

Recuperación 1^{er} Parcial de Diseño y Gestión de Bases de Datos (27/01/2021)

1. a) 0'2 puntos b) 0'6 puntos

- a) Di cuáles son las propiedades que debe cumplir el correcto procesamiento de transacciones. Define una, SÓLO UNA, claramente.

Se incluyen las cuatro definiciones, pero sólo había que poner una.

- Atomicidad: una transacción es una unidad de ejecución (o se ejecutan todas sus operaciones o no se ejecuta ninguna de ellas).
- Consistencia: una transacción debe conducir la BD de un estado consistente a otro estado consistente (*estado consistente*: se cumplen todas las restricciones de integridad del esquema).
- Aislamiento: una transacción debe ejecutarse como si se ejecutase de forma aislada (en solitario).
- Persistencia: los cambios de una transacción confirmada por el SGBD deben quedar grabados permanentemente en la BD.

- b) Dado el esquema de abajo:

- Suponiendo que la base de datos está vacía (no hay filas en las tablas), indica una actualización de la base de datos (expresada en lenguaje natural) que necesite de una transacción para llevarse a cabo en el esquema.
- Indica para cada restricción si debe comprobar en modo INMEDIATO O DIFERIDO y si es DIFERIBLE o NO DIFERIBLE.
- Define la transacción que resolvería la actualización del primer punto (para esto puedes usar las instrucciones del SQL o en castellano, pero de forma clara y no ambigua).

Empleado (dni: char(10), nombre: char(60), salario: real, dpto: char(5))

CP: {dni}

VNN: {dpto}

CAj: {dpto} → Departamento f(dpto)=código

/* dpto es el código del departamento al que pertenece el empleado */

Departamento (código: char(5), nombre: char(50), ubicación: char(15), dni_director: char(10))

CP: {código}

VNN: {nombre, dni_director}

CAj: {dni_director} → Empleado(dni)

Único: {dni_director}

/* dni_director es el dni del empleado que dirige el departamento */

- Transacción: Insertar al empleado ('0000', 'Pepa', 1.500, 'RRHH') que pertenece y dirige el departamento ('RRHH', 'Recursos Humanos', 'Planta 3ª', '0000')
- Las claves ajenas hay que diferirlas (al menos una), las demás restricciones de integridad pueden definirse como se quiera. Yo elijo para ellas INMEDIATO-DIFERIBLE

Empleado (dni: char(10), nombre: char(60), salario: real, dpto: char(5))

CP: {dni}

INMEDIATO-DIFERIBLE

VNN: {dpto}

INMEDIATO-DIFERIBLE

CAj: {dpto} → Departamento f(dpto)=código

DIFERIDO-DIFERIBLE

Departamento (código: char(5), nombre: char(50), ubicación: char(15), dni_director: char(10))

CP: {código}

INMEDIATO-DIFERIBLE

VNN: {nombre, dni_director}

INMEDIATO-DIFERIBLE

CAj: {dni_director} → Empleado(dni)

DIFERIDO-DIFERIBLE

Único: {dni_director}

INMEDIATO-DIFERIBLE

- iii. BEGIN

INSERT INTO Empleado VALUES ('0000', 'Pepa', 1.500, 'RRHH');

INSERT INTO Departamento ('RRHH', 'Recursos Humanos', 'Planta 3ª', '0000');

END;

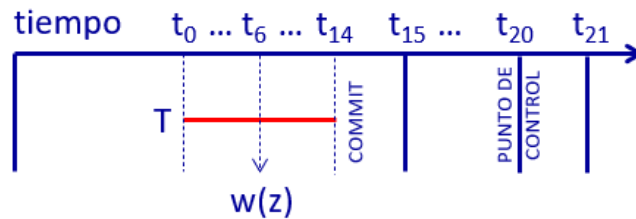
Recuperación 1^{er} Parcial de Diseño y Gestión de Bases de Datos (27/01/2021)

2. a) 0'4 puntos b) 0'2 puntos c) 0'6 puntos

a) Sean:

- El siguiente diagrama que muestra la ejecución de una transacción T .
- t el instante de tiempo en que la modificación realizada por T sobre z llega a disco.

Para cada una de las estrategias de actualización estudiadas, indica el intervalo de valores que puede tomar t ($t_i < t < t_j$) utilizando para ello los instantes de tiempo marcados en el diagrama.



- Actualización INMEDIATA-NO FORZAR: $t_6 < t < t_{20}$
- Actualización INMEDIATA-FORZAR: $t_6 < t < t_{15}$
- Actualización DIFERIDA-NO FORZAR: $t_{14} < t < t_{20}$
- Actualización DIFERIDA-FORZAR: $t_{14} < t < t_{15}$

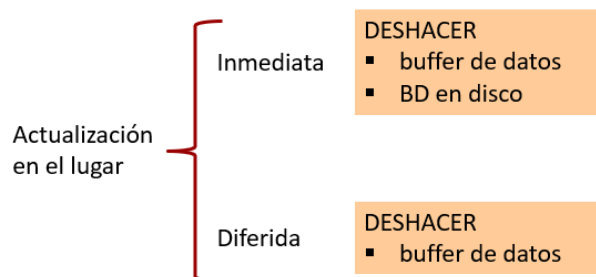
a) ¿En un sistema cuya estrategia de actualización sea DIFERIDA/FORZAR es necesario la copia de seguridad? Justifica brevemente tu respuesta.

La copia de seguridad siempre es necesaria ya que si se corrompe la bd la forma de restaurarla consiste en recuperar la copia de seguridad y rehacer todas las transacciones confirmadas desde la fecha de la copia de seguridad.

b) Explica qué hace el SGBD para restaurar la base de datos cuando se anula una transacción (*rollback* de usuario o del sistema). Considera las posibles estrategias de actualización.

En este caso es indiferente si la estrategia es FORZAR o NO-FORZAR ya que la transacción no ha alcanzado el punto de confirmación. Sí que es relevante si es INMEDIATA o DIFERIDA. Las tareas que hay que hacer en cada caso se ven en la siguiente imagen.

Recuperación transacciones anuladas (funcionamiento correcto del sistema)



Recuperación 1^{er} Parcial de Diseño y Gestión de Bases de Datos (27/01/2021)

3. a) 0'3 puntos b) 0'3 puntos c) 0'5 puntos d) 0'4 puntos e) 0'5 puntos

a) Explica brevemente cómo funciona el protocolo de bloqueo en dos fases IMPLÍCITO.

Los bloqueos y desbloques (y las promociones) se realizan implícitamente por el SGBD:

- Una operación de lectura de un elemento x implica una petición de bloque de lectura. La transacción se espera si el elemento está bloqueado en escritura.
- Una operación de escritura de un elemento x implica una petición de bloque de escritura (o una promoción del bloqueo de lectura si ya se tiene). La transacción se espera si el elemento está bloqueado en escritura.
- Los desbloques los realiza automáticamente el SGBD cuando la transacción termina (con confirmación o anulación).

b) Suponiendo que x e y están desbloqueadas, dado el siguiente plan de ejecución concurrente, donde las primeras operaciones determinan el inicio de las transacciones: $P = r_1(x), r_3(y), w_1(x), r_2(x), r_1(y), w_3(y), c_1, w_2(x), r_3(x), c_2, c_3$, comprueba si el protocolo B2F IMPLÍCITO admite el plan P . Indica las operaciones que se realizan en cada uno de los instantes. Si el protocolo lo admite, ¿cuál es el plan en serie equivalente por conflictos?

T1	T2	T3	Estado de x	Estado de y
$r_1(x)$ (bloqueo implícito de x)			Bloqueada en lectura por T1	Desbloqueada
		$r_3(y)$ (bloqueo implícito de y)	Bloqueada en lectura por T1	Bloqueada en lectura por T3
$w_1(x)$ (promoción implícita del bloqueo de x)			Bloqueada en escritura por T1	Bloqueada en lectura por T3
	$r_2(x)$ (ESPERA)		Bloqueada en escritura por T1	Bloqueada en lectura por T3

El plan no es posible con el protocolo B2F IMPLÍCITO ya que la lectura de x por la transacción T_2 nunca podría ocurrir antes de que T_1 termine y se produzca el desbloqueo implícito de x .

