

# 1<sup>er</sup> Parcial de Diseño y Gestión de Bases de Datos (26/10/2020)

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

- a) Indica clara y concisamente qué significa procesar correctamente una transacción.
- b) Dado el esquema de abajo:
- 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).

Asignatura(cod\_asg: char(5), nombre: char(50), semestre: char(2))

CP:{cod\_asg}

VNN:{nombre,semestre}

Profesor(dni: char(9), nombre: char(80), teléfono: char(8))

CP:{dni}

VNN:{nombre}

Docencia(dni: char(9), cod\_asg: char(5))

CP:{dni,cod\_asg}

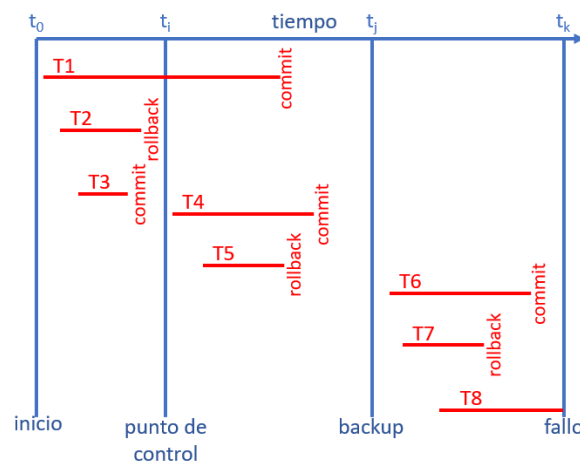
CAj:{dni}→Profesor(dni)

CAj:{cod\_asg}→Asignatura(cod\_asg)

Restricción general: "Todo profesor debe impartir docencia de al menos una asignatura".

## 2. a) 0'2 puntos b) 0'25 puntos c) 0'75 puntos

- a) En qué estrategia o estrategias de actualización no son necesarios los puntos de control. Justifica brevemente tu respuesta.
- b) ¿En un sistema cuya estrategia de actualización sea DIFERIDA/FORZAR es necesario el diario? Justifica brevemente tu respuesta.
- c) En el diagrama de más abajo se muestran las transacciones ejecutadas en el sistema. Supón que antes de hacer la copia de seguridad se bajan a disco todas las modificaciones realizadas por transacciones confirmadas. En el instante  $t_k$  se produce un fallo que obliga al SGBD a realizar tareas de recuperación cuando se vuelve a arrancar el sistema. Para cada uno de los supuestos siguientes indica qué haría el SGBD para cada transacción en cada una de las cuatro estrategias posibles de actualización:
- El fallo supone la pérdida de memoria principal.
  - El fallo supone la pérdida de la memoria secundaria donde está almacenada la base de datos, pero la copia del diario en disco no se ha perdido.
  - El fallo supone la pérdida de la memoria secundaria donde está almacenada la base de datos y también se ha perdido la copia del diario en disco.



# 1er Parcial de Diseño y Gestión de Bases de Datos (26/10/2020)

## 3. a) 0'3 puntos b) 0'5 puntos c) 0'4 puntos

Dado el siguiente plan de ejecución concurrente, donde las primeras operaciones determinan el inicio de las transacciones:  $P = r_1(x), r_2(x), w_2(x), r_2(y), r_1(z), w_3(y), r_2(k), c_2, r_3(k), r_1(k), c_1, w_3(z), c_3$

- Justifica que el plan de ejecución es serializable por conflictos.  
¿Cuáles son los posibles planes en serie equivalentes por conflictos?  
¿Es el plan cronológico uno de ellos?
- Comprueba si el protocolo B2F explícito admite el plan  $P$  dando una traza de su ejecución que incluya los bloqueos y desbloqueos necesarios. Supón que las cuatro variables están desbloqueadas al inicio del plan. También puedes usar promociones y degradación de los bloqueos. Responde a esta pregunta en la tabla que se te proporciona.
- Comprueba si el protocolo OMT admite el plan  $P$  dando una traza donde se vean las marcas de tiempo de los elementos de datos durante la ejecución del plan. Supón que las cuatro variables tienen unas marcas de tiempo de lectura y escritura con un valor  $q$  que es anterior al inicio del plan. Responde a esta pregunta en la tabla que se te proporciona.

## 4. a) 0'4 puntos b) 0'4 puntos

- Supón un entorno concurrente controlado por el protocolo Multiversión en el que una transacción  $T$  ejecuta la instrucción  $w(x)$ . Supón también que hay tres versiones del elemento  $x$  (sean  $x_0, x_1$  y  $x_2$ ), que las marcas de tiempo son las que se indican más abajo y que  $t_i < t_j$  si  $i < j$ :

MT(T) = $t_8$		MT_L	MT_E
	$x_0$	$t_1$	$t_1$
	$x_1$	$t_9$	$t_2$
	$x_2$	$t_{10}$	$t_{10}$

- ¿Puede  $T$  ejecutar esa operación?  
¿Qué condición o condiciones se evalúan para permitir su ejecución?  
¿Qué se controla con esa condición?
- ¿Qué es un plan recuperable? Pon un ejemplo de plan concurrente recuperable que tenga lectura sucia y uno de plan concurrente no recuperable también con lectura sucia.

