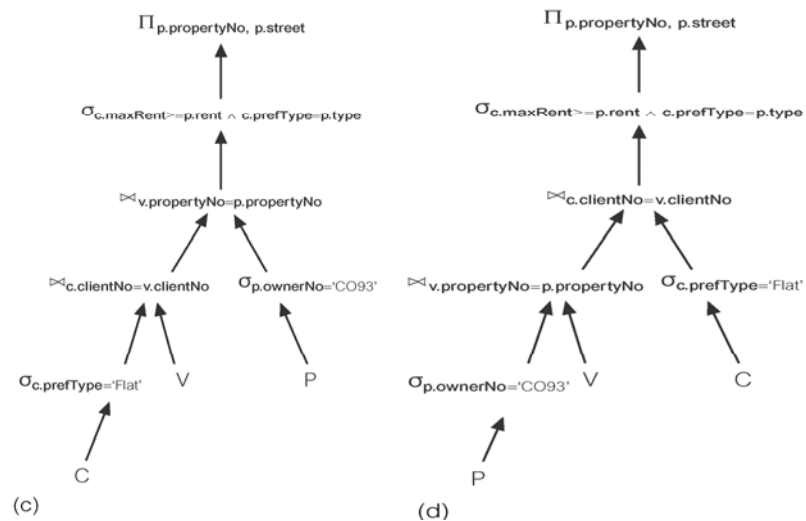
M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Uso de reglas de transformación



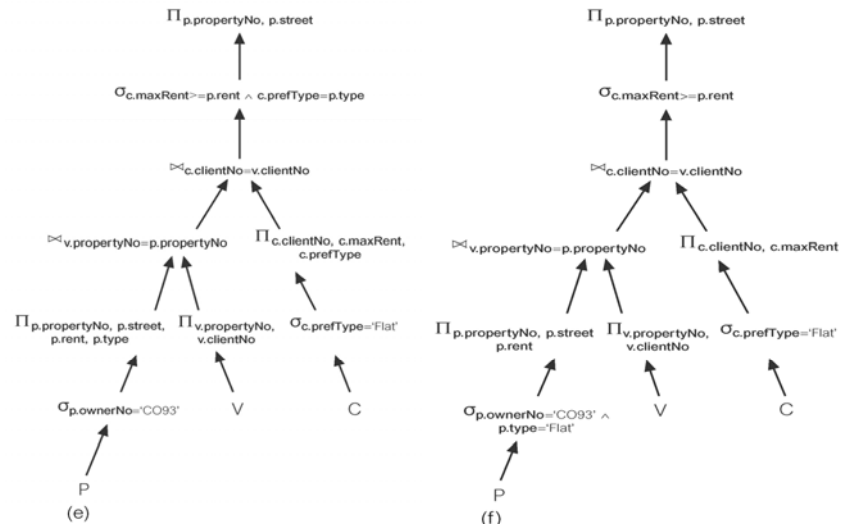
M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Uso de reglas de transformación



M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL

Uso de reglas de transformación



M.C. Luis R. Basto Díaz, Ene - Abr
2009, CEL