




UNIVERSIDAD DE SEVILLA  
Escuela Técnica Superior  
de Ingeniería Informática  
Departamento de Lenguajes  
y Sistemas Informáticos

# Procesamiento y Optimización de Consultas

*Grupo de Ingeniería del Software y Bases de Datos  
Departamento de Lenguajes y Sistemas Informáticos  
Universidad de Sevilla  
diciembre 2011*





UNIVERSIDAD DE SEVILLA  
Escuela Técnica Superior  
de Ingeniería Informática  
Departamento de Lenguajes  
y Sistemas Informáticos

Procesamiento y Optimización de Consultas

- Objetivos de este tema
  - Conocer las fases del procesado de consultas en bases de datos relacionales.
  - Conocer los principales factores que afectan a la eficiencia de una consulta en bases de datos relacionales.
  - Ser capaz de escribir consultas eficientes a bases de datos relacionales aplicando simplificación de expresiones, equivalencias de álgebra relacional y heurísticas.


1. Procesado de consultas

2. Coste de una consulta

3. Simplificación de expresiones

4. Equivalencias en AR

5. Heurísticas de optimización



diciembre 2011

Introducción a la Ingeniería del Software y a los Sistemas de Información

1

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Procesado de consultas

2. Coste de una consulta

3. Simplificación de expresiones

4. Equivalencias en AR

5. Heurísticas de optimización

Procesamiento y Optimización de Consultas

Fases habituales del procesado de consultas

Esquema Base Datos

Consulta SQL

Estadísticas Base Datos

Consulta AR

Datos

Resultado

Plan de ejecución

Optimización

Ejecución

Análisis y Traducción

dicembre 2011

Introducción a la Ingeniería del Software y a los Sistemas de Información

2

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Procesado de consultas

2. Coste de una consulta

3. Simplificación de expresiones

4. Equivalencias en AR

5. Heurísticas de optimización

Procesamiento y Optimización de Consultas

Ejemplo de procesado de consulta

– Supongamos una base de datos relacional sobre vinos, productores y cosechas donde las estadísticas son las siguientes:

• Vinos: 500 tuplas

• Cosechas: 1.200 tuplas

• **select** cantidad > 100 ( Cosechas ): 100 tuplas

Vinos

PK vid

vnombre

vendimia

graduación

Cosechas

PK cid

FK1 vid

FK2 pid

año

cantidad

Productores

PK pid

pnombre

región


dicembre 2011

Introducción a la Ingeniería del Software y a los Sistemas de Información

3

IISSI

2



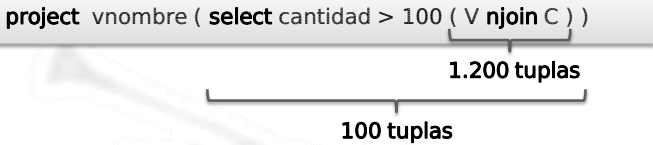
Escuela Técnica Superior  
de Ingeniería Informática  
Departamento de Lenguajes  
y Sistemas Informáticos

# Procesamiento y Optimización de Consultas

1. Procesado de consultas
2. Coste de una consulta
3. Simplificación de expresiones
4. Equivalencias en AR
5. Heurísticas de optimización

- Ejemplo de procesamiento de consulta
  - Queremos ejecutar la siguiente consulta en SQL:
 

```
select V.vnombre from V Vinos, C Cosechas
where V.vid = C.vid and C.cantidad > 100;
```
  - El primer paso es traducirla a álgebra relacional.
  - **Opción 1:** join, selección, proyección



**project** vnombre ( **select** cantidad > 100 ( V **njoin** C ) )


1.200 tuplas

100 tuplas

diciembre 2011

Introducción a la Ingeniería del Software y a los Sistemas de Información

4



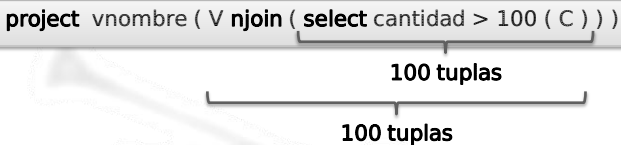
Escuela Técnica Superior  
de Ingeniería Informática  
Departamento de Lenguajes  
y Sistemas Informáticos

# Procesamiento y Optimización de Consultas

1. Procesado de consultas
2. Coste de una consulta
3. Simplificación de expresiones
4. Equivalencias en AR
5. Heurísticas de optimización

- Ejemplo de procesamiento de consulta
  - Queremos ejecutar la siguiente consulta en SQL:
 

```
select V.vnombre from V Vinos, C Cosechas
where V.vid = C.vid and C.cantidad > 100;
```
  - El primer paso es traducirla a álgebra relacional.
  - **Opción 2:** selección, join, proyección



**project** vnombre ( V **njoin** ( **select** cantidad > 100 ( C ) ) )


100 tuplas

100 tuplas

diciembre 2011

Introducción a la Ingeniería del Software y a los Sistemas de Información

5



UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior  
de Ingeniería Informática  
Departamento de Lenguajes  
y Sistemas Informáticos

1. Procesado de consultas
2. Coste de una consulta
3. Simplificación de expresiones
4. Equivalencias en AR
5. Heurísticas de optimización

## Procesamiento y Optimización de Consultas

- Ejemplo de procesamiento de consulta
  - Queremos ejecutar la siguiente consulta en SQL:
 

```
select V.vnombre from V Vinos, C Cosechas
where V.vid = C.vid and C.cantidad > 100;
```
  - El primer paso es traducirla a álgebra relacional.
  - **Opción 3:** producto cartesiano, selección, proyección

```
project vnombre (
  select V.vid = C.vid and cantidad > 100 ( V product C ) )
```


600.000 tuplas

100 tuplas

diciembre 2011

Introducción a la Ingeniería del Software y a los Sistemas de Información

6



Escuela Técnica Superior  
de Ingeniería Informática  
Departamento de Lenguajes  
y Sistemas Informáticos

1. Procesado de consultas
2. Coste de una consulta
3. Simplificación de expresiones
4. Equivalencias en AR
5. Heurísticas de optimización

## Procesamiento y Optimización de Consultas

- Ejemplo de procesamiento de consulta
  - ¿Cuál de las tres consultas en álgebra relacional es más eficiente? ¿Cuál seleccionaría? ¿Por qué?

```
project vnombre ( select cantidad > 100 ( V njoin C ) )
```

1.200 tuplas

100 tuplas

```
project vnombre ( V njoin ( select cantidad > 100 ( C ) ) )
```


100 tuplas

100 tuplas

```
project vnombre (
  select V.vid = C.vid and cantidad > 100 ( V product C ) )
```

600.000 tuplas

100 tuplas



diciembre 2011

Introducción a la Ingeniería del Software y a los Sistemas de Información

7

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Procesado de consultas

2. Coste de una consulta


3. Simplificación de expresiones

4. Equivalencias en AR

5. Heurísticas de optimización

Procesamiento y Optimización de Consultas

- Coste de una consulta
  - El coste, en tiempo de ejecución, de una consulta a una base de datos depende de:
    - Tiempo de acceso al sistema de E/S (90%)
    - Tiempo de procesamiento de CPU (10%)
  - El tiempo de acceso al sistema de E/S depende de:
    - Volumen de datos: número y tamaño de las tuplas, tanto de las relaciones involucradas como de los resultados intermedios
    - Organización física: índices, tablespaces, ...
    - Tamaño de los buffers en memoria para almacenar los resultados intermedios



diciembre 2011

Introducción a la Ingeniería del Software y a los Sistemas de Información

8

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Procesado de consultas

2. Coste de una consulta


3. Simplificación de expresiones

4. Equivalencias en AR

5. Heurísticas de optimización

Procesamiento y Optimización de Consultas

- Para reducir el coste de una consulta...
  - Reducir los accesos al sistema de E/S
    - Reducir los resultados intermedios: menos accesos y más probabilidad de que quepan en los buffers en memoria.\*
    - Seleccionar por atributos indexados: accesos mucho más eficientes y búsquedas mucho más rápidas.
  - Reducir el procesamiento de CPU
    - Simplificar expresiones de selección: ahorran tiempo de procesamiento y, a veces, accesos al sistema de E/S.



diciembre 2011

Introducción a la Ingeniería del Software y a los Sistemas de Información

9

\* Si los resultados intermedios no caben en los buffers en memoria, deben almacenarse en disco.

IISSI

5

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Procesado de consultas

2. Coste de una consulta

3. Simplificación de expresiones

4. Equivalencias en AR

5. Heurísticas de optimización

Procesamiento y Optimización de Consultas

- Ejemplo de simplificación de expresiones

```
select región from Productores
where nombre = 'Osborne' → o
or not ( región = 'Jerez' ) → j
and ( región = 'Jerez' or región = 'La Mancha' ) → m
and not ( región = 'La Mancha' );
```
- Simplificando la condición del **where**...
$$o \vee \bar{j} \wedge (j \vee m) \wedge \bar{m} \equiv o \vee ((\bar{j} \wedge j) \vee (\bar{j} \wedge m)) \wedge \bar{m} \equiv$$
$$o \vee (\bar{j} \wedge m) \wedge \bar{m} \equiv o \vee ((\bar{j} \wedge \bar{m}) \wedge (m \wedge \bar{m})) \equiv o$$

falso falso falso

dicembre 2011

Introducción a la Ingeniería del Software y a los Sistemas de Información

10

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Procesado de consultas

2. Coste de una consulta

3. Simplificación de expresiones

4. Equivalencias en AR

5. Heurísticas de optimización

Procesamiento y Optimización de Consultas

- Ejemplo de simplificación de expresiones

```
select región from Productores
where nombre = 'Osborne'
or not ( región = 'Jerez' )
and ( región = 'Jerez' or región = 'La Mancha' )
and not ( región = 'La Mancha' );
```


↓

```
select región from Productores
where nombre = 'Osborne';
```

dicembre 2011

Introducción a la Ingeniería del Software y a los Sistemas de Información

11



UNIVERSIDAD DE SEVILLA  
Escuela Técnica Superior  
de Ingeniería Informática  
Departamento de Lenguajes  
y Sistemas Informáticos

1. Procesado de consultas

2. Coste de una consulta

3. Simplificación de expresiones

4. Equivalencias en AR

5. Heurísticas de optimización


Procesamiento y Optimización de Consultas

- Algunas equivalencias lógicas
  - Conjunciones y negaciones básicas
    - $a \wedge \neg a \equiv falso$
    - $a \wedge falso \equiv falso$
    - $a \wedge cierto \equiv a$
    - $a \vee \neg a \equiv cierto$
    - $a \vee falso \equiv a$
    - $a \vee cierto \equiv cierto$
  - Leyes de De Morgan
    - $\neg (a \wedge b) \equiv \neg a \vee \neg b$
    - $\neg (a \vee b) \equiv \neg a \wedge \neg b$
  - Implicación
    - $a \rightarrow b \equiv \neg a \vee b$
  - Cuantificadores
    - $\forall x \cdot P(x) \equiv \neg \exists x \cdot \neg P(x)$
    - $\forall x \cdot \neg P(x) \equiv \neg \exists x \cdot P(x)$

diciembre 2011

Introducción a la Ingeniería del Software y a los Sistemas de Información

12



UNIVERSIDAD DE SEVILLA  
Escuela Técnica Superior  
de Ingeniería Informática  
Departamento de Lenguajes  
y Sistemas Informáticos

1. Procesado de consultas

2. Coste de una consulta

3. Simplificación de expresiones

4. Equivalencias en AR

5. Heurísticas de optimización

Procesamiento y Optimización de Consultas

- Equivalencias en álgebra relacional\*
  - Cascada de selecciones
    - $\sigma_{c_1 \wedge c_2}(R) \equiv \sigma_{c_1}(\sigma_{c_2}(R))$
  - Cascada de proyecciones ( $a_1 \subseteq a_2 \subseteq \dots \subseteq a_n$ )
    - $\Pi_{a_1}(\Pi_{a_2}(\dots \Pi_{a_n}(R))) \equiv \Pi_{a_1}(R)$
  - Asociatividad de joins
    - $(R \bowtie S) \bowtie T \equiv R \bowtie (S \bowtie T)$

diciembre 2011


Introducción a la Ingeniería del Software y a los Sistemas de Información

13

\* Se usa la notación teórica por simplificar. Consultar el tema de álgebra relacional.

IISSI

7



UNIVERSIDAD DE SEVILLA  
Escuela Técnica Superior de Ingeniería Informática  
Departamento de Lenguajes y Sistemas Informáticos

1. Procesado de consultas

2. Coste de una consulta

3. Simplificación de expresiones

4. Equivalencias en AR

5. Heurísticas de optimización

Procesamiento y Optimización de Consultas

• Equivalencias en álgebra relacional\*

– Distribución de la selección sobre el join

- $\sigma_{c_R \wedge c_S}(R \bowtie S) \equiv \sigma_{c_R}(R) \bowtie \sigma_{c_S}(S)$
- Donde  $c_R$  sólo incluye atributos de  $R$  y  $c_S$  sólo de  $S$ .

– Distribución de la proyección sobre el join


- $\Pi_{A_R \cup A_S}(R \bowtie S) \equiv \Pi_{A_R \cup A_S}(\Pi_{A_R \cup A_{\bowtie}}(R) \bowtie \Pi_{A_S \cup A_{\bowtie}}(S))$
- Donde  $A_R$  sólo incluye atributos de  $R$ ,  $A_S$  sólo de  $S$  y  $A_{\bowtie}$  los atributos sobre los que se realiza el join.

\* Se usa la notación teórica por simplificar. Consultar el tema de álgebra relacional.

diciembre 2011

Introducción a la Ingeniería del Software y a los Sistemas de Información

14



UNIVERSIDAD DE SEVILLA  
Escuela Técnica Superior de Ingeniería Informática  
Departamento de Lenguajes y Sistemas Informáticos

1. Procesado de consultas

2. Coste de una consulta

3. Simplificación de expresiones

4. Equivalencias en AR

5. Heurísticas de optimización

Procesamiento y Optimización de Consultas

• Equivalencias en álgebra relacional\*

– Distribución de la proyección sobre la selección

- $\Pi_A(\sigma_c(R)) \equiv \Pi_A(\sigma_c(\Pi_{A \cup A_c}(R)))$
- Donde  $A_c$  son los atributos de  $R$  que se usan en la condición de la selección.

– Distribución de la selección sobre  $\cup$ ,  $\cap$  y  $-$ 

- $\sigma_c(R \cup S) \equiv \sigma_c(R) \cup \sigma_c(S)$  igual para  $\cap$  y  $-$

– Distribución de la proyección sobre la unión

- $\Pi_A(R \cup S) \equiv \Pi_A(R) \cup \Pi_A(S)$


\* Se usa la notación teórica por simplificar. Consultar el tema de álgebra relacional.

diciembre 2011

Introducción a la Ingeniería del Software y a los Sistemas de Información

15





UNIVERSIDAD DE SEVILLA  
Escuela Técnica Superior de Ingeniería Informática  
Departamento de Lenguajes y Sistemas Informáticos


1. Procesado de consultas

2. Coste de una consulta

3. Simplificación de expresiones

4. Equivalencias en AR

5. Heurísticas de optimización




Procesamiento y Optimización de Consultas

- **Heurísticas de optimización de consultas**
  1. **Realizar las selecciones tan pronto como sea posible**
    - Reduce el número de tuplas de los resultados intermedios.
  2. **Realizar las selecciones sobre atributos indexados antes que sobre los no indexados**
    - Reduce las operaciones de E/S al usar los índices y se ejecutan mucho más rápido. Aplicar la cascada de selecciones si es necesario.
  3. **Realizar las proyecciones tan pronto como sea posible**
    - Reduce el número de atributos de los resultados intermedios.

diciembre 2011

Introducción a la Ingeniería del Software y a los Sistemas de Información

16



UNIVERSIDAD DE SEVILLA  
Escuela Técnica Superior de Ingeniería Informática  
Departamento de Lenguajes y Sistemas Informáticos


1. Procesado de consultas

2. Coste de una consulta

3. Simplificación de expresiones

4. Equivalencias en AR

5. Heurísticas de optimización



Procesamiento y Optimización de Consultas

- **Heurísticas de optimización de consultas**
  4. **Realizar las operaciones de selección y join más restrictivas primero\***
    - Reduce el número de tuplas de los resultados intermedios. Aplicar asociatividad si es necesario.
  5. **Eliminar proyecciones redundantes**
    - Reduce el número de atributos de los resultados intermedios. Aplicar la cascada de proyecciones.
  6. **Usar DISTINCT sólo cuando sea imprescindible**
    - Evita tener que comparar resultados intermedios para detectar duplicados y eliminarlos del resultado.

diciembre 2011

Introducción a la Ingeniería del Software y a los Sistemas de Información

17

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Sistemas Informáticos

1. Procesado de consultas

2. Coste de una consulta

3. Simplificación de expresiones

4. Equivalencias en AR

5. Heurísticas de optimización

Procesamiento y Optimización de Consultas

• Bibliografía

– R. Elmasri, S. Navathe, **Fundamentos de Sistemas de Bases de Datos** (5ª edición). Ed. Addison-Wesley, 2007.

• Capítulo 15.

– R. Ramakrishnan, J. Gehrke, **Database Management Systems** (3ª edición). Ed. McGraw-Hill, 2003.

• Capítulos 12 y 15.

– A. Silberschatz *et al.*, **Fundamentos de Bases de Datos** (5ª edición). Ed. McGraw-Hill, 2006.

• Capítulos 13 y 14.

diciembre 2011

Introducción a la Ingeniería del Software y a los Sistemas de Información

18

IISSI

10