

FUNDAMENTOS DE BASES DE DATOS

Examen Febrero 2010

Solución

Ejercicio 1 (25 puntos).

Se desea modelar la realidad correspondiente a una productora de discos y conciertos musicales. La productora cuenta con un conjunto de empresarios de los cuales se conoce su CI, nombre y teléfonos. Los empresarios pueden ser representantes de artistas o productores. De los productores interesa registrar si ha realizado trabajos en el exterior, mientras que de los representantes se registra el año en el que comenzó a trabajar como representante.

Los artistas tienen asignado un código, este código o el nombre de los artistas los identifica. Además, se les conoce el tipo (si es solista melódico, dúo folklórico, banda de rock, etc.), y el nombre de cada uno de los premios que ha ganado. Todos los artistas tienen un representante y se registra el porcentaje que obtiene el representante. Se sabe que los representantes no les cobran el mismo porcentaje a todos los artistas.

Todos los artistas graban al menos un disco. Todos los discos tienen un número que los identifica e interesa saber su año de lanzamiento y el título del disco. Un disco puede contener canciones de más de un artista e interesa registrar qué canciones graba cada artista en un disco. De cada canción se conoce su título y duración.

Los discos son producidos por un único productor. Para cada disco, cada artista realiza un ensayo con el productor de ese disco. De cada ensayo se registra la fecha y la dirección donde se realizó el mismo.

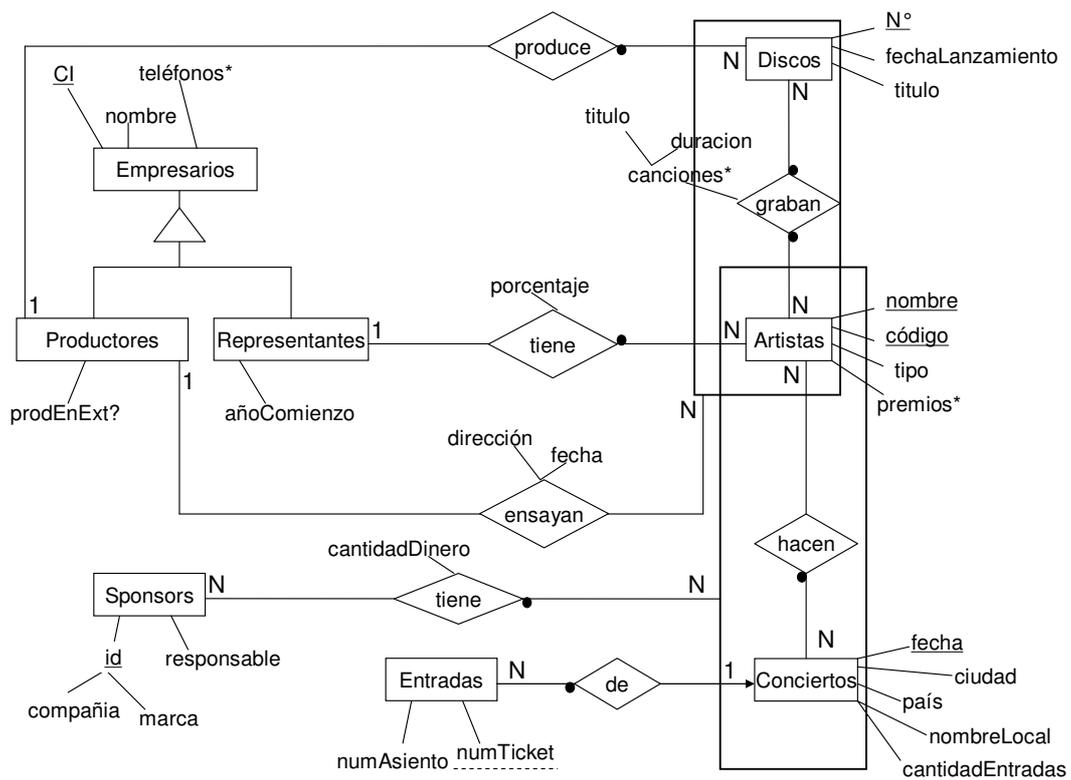
Se sabe que hay artistas que no realizan conciertos nunca. Los conciertos son identificados por la fecha que fueron realizados y se registra el país, la ciudad, el nombre del establecimiento donde se realizó el mismo y la cantidad de entradas vendidas. En un concierto pueden participar varios artistas.