c) Dado el siguiente plan de ejecución concurrente, donde las marcas de tiempo de las transacciones T_1, T_2 y T_3 vienen dadas por su primera instrucción en el plan

t	T_1	T_2	T_3
t_0	$r_1(x)$		
t_1			$r_3(x)$
t_2		$r_2(x)$	
t_3	$r_1(y)$		
t_4	c_1		
t_5			$w_3(y)$
t_6		$w_2(x)$	
t_7			$r_3(x)$
t_8			c_3
t_9		c_2	

Suponiendo que las siguientes marcas de tiempo son:

$$\begin{aligned} MT_L(x) &= t_{-2} & MT_E(x) &= t_{-3} \\ MT_L(y) &= t_{-4} & MT_E(y) &= t_{-8} \end{aligned}$$

y que se cumple que $\forall i \forall j t_i < t_j$ si $i < j$.

Comprueba si el protocolo OMT admite el plan de ejecución. Si no lo admite, indicar en qué operación falla y por qué. Dibuja una tabla donde quede claro cómo van cambiando las marcas de tiempo.

Recuperación 1^{er} Parcial de Diseño y Gestión de Bases de Datos (27/01/2021)

			MT_L(x)	MT_E(x)	MT_L(y)	MT_E(y)
T1	T2	T3	t ₋₂	t ₋₃	t ₋₄	t ₋₈
r ₁ (x)			t ₀	t ₋₃	t ₋₄	t ₋₈
		r ₃ (x)	t ₁	t ₋₃	t ₋₄	t ₋₈
	r ₂ (x)		t ₂	t ₋₃	t ₋₄	t ₋₈
r ₁ (y)			t ₂	t ₋₃	t ₀	t ₋₈
c ₁			t ₂	t ₋₃	t ₀	t ₋₈
		w ₃ (y)	t ₂	t ₋₃	t ₀	t ₁
	w ₂ (x)		t ₂	t ₂	t ₀	t ₁
		r ₃ (x)	Esta operación no puede ejecutarse			

La lectura de x por T_3 no puede realizarse ya que la marca de tiempo de escritura de x es t_2 que es mayor que la marca de tiempo de T_3 que es t_1 . Si se permitiera la lectura, T_3 leería un valor de x escrito por una transacción más joven que ella (T_2).

- d) Supón un entorno concurrente controlado por el protocolo Multiversión en el que una transacción T ejecuta la instrucción $w(z)$. Supón también que hay tres versiones del elemento z (sean z_0, z_1 y z_2), que las marcas de tiempo son las que se indican más abajo y que $t_i < t_j$ si $i < j$:

	MT_L	MT_E
z_0	t_1	t_1
z_1	t_9	t_9
z_2	t_{10}	t_{10}

$MT(T) = t_8$

Sí T puede ejecutar esa operación, ¿qué versión leería?

¿Qué condición o condiciones se evalúan para permitir su ejecución?

¿Qué se controla con esa condición?

De las tres versiones de z que hay se busca la que tenga mayor marca de tiempo de escritura que cumpla que: $MT_E(z_i) \leq MT(T)$. En este caso esa versión es z_0 . Dado que $MT_L(z_0) < MT(T)$, la operación es aceptada ya que la transacción más joven que ha leído z_0 es más vieja que T .

- e) ¿Qué es un plan sin anulación en cascada? Si en estos planes no hay anulación en cascada ni lectura sucia, ¿qué problema pueden presentar para que se definan los planes estrictos?

Plan “sin anulación en cascada”: las transacciones del plan sólo pueden leer elementos de datos actualizados (escritos) por transacciones confirmadas.

Problema;

P: ... $w_1(x)$, ..., $w_2(x)$, a_1 ... P es “sin anulación en cascada”, pero, aunque no se da la lectura sucia (y por lo tanto tampoco la anulación en cascada), si T_1 se anula, la recuperación es más compleja: si se aplicara el valor_antes de la actualización de T_1 se perdería la actualización de T_2 .

La recuperación obliga al SGBD a analizar si ha habido actualizaciones posteriores del mismo elemento de datos.