

Segundo Parcial Base de Datos 2

Ejercicio 1 (20 puntos)

En la base de datos de una empresa de servicios informáticos se tienen las siguientes tablas relacionales:

Abonados (Id_equipo, vendedor, tipo-abono, v-cuota, fecha, marca)

Esta tabla representa los equipos abonados a la empresa.

En esta tabla, para cada equipo se tiene el vendedor que realizó la suscripción, el valor actual de la cuota y la fecha en que se realizó la suscripción.

Servicios (Id_equipo, Cod-servicio, fecha, importe, status)

Esta tabla representa los servicios realizados a cada equipo. Para cada equipo, dado un código de servicio y una fecha, la tabla contiene el importe de ese servicio y el status. El atributo status indica si el importe fue saldado al contado, si es a crédito o si está con mora.

Cientes (CI_Cli, Id_equipo)

Esta tabla contiene los equipos abonados junto con los clientes dueños de los mismos.

Dada la siguiente consulta SQL:

```
SELECT CI_Cli, vendedor
FROM Cientes C, Servicios S, Abonados A
WHERE A.Id_equipo = S.Id_equipo AND
A.Id_equipo = C.Id_equipo AND
A.marca = 'Dell' AND
S.status = 'Mora'
```

Considerando la siguiente información:

Tabla	Columna	Valores Distintos
Servicios	Status	3 (dist. Uniforme)
Abonados	Marca	25 (dist. Uniforme)

Tabla	Cant. Tuplas	Información adicional
Abonados	720	
Servicios	3500	- Un equipo puede estar en estado de mora en un solo servicio. - Solo 15 equipos de los que se encuentran en mora son marca "Dell".
Cientes	720	- En esta tabla se encuentran todos los equipos abonados.

Índice	Tabla/Atributo	Tipo
Ind_Cli1	Cientes/CI_Cli	Secundario
Ind_Cli2	Cientes/Id_equipo	Primario
Ind_Abo1	Abonados/marca	Secundario
Ind_Abo2	Abonados/Id_equipo	Primario

- Construir el árbol canónico de la consulta SQL dada.
- Dar un plan lógico para la consulta, optimizado mediante las heurísticas vistas en el curso, sin considerar los tamaños. Mostrar los pasos aplicados.
- Calcular los tamaños de los resultados de cada operación aplicada (considerando solamente las operaciones de selección y join). En caso de que lo considere conveniente, modificar el plan lógico de la parte a teniendo en cuenta los tamaños.
- Dar un plan físico cualquiera para el plan lógico de la parte b, utilizando los índices cuando es posible.

Ejercicio 2 (20 puntos)

Considere las siguientes transacciones T1 y T2:

T1: r1(X) w1(X) r1(Y) w1(Z) c1

T2: r2(Y) w2(Y) r2(Z) w2(Z) c2

- Dar una historia de T1 y T2, serializable pero no recuperable.
- Dar una historia de T1 y T2, que evite abortos en cascada.
- Dar nuevas T1, T2 y una historia de ellas, mostrando bloqueos (de lectura y escritura) y desbloqueos, donde las transacciones sigan 2PL básico.
- ¿Qué cambiaría en los bloqueos y desbloqueos de T1 y T2 para asegurar que todas las historias formadas por estas transacciones sean estrictas?

Ejercicio 3 (20 puntos)

a) Dada la siguiente situación en la ejecución de una historia:

r1(X) w1(X) r2(X) r1(Y) w2(X) w1(Y) r2(Y) a1

¿Qué operaciones se revertirán y por qué?

b) Si la situación es la siguiente:

r1(X) w1(X) r2(X) r1(Y) w2(X) w1(Y) r2(Y) caída del sistema

¿Cómo recuperará el sistema si trabaja con Actualización Inmediata (en sus dos formas)?

	Algoritmo	Costo	Condición	Organización
$\sigma_c(R)$	Búsqueda Lineal	b_R (peor caso) $b_R/2$ (prom)	Todas	--
	Búsqueda Binaria	$\log_2 b_R + \lceil s/bf_R \rceil - 1$	Todas	Ordenado
	Índice Primario	$x + 1$	Igualdad	Ordenado
	Hash	1 o 2	Igualdad	--
	Índice Primario	$x + (b/2)$ (prom)	de orden	Ordenado
	Índice Cluster	$x + \lceil s/bf_R \rceil$	Todas	Ordenado
	Índice secundario B+	$x + s$ (peor caso)	Todas	--
	Grabación Intermedia	$\lceil s/bf_R \rceil$	Todas	--

	Algoritmo	Costo	Condición	Organización
$R \gg _c S$	Loop Anidado (registros)	$b_R + (n_R * b_s)$	Todas	--
	Loop Anidado (bloque)	$b_R + \lceil b_R / (M-2) \rceil * b_s$	Todas	--
	Sort Merge	$b_R + b_s +$ costo ordenación	Todas	--
	Index join	$b_R + n_R * Z$	Todas	Índice en S