# SOLUCIONES 1<sup>er</sup> Parcial de Diseño y Gestión de Bases de Datos (26/10/2020)

1 a) Procesar correctamente una transacción significa que:

- a) todas las operaciones de la transacción se ejecutan con éxito y sus cambios (actualizaciones) quedan grabados permanentemente en la base de datos,
- o bien
- (b) la transacción no tiene ningún efecto en la base de datos.

1 b)

- Actualización: "Insertar un nuevo profesor con dni '0000', nombre 'Petra Pérez' y teléfono '1111'. Petra va a impartir docencia en la asignatura de código '11596' "
- Modo de comprobación de las restricciones:

Asignatura(cod\_asg: char(5), nombre: char(50), semestre: char(2))

CP:{cod\_asg} INMEDIATA-DIFERIBLE

VNN:{nombre,semestre} INMEDIATA-DIFERIBLE

Profesor(dni: char(9), nombre: char(80), teléfono: char(8))

CP:{dni} INMEDIATA-DIFERIBLE

VNN:{nombre} INMEDIATA-DIFERIBLE

Docencia(dni: char(9), cod\_asg: char(5))

CP:{dni,cod\_asg} INMEDIATA-DIFERIBLE

CAj:{dni}→Profesor(dni) DIFERIDA-DIFERIBLE

CAj:{cod\_asg}→Asignatura(cod\_asg) INMEDIATA-DIFERIBLE

Restricción general:

RG1: "Todo profesor debe impartir docencia de al menos una asignatura". DIFERIDA-DIFERIBLE

- Transacción:

INICIO:

INSERT INTO Profesor VALUES ('0000', 'Petra Pérez', '1111');

INSERT INTO Docencia VALUES ('0000','11596');

COMMIT;

2 a) De los puntos de control se puede prescindir cuando la estrategia es FORZAR ya que en este caso los cambios realizados por una transacción son bajados a disco cuando se confirma la transacción. Cuando se establece un punto de control, los cambios realizados por transacciones confirmadas son llevados a disco, en el caso de la estrategia forzar se sabe que, si la transacción está confirmada, estos cambios ya están en disco.

2b) Del diario no se puede prescindir nunca, ni siquiera con la estrategia DIFERIDA/FORZAR ya que, aunque el algoritmo en este caso es NO DESHACER/NO REHACER sería necesario en dos casos:

- Fallo del disco: sería necesario para rehacer sobre la copia de seguridad todas las transacciones confirmadas desde la fecha de la copia de seguridad.
- Para deshacer transacciones anuladas por el usuario (rollback) o por el sistema sin que haya habido ningún fallo. El diario se usará para rehacer los buffers de datos en memoria que pudieran haber sido modificados por la transacción anulada.

2 c)

- i) La recuperación dependerá de la estrategia de actualización:

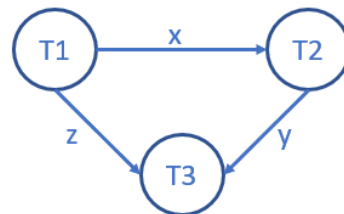
Actualización INMEDIATA-NO FORZAR→ Algoritmo DESHACER-REHACER							
T1	T2	T3	T4	T5	T6	T7	T8
Nada	Nada	Nada	Nada	Nada	Rehacer	Nada	Deshacer
Actualización INMEDIATA- FORZAR→ Algoritmo DESHACER-NO REHACER							
T1	T2	T3	T4	T5	T6	T7	T8
Nada	Nada	Nada	Nada	Nada	Nada	Nada	Deshacer

## SOLUCIONES 1<sup>er</sup> Parcial de Diseño y Gestión de Bases de Datos (26/10/2020)

Actualización DIFERIDA-NO FORZAR→ Algoritmo NO DESHACER-REHACER							
T1	T2	T3	T4	T5	T6	T7	T8
Nada	Nada	Nada	Nada	Nada	Rehacer	Nada	Nada
Actualización DIFERIDA- FORZAR→ Algoritmo NO DESHACER-NO REHACER							
T1	T2	T3	T4	T5	T6	T7	T8
Nada	Nada	Nada	Nada	Nada	Nada	Nada	Nada

- ii. Si se ha perdido la base de datos en disco, pero no el diario en disco, independientemente de la estrategia de actualización, el SGBD recuperaría la copia de seguridad del instante  $t_j$  y aplicaría los cambios realizados por las transacciones confirmadas antes del fallo y después de la copia de seguridad, es decir rehacería  $T_6$  usando para ello el diario en disco. Con el resto de las transacciones no habría que hacer nada ya que  $T_2$ ,  $T_5$ ,  $T_7$  y  $T_8$  no han dejado rastro en la copia de seguridad y  $T_1$ ,  $T_3$  y  $T_4$  tenían sus cambios en él (por la suposición que hemos hecho de que antes de hacer la copia de seguridad se han bajado todos los cambios realizados por transacciones confirmadas).
- iii. Si se ha perdido la base de datos en disco y también el diario en disco, independientemente de la estrategia de actualización, el SGBD recuperaría la copia de seguridad del instante  $t_j$  no pudiendo rehacer ninguna transacción de las confirmadas después de ese instante ya que no hay información que lo permita (el diario se ha perdido). Es decir, se perdería el efecto de  $T_6$ .

3 a) El grafo de serialización es el siguiente:



Como el grafo no tiene ciclos, el plan es serializable por conflictos.

El único plan en serie equivalente por conflictos es:  $T_1 \rightarrow T_2 \rightarrow T_3$

El plan en serie cronológico es  $T_1 \rightarrow T_2 \rightarrow T_3$  ya que en él las transacciones se procesan siguiendo el orden de sus marcas de tiempo en el plan concurrente y sí es equivalente por conflictos.

# SOLUCIONES 1<sup>er</sup> Parcial de Diseño y Gestión de Bases de Datos (26/10/2020)

3 b) El plan sí que es admitido por el protocolo de bloqueos B2F explícito como se puede ver a continuación (no es la única solución posible):

				Estado de bloqueo de los elementos							
				x		y		z		k	
	T <sub>1</sub>	T <sub>2</sub>	T <sub>3</sub>	BL	BE	BL	BE	BL	BE	BL	BE
t <sub>0</sub>	B_L(x)										
t <sub>1</sub>	r <sub>1</sub> (x)			T1							
t <sub>2</sub>	B_L(z)			T1				T1			
t <sub>3</sub>	B_L(k)			T1				T1		T1	
t <sub>4</sub>	D(x)							T1		T1	
t <sub>5</sub>		B_E(x)						T1		T1	
t <sub>6</sub>		r <sub>2</sub> (x)			T2			T1		T1	
t <sub>7</sub>		w <sub>2</sub> (x)			T2			T1		T1	
t <sub>8</sub>		B_L(y)			T2			T1		T1	
t <sub>9</sub>		r <sub>2</sub> (y)			T2	T2		T1		T1	
t <sub>10</sub>	r <sub>1</sub> (z)				T2	T2		T1		T1	
t <sub>11</sub>	D(z)				T2	T2				T1	
t <sub>12</sub>		B_L(k)			T2	T2				T1	
t <sub>13</sub>		D(y)			T2					T1,T2	
t <sub>14</sub>			B_E(y)		T2					T1,T2	
t <sub>15</sub>			w <sub>3</sub> (y)		T2		T3			T1,T2	
t <sub>16</sub>		r <sub>2</sub> (k)			T2		T3			T1,T2	
t <sub>17</sub>		D(x)					T3			T1,T2	
t <sub>18</sub>		D(k)					T3			T1	
t <sub>19</sub>		c <sub>2</sub>					T3			T1	
t <sub>20</sub>			B_L(k)				T3			T1	
t <sub>21</sub>			r <sub>3</sub> (k)				T3			T1,T3	
t <sub>22</sub>	r <sub>1</sub> (k)						T3			T1,T3	
t <sub>23</sub>	D(k)						T3			T3	
t <sub>24</sub>	c <sub>1</sub>						T3			T3	
t <sub>25</sub>			B_E(z)				T3			T3	
t <sub>26</sub>			w <sub>3</sub> (z)				T3		T3	T3	
t <sub>27</sub>			D(y)				T3		T3	T3	
t <sub>28</sub>			D(k)						T3	T3	
t <sub>29</sub>			D(z)								
t <sub>30</sub>			c <sub>3</sub>								

Comentarios:

- B\_L(x): bloqueo de lectura sobre x.
- B\_E(x): bloqueo de escritura sobre x.
- D(x): desbloqueo de x.
- El plan presentado que incluye los bloqueos no es el único posible (por ejemplo, el desbloqueo de z en t<sub>10</sub>, podría hacerse más tarde siempre que fuera antes de la petición de bloqueo de z por T<sub>3</sub>).

Este plan es equivalente al plan en serie T<sub>1</sub> -> T<sub>2</sub> -> T<sub>3</sub> ya que, en ambos planes, los pares de operaciones en conflicto aparecen en el mismo orden. Éste es además el plan en serie cronológico.

# SOLUCIONES 1<sup>er</sup> Parcial de Diseño y Gestión de Bases de Datos (26/10/2020)

3 c)  $MT_{(T_1)} = t_0$ ,  $MT_{(T_2)} = t_1$ ,  $MT_{(T_3)} = t_5$

Dado que  $T_1$ ,  $T_2$ , y  $T_3$  no han empezado antes de  $q$ , las marcas de lectura y escritura de  $x$ ,  $y$ ,  $z$  y  $k$  son, con certeza, menores que  $q$ . Las celdas sombreