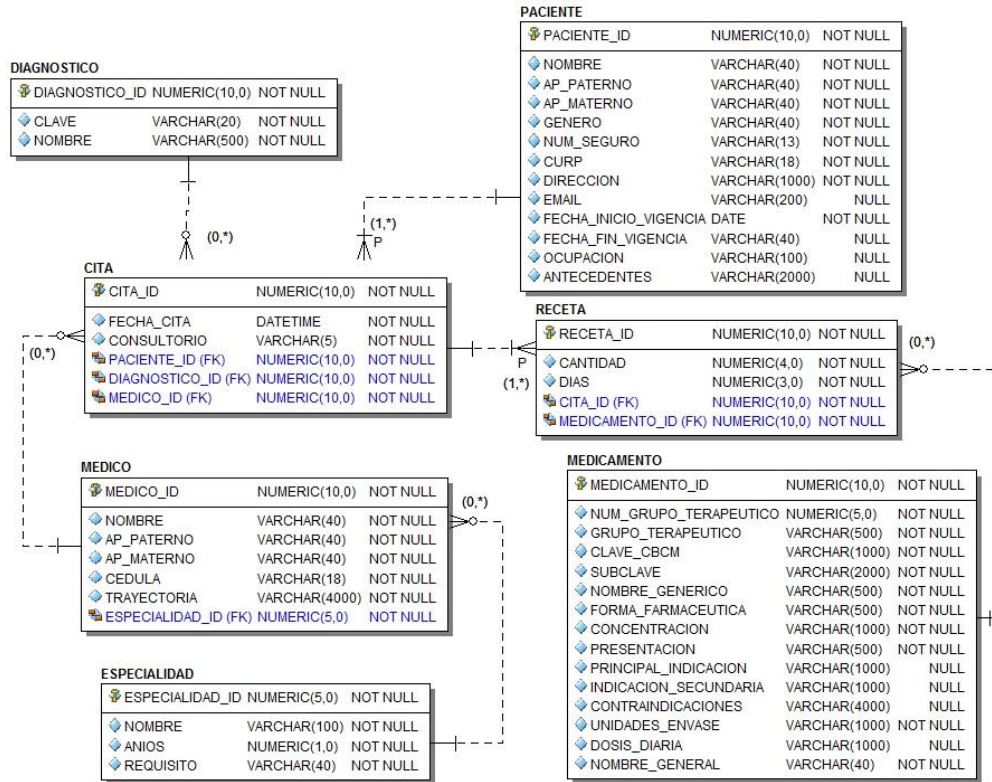


TEMA 5. OPTIMIZACIÓN DE CONSULTAS DISTRIBUIDAS
SERIE DE EJERCICIOS - PARTE 3

Nota: Los ejercicios se entregan de forma individual en cualquier formato.

Para los siguientes ejercicios considere nuevamente el caso de estudio del registro de citas médicas.



1.1. ESTADÍSTICAS.

1.1.1. Ejercicio 1.

Considere la siguiente tabla de datos que contiene el número de infracciones que un automovilista ha cometido.

- Con base a los valores de la tabla, construya un histograma para el campo num. infracciones.
- Empleando las fórmulas de selectividad, determine el número de registros que se obtendrían al ejecutar cada una de las siguientes consultas:

```
select * from infracciones where num_infracciones = 3
```

```
select * from infracciones where num_infracciones = 10
```

 Realice el cálculo bajo las 2 siguientes variantes:
 - Considerando el histograma
 - Sin considerar el histograma.
- Empleando fórmulas de selectividad determinar el número de registros esperados para:
 - ```
select * from infracciones where num_infracciones > 2
```
  - ```
select * from infracciones where num_infracciones > 11
```

1.2. ORDENAMIENTO DE OPERACIONES JOIN DISTRIBUIDOS

1.2.1. Ejercicio 1.

Considere la siguiente sentencia SQL. Empleando el algoritmo INGRES para optimizar y ordenar JOINS:

```
select m.nombre
from medicamento me, receta r, cita c, medico m
where me.medicamento_id = r.medicamento_id
and r.cita_id = c.cita_id
and c.medico_id = m.medico_id
and me.subclave = '010.000.0091.00';
```

- A. Descomponer la sentencia empleando las técnicas de separación y sustitución. Emplear el siguiente orden de separación:
medicamento -> receta -> cita -> medico. Para la etapa de sustitución, asumir que solo existen 2 médicos con id's {1,2}.
- B. ¿Cuántas consultas separadas y sustituidas se generaron?

1.2.2. Ejercicio 2.

Considerar una BDD de 2 sitios.

- En el sitio 1 se encuentra la relación Medico (M)
- En el sitio 2 se encuentra la relación Especialidad (E)
- El manejador ha empleado la estrategia 2 del algoritmo System R* Distribuido que dice:

Estrategia 2: Enviar la relación Inner al sitio de la relación Outer para ejecutar NESTED LOOP.

- Notar que, en este caso, el JOIN no se puede ejecutar hasta recibir el 100% de las tuplas de la relación inner
- Lo anterior implica que la relación inner se debe almacenar en una tabla temporal Temp.

Considerar los siguientes costos.

- Card(M) = 5000
 - Costo de leer un registro de disco: 3
 - Tamaño en Bytes de un registro: 2
 - Costo de transmisión por byte: 4
 - Costo para guardar un registro en memoria (tabla temporal) = 0.5
 - Card(E) = 53
 - Costo de leer un registro en memoria (tabla temporal) = 0.4
- A. Determinar el costo total que implicaría realizar una operación JOIN entre las tablas M, E. Considere que la tabla de menor cardinalidad es la que se va a transferir.

1.2.3. Ejercicio 3.

- Considere las 2 siguientes tablas de datos que se encuentran en 2 sitios.
- Se desea obtener el identificador del medicamento, nombre general, cantidad e identificador de la cita para todos los medicamentos que fueron recetados y cuyo Nombre inicie con 'P'.
- El manejador requiere verificar si la técnica de uso de Semi-Joins proporciona un costo total de transmisión menor con respecto a la ejecución directa del JOIN. Para ello el manejador estimará el costo total de aplicar un Semi-Join de la siguiente forma:

$$R \bowtie_A S \Leftrightarrow R \bowtie_A (S \ltimes_A R)$$

MEDICAMENTO se encuentra en el sitio 1

Medicamento_id	Nombre_general
1	Aspirina
2	Paracetamol
3	Mimepris
4	Penicilina
5	Morfina
6	Acetol

RECETA se encuentra en el sitio 2

Receta_id	cantidad	Medicamento_id	Cita_id
100	4	1	40
200	3	2	40
300	5	2	30
400	2	4	30
500	4	4	20
600	2	4	20

Considerar adicionalmente los siguientes costos:

- Tamaño promedio en bytes de un atributo numérico: 4 bytes
- Tamaño promedio en bytes de un valor del campo `nombre_general`: 10 bytes
- Costo de transmisión por byte: 2.5

Paso 1:

El manejador envía los datos necesarios de MEDICAMENTO hacia el sitio 2 para calcular el semi-join en el sitio 2.

- Generar una expresión SQL que obtenga los datos necesarios que serán enviados al sitio 2.
- Dibujar una tabla con los datos que serán enviados al Sitio 2.
- Calcular el costo de transmisión CT para enviar la tabla anterior considerar: $CT = \text{bytes a enviar} * \text{costo de transmisión por byte}$
- Suponer que la tabla enviada al sitio 2 se almacena en una tabla temporal llamada TEMP. Genere una sentencia SQL que obtendrá el resultado del semi-join. La sentencia deberá incluir los atributos que deberán enviarse de regreso al sitio 1.
- Dibujar una tabla con los datos que obtendrá la sentencia del punto anterior.
- Obtener el valor del costo de transmisión CT requerido para enviar la tabla de datos anterior.
- Suponer que la tabla enviada de regreso al sitio 1 se almacena en una tabla temporal TEMP. Genere una sentencia SQL que obtendrá el resultado final de la operación JOIN.
- Dibujar una tabla con los datos obtenidos por la sentencia anterior.
- Determinar si la estrategia de Semi-joins generó un costo menor. Para ello considerar la fórmula:

$$CT(\pi_A(S)) + CT(R \bowtie S) < CT(R)$$

S= MEDICAMENTO

R=RECETA

- Es decir, si se omite el uso de un Semi-Join, la tabla R se tendría que transmitir al Sitio 1 para hacer la operación JOIN. Esto implica un valor para $CT(R)$.
- Considerando la fórmula anterior, obtener el valor de $CT(R)$ considerando los costos mostrados anteriormente y determine cuál de las 2 estrategias es la mas adecuada.