Para la organización de cada concierto se registra el número de cada entrada y el número de asiento correspondiente a esa entrada. Los números de las entradas pueden repetirse en conciertos diferentes.

En un concierto puede haber diferentes sponsors para cada uno de los artistas que participa en el concierto. A su vez, un sponsor puede promover a varios artistas en conciertos.

De los sponsors se conoce la marca presentada, el nombre de la compañía que la representa (ambos datos identifican al sponsor) y el nombre de un responsable. Además, se registra la cantidad de dinero con la que el sponsor apoyó al artista en el concierto que participó.

Se pide: Modelo Entidad Relación completo.



- $\text{Representantes} \cap \text{Productores} = \emptyset$
- $\text{Representantes} \cup \text{Productores} = \text{Empresarios}$
- Un artista que participa en un disco ensaya con el productor que produce dicho disco.

Ejercicio 2 (25 puntos).

Considere la siguiente relación con el siguiente conjunto de dependencias funcionales:

$$R(A,B,C,D,E,G,H) F=\{BD\rightarrow A, B\rightarrow H, DH\rightarrow E, GC\rightarrow D, D\rightarrow G\}$$

- 1) Indicar cuáles de las siguientes dependencias están en F^+ y cuáles no. Justifique.
 - a. $AB\rightarrow G$
 - b. $CBG\rightarrow E$
- 2) Hallar todas las claves de R. Justificar.
- 3) Llevar a 3NF con join sin pérdida y preservación de dependencias con respecto a F, aplicando los algoritmos vistos en el curso.
- 4) Llevar el resultado en 3NF que obtuvo en la parte 3 a BCNF, aplicando los algoritmos vistos en el curso. Discuta si se pierden dependencias funcionales o no.
- 5) Dada la siguiente descomposición de R: $\rho = \{ R_1(A,B,D,G), R_2(A,C,D,H,E) \}$, decir si ρ tiene join sin pérdida con respecto a F, justificando.

Solución:

$$R(A,B,C,D,E,G,H) F=\{BD\rightarrow A, B\rightarrow H, DH\rightarrow E, GC\rightarrow D, D\rightarrow G\}$$

- 6)
 - a. $AB\rightarrow G$
 $(AB)^+ = \{A,B,H\}$, $G \notin (AB)^+$, por lo tanto la dependencia $AB\rightarrow G \notin F^+$.
 - b. $CBG\rightarrow E$
 $(CBG)^+ = \{C,B,G,H,D,A,E\}$, $E \in (CBG)^+$, por lo tanto la dependencia $CBG\rightarrow E \in F^+$

7)

B, C no aparecen del lado derecho en ninguna df \Rightarrow pertenecen a toda clave
 A, E no aparecen del lado izquierdo en ninguna df, pero sí del lado derecho
 \Rightarrow no pertenecen a ninguna clave

$$(BC)^+ = \{ B, C, H \}$$

$$(BCD)^+ = \{ B, C, D, A, H, E, G \} \Rightarrow \text{(BCD) es clave}$$

$$(BCG)^+ = \{ B, C, G, D, A, H, E \} \Rightarrow \text{(BCG) es clave}$$

$$(BCH)^+ = (BC)^+ = \{ B, C, H \}$$

Cualquier otra clave no contiene a A ni E (porque no pertenecen a ninguna clave).

Tampoco puede contener a BCD o BCG (porque sería superclave) pero tiene que contener a BC.

La única combinación posible es BCH, y ya probamos que no es superclave, por lo tanto no es clave. Entonces no hay más claves.

8)

El algoritmo visto en el curso que garantiza obtener una descomposición en 3NF con JSP y preservación de df requiere partir de un cubrimiento minimal del conjunto de dependencias.

(i) Cubrimiento minimal de F.

1er. Paso – Las dfs ya tienen un solo atributo a la derecha.

2º. Paso – Buscamos atributos redundantes a la izquierda.

$(B)^+ = \{B, H\}$
 $(D)^+ = \{D, G\}$ $BD \rightarrow A$ no tiene atributos redundantes.
 $(H)^+ = \{H\}$ $DH \rightarrow E$ no tiene atributos redundantes.
 $(G)^+ = \{G\}$
 $(C)^+ = \{C\}$ $CG \rightarrow D$ no tiene atributos redundantes.

3er. Paso – Buscamos dfs redundantes.

Se observa que no se puede eliminar ninguna df, porque en cada df que consideramos, los atributos de la derecha no aparecen a la derecha en ninguna otra df, por lo tanto no podrían ser determinados de ninguna otra forma. (Propiedad demostrada en el práctico del curso).

Por lo tanto **F es un conjunto de dependencias minimal.**

(ii) Formamos los esquemas relación según lo indica el algoritmo. Considerando cada una de las df de F.

R1 (BDA)
 R2 (BH)
 R3 (DHE)
 R4 (GCD)
 R5 (DG) - Se elimina por estar incluida en R4

(iii) Agregamos un esquema relación con una clave, para asegurar JSP.

R5 (BCD)

Descomposición en 3NF con preservación de dfs y JSP:

R1(ABD), R2(BH), R3(DEH), R4(GCD), R5(BCD)

9)

El primer paso es determinar si la descomposición de la parte anterior se encuentra en BCNF según F. Para esto es necesario analizar en que forma normal se encuentra cada uno de los esquemas que componen la descomposición. Por lo que es necesario proyectar las df de F en cada esquema y determinar sus claves.

$\Pi_F(R1) = \{BD \rightarrow A\}$ BD clave, por lo tanto el lado derecho de las df es superclave entonces el esquema está en BCNF.

$\Pi_F(R2) = \{B \rightarrow H\}$ B clave, por lo tanto el lado derecho de las df es superclave entonces el esquema está en BCNF.

$\Pi_F(R3) = \{DH \rightarrow E\}$ DH clave, por lo tanto el lado derecho de las df es superclave entonces el esquema está en BCNF.

$\Pi_F(R4) = \{GC \rightarrow D, D \rightarrow G\}$ GC y DC claves, por lo tanto el lado derecho de la df $D \rightarrow G$ en su lado derecho no tiene una superclave entonces viola la condición de BCNF.

$\Pi_F(R5) = \{\}$ No hay dfs, está en BCNF.

Según el algoritmo visto en el curso que garantiza BCNF con JSP se descompone R4 en:

R41(DG) con $\Pi_F(R41) = \{D \rightarrow G\}$, D clave, por lo tanto el lado derecho de las df es

superclave entonces el esquema está en BCNF.

R42(CD), incluido en R5 por lo tanto se elimina.

Descomposición en BCNF con JSP:

R1(ABD), R2(BH), R3(DEH), R41(DG), R5(BCD)

Para determinar si se pierden df en la descomposición es necesario determinar si $F \subseteq (\cup_i \Pi_{F^+}(R_i))^+$.

Sea $K = \cup_i \Pi_{F^+}(R_i) = \{BD \rightarrow A, B \rightarrow H, DH \rightarrow E, D \rightarrow G\}$

Todas las dependencias de F pertenecen a K con la excepción de $GC \rightarrow D$. Por lo tanto es necesario determinar si esta df se deduce de las df de K.

$(GC)^+_{K^+} = \{G, C\}$, por lo tanto $GC \rightarrow D \notin K^+$, entonces es una df de F que se pierde en la descomposición.

10) $\rho = \{R1(A,B,D,G), R2(A,C,D,H,E)\}$

Según el teorema visto en el curso para descomposiciones en dos subesquemas, ρ tiene JSP con respecto a F sii

$$\begin{aligned} & R1 \cap R2 \rightarrow R1 - R2 \in F^+ \\ \text{O} & \\ & R1 \cap R2 \rightarrow R2 - R1 \in F^+ \end{aligned}$$

$$\begin{aligned} R1 \cap R2 &= \{A, D\} \\ R1 - R2 &= \{B, G\} \\ R2 - R1 &= \{C, H, E\} \end{aligned}$$

$$(AD)^+ = \{A, D, G\}$$

$$\begin{aligned} B \notin (AD)^+ \text{ o sea que } AD \rightarrow B \notin F^+ &\Rightarrow R1 \cap R2 \rightarrow R1 - R2 \notin F^+ \\ C \notin (AD)^+, H \notin (AD)^+, E \notin (AD)^+ \text{ o sea que } AD \rightarrow CHE \notin F^+ & \\ \Rightarrow R1 \cap R2 \rightarrow R2 - R1 \notin F^+ & \end{aligned}$$

Aplicando el teorema antes mencionado es posible afirmar que:
la descomposición ρ **no tiene JSP**.

Ejercicio 3 (25 puntos).

La administración de un jardín de infantes mantiene toda la información en una base de datos con el siguiente esquema:

DOCENTES (CI, Nombre, Teléfono, añoIngreso)

Esta tabla contiene información acerca de los docentes que trabajan en el jardín de infantes. Registra la cédula de identidad del docente, su nombre, su teléfono y el año de ingreso al trabajo.

ACTIVIDADES (IdAct, Nombre, Duración)

Esta tabla contiene información acerca de las actividades que se realizan en el jardín. Registra el identificador de la actividad, el nombre (gimnasia, natación, artes plásticas, música, etc.) y la duración (en minutos) de la misma en cada clase.

ALUMNOS (CI, Nombre, Teléfono, Nivel)

Esta tabla contiene información acerca de los alumnos del jardín de infantes. Registra la cédula de identidad del alumno, su nombre, su teléfono y el nivel (2 años, 3 años, etc.), al cual asiste el niño.

ASIGNACIÓN_ACT (IdDoc, IdAct, DiaSemana, Hora)

Esta tabla contiene la asignación de los docentes a las actividades dentro del jardín de infantes. Cada docente puede ser asignado a varias actividades y además puede ser asignado a la misma actividad en varios horarios diferentes en el mismo día o en distintos días.

ASISTENCIA_ACT (IdAlum, IdAct, IdDoc)

Esta tabla contiene información acerca de los alumnos que asisten a cada actividad con el docente correspondiente. Un alumno asiste a una actividad con un único docente.

En este esquema no existen tablas vacías y se cumplen las siguientes dependencias de inclusión.

- a. $\Pi_{IdDoc}(ASIGNACIÓN_ACT) \subseteq \Pi_{CI}(DOCENTES)$
- b. $\Pi_{IdAct}(ASIGNACIÓN_ACT) \subseteq \Pi_{IdAct}(ACTIVIDADES)$
- c. $\Pi_{IdDoc, IdAct}(ASISTENCIA_ACT) \subseteq \Pi_{IdDoc, IdAct}(ASIGNACION_ACT)$
- d. $\Pi_{IdAlum}(ASISTENCIA_ACT) \subseteq \Pi_{CI}(ALUMNOS)$

Resolver en álgebra relacional las siguientes consultas:

1) Devolver el nombre de los docentes que hayan ingresado antes del año 2000 y su asignación es solamente en el horario de la mañana (antes de las 12:00 hs.)

$$A = \Pi_{CI, Nombre} (\sigma_{añoIngreso < '2000'} (DOCENTES))$$

Obtengo docentes que ingresaron antes del año 2000.

$$B = \Pi_{IdDoc} (\sigma_{Hora \geq '12:00'} (ASIGNACIÓN_ACT))$$

Busco todos los docentes con actividades que se dictan en el horario de la tarde.

$$C = \Pi_{IdDoc} (\sigma_{Hora < '12:00'} (ASIGNACIÓN_ACT))$$

Busco todos los docentes con actividades que se dictan en el horario de la mañana.

$$D = C - B$$

Id de los docentes con asignación solamente en el horario de la mañana.

$$SOL = \Pi_{CI, Nombre} (A \bowtie_{IdDoc = CI} D)$$

Obtengo CI y nombre de los docentes que solo dictan actividades en el horario de la mañana y que han ingresado antes del año 2000.

2) Devolver el nombre y teléfono de todos los alumnos del nivel 3 que asisten a actividades con duración mínima.

$$A = \Pi_{\$1} (ACTIVIDADES \bowtie_{\$3 > \$6} ACTIVIDADES)$$

$$B = \Pi_{IdAct} (ACTIVIDADES) - A$$

Obtengo todas las actividades tales que no hay otra actividad con duración menor.

$$C = \Pi_{CI, Nombre, Teléfono} (\sigma_{Nivel = '3'} (ALUMNOS))$$

Obtengo los datos de todos los alumnos de 3 años.

$$D = \Pi_{Nombre, Teléfono, IdAct} (C \bowtie_{CI = IdAlum} ASISTENCIA_ACT)$$

Obtengo las actividades de todos los alumnos de 3 años.

SOL = $\Pi_{\text{Nombre, Teléfono}} (D * B)$
Obtengo los datos de todos los alumnos de 3 años que asisten a las actividades con menor duración.

Resolver en cálculo relacional las siguientes consultas:

- 3) Devolver el nombre y teléfono de los alumnos que asistieron a todas las actividades que tienen una duración mayor de 30 minutos.

$$R = \{t.\text{Nombre}, t.\text{Teléfono} / \text{ALUMNOS}(t) \wedge$$

$$(\forall u)(\text{ACTIVIDADES}(u) \wedge u.\text{Duración} > 30$$

$$\rightarrow$$

$$(\exists w)(\text{ASISTENCIA_ACT}(w) \wedge w.\text{IdAct} = u.\text{IdAct} \wedge$$

$$w.\text{IdAlum} = t.\text{CI}$$

$$)$$

$$)$$

$$(\exists z)(\text{ACTIVIDADES}(z) \wedge z.\text{Duración} > 30$$

$$)$$

$$\}$$

- 4) Obtener los nombres de los docentes que están asignados solamente a una actividad.

$$R = \{t.\text{Nombre} / \text{DOCENTES}(t) \wedge$$

$$(\exists v)(\text{ASIGNACIÓN_ACT}(v) \wedge v.\text{IdDoc} = t.\text{CI} \wedge$$

$$\neg (\exists w)(\text{ASIGNACIÓN_ACT}(w) \wedge w.\text{IdDoc} = t.\text{CI} \wedge$$

$$w.\text{IdAct} \neq v.\text{IdAct}$$

$$)$$

$$)$$

$$\}$$

Resolver en SQL, sin utilizar vistas ni sub-consultas en el FROM, las siguientes consultas:

- 5) Obtener el nombre y el año de ingreso de los docentes que realizan más de 4 actividades distintas a las cuales asisten niños del nivel 5.

Segunda opción:

```
SELECT D.Nombre, D.añoIngreso
FROM Docentes D
WHERE D.Ci IN
    (
        SELECT AD.IdDoc
        FROM ASIGNACION_ACT AD, ASISTENCIA_ACT AA, ALUMNOS A
        WHERE AD.IdAct = AA.IdAct AND AA.IdAlum = A.CI AND A.Nivel = 5
        GROUP BY AD.IdDoc
        HAVING count(distinct AD.IdAct) > 4
    )
```

- 6) Dar los nombres de los docentes que dictan algunas de las actividades de menor duración los martes o los jueves.

Sin usar las funciones built_in del SQL.

```
SELECT d.Nombre
FROM DOCENTES d, ASIGNACIÓN_ACT a, ACTIVIDADES a1
WHERE d.CI = a.IdDoc
AND a.IdAct = a1.IdAct
```

```

AND (a.DiaSemana = 'martes' OR a.DiaSemana = 'jueves')
AND NOT EXISTS (SELECT *
                 FROM ACTIVIDADES a2
                 WHERE a2.IdAct = a1.IdAct
                 AND a2.Duración < a1.Duración);

```

Usando las funciones built_in del SQL.

```

SELECT d.Nombre
FROM DOCENTES d, ASIGNACIÓN_ACT a, ACTIVIDADES a1
WHERE d.CI = a.IdDoc
AND a.IdAct = a1.IdAct
AND (c.DiaSemana = 'martes' OR c.DiaSemana = 'jueves')
AND a1.Duración IN (SELECT MIN(a2.Duración)
                   FROM ACTIVIDADES a2
                   WHERE a2.IdAct = a1.IdAct);

```

Ejercicio 4 (25 puntos).

Considere el siguiente esquema:

ACTIVIDADES (IdAct, Nombre, Duracion)
ASIGNACION_ACT (IdDoc, IdAct, DiaSemana, Hora)

A continuación se presenta una consulta sobre dicho esquema:

```

SELECT S.IdDoc
FROM ACTIVIDADES A, ASIGNACION_ACT S
WHERE S.IdAct = A.IdAct
AND A.Nombre = 'GIMNASIA'

```

Con el siguiente catálogo:

Tabla	Columna	Valores distintos
ACTIVIDADES	IdAct	50
ACTIVIDADES	Nombre	50
ASIGNACION_ACT	IdAct	50
ASIGNACION_ACT	IdDoc	100

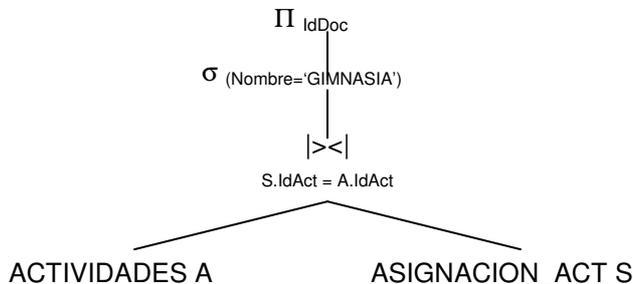
Tabla	Cant. tuplas	Factor de bloqueo
ACTIVIDADES	50	4
ASIGNACION_ACT	400	10
ACTIVIDADES >< ASIGNACION_ACT		2

Indice	Tabla/Atributo	Tipo	Cant. Niveles
Act_Nombre	ACTIVIDADES/Nombre	Secundario	2

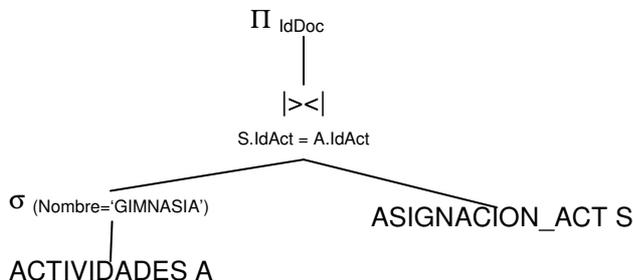
Se pide:

a) Dados los siguientes planes lógicos, indique cuál debería tener mejor performance y justifique su respuesta.

1)



2)



b) Construya el mejor plan físico que pueda para cada uno de los planes anteriores. Calcule los costos. Justifique sus respuestas.

NOTA: Asuma que tiene disponibles 3 buffers para el intercambio de datos entre disco y memoria.

Solución:

a) El esquema lógico 2 debería tener mejor performance. Las heurísticas vistas en el curso nos indican que hay que realizar las selecciones lo antes posible, de modo de reducir los registros a reunir. Por esta razón la opción 2 debería tener mejor performance.

b)

El mejor plan físico para el **plan 1** es el siguiente:

- join por bucle anidado por bloque
- selección por búsqueda lineal

Calculamos el costo de realizar el primer plan físico:

- nA = cantidad de registros de ACTIVIDADES
- fbrA = factor de bloqueo de ACTIVIDADES
- nS = cantidad de registros de ASIGNACION_ACT
- fbrS = factor de bloqueo de ASIGNACION_ACT
- fbrX = factor de bloqueo del join
- bR = cantidad de bloques de ACTIVIDADES
- bS = cantidad de bloques de ASIGNACION_ACT
- M = cantidad de buffers para el intercambio de datos
- |T1| = cantidad de tuplas del join
- bT1 = cantidad de bloques del join

$$\begin{aligned} \text{Costo de lectura del join} &= bR + \lceil bR/(M-2) \rceil * bS = \\ &= \lceil nA/fbrA \rceil + \lceil (nA/fbrA)/(M-2) \rceil * \lceil nS/fbrS \rceil = \\ &= \lceil 50/4 \rceil + \lceil (50/4)/(3-2) \rceil * \lceil 400/10 \rceil = 13 + (13*40) = 533 \end{aligned}$$

$$\text{Costo de escritura del join} = bT1 = \lceil |T1| / fbrX \rceil = \lceil 400 / 2 \rceil = 200$$

$$\text{Costo de la selección} = \lceil bT1 / 2 \rceil = 100$$

$$\text{Costo total} = 533 + 200 + 100 = \mathbf{833}$$

El mejor plan físico para el **plan 2** es el siguiente:

- selección por búsqueda por índice secundario
- join por bucle anidado por bloque

Calculamos el costo de realizar el segundo plan físico:

x = cantidad de niveles del índice secundario

s = cantidad tuplas con un valor en particular seleccionado

|T2| = cantidad de tuplas de la selección

bT2 = cantidad de bloques de la selección

$$\text{Costo de lectura de la selección} = x + s = 2 + (50/50) = 3$$

$$\text{Costo de escritura de la selección} = bT2 = \lceil |T2| / fbrA \rceil = \lceil 1/4 \rceil = 1$$

$$\text{Costo del join} = bT2 + \lceil bT2/(M-2) \rceil * bS = 1 + \lceil 1/(3-2) \rceil * (400/10) = 41$$

$$\text{Costo total} = 3 + 1 + 41 = \mathbf{45}$$

Implementaciones de los Operadores.

Oper.	Algoritmo	Costo	Condición	Organización
$\sigma_c(R)$	Búsqueda Lineal	b_R peor caso, $b_R/2$ promedio	Cualquier Caso	Cualquiera
	Búsqueda Binaria	$\log_2 b_R + \lceil s/bf_R \rceil - 1$	Cualquier caso	Registros ordenados
	Índice Primario	$x + 1$	Por igualdad a un valor	Registros Ordenados
	Hash	1 o 2 según el tipo	Por igualdad a un valor	Cualquiera
	Índices Primario	$x + (b/2)$ (promedio)	Por relación de orden.	Índice ordenado
	Índice Cluster	$x + \lceil s/bf_R \rceil$	Cualquier Caso	Registros Ordenados

CSI-INCO

Fundamentos de Bases de Datos

45

Implementaciones de los Operadores.

Oper.	Algoritmo	Costo	Cond.	Organización
$\sigma_c(R)$	Índice secundario B+	$x + s$ peor caso	Cualquier Caso	Cualquiera
	Grabación Intermedia	s/bf_R	Cualquier caso	Cualquiera
$R \bowtie X \bowtie S$	Loop Anidado (registros)	$b_R + (n_R * b_s)$	Cualquier caso	Cualquiera
	Loop Anidado (bloque)	$b_R + \lceil b_R / (M - 2) \rceil * b_s$	Cualquier caso	Cualquiera
	Sort Merge	$b_R + b_s + \text{costo ords.}$	Cualquier caso	índice en disco
	Index join	$b_R + n_R * Z$	Cualquier caso	índice en disco

CSI-INCO

Fundamentos de Bases de Datos

46

donde Z depende del tipo de índice:

secundario: $Z = x + sS$, cluster: $Z = x + \lceil sS/bfS \rceil$, primario: $Z = x + 1$, hash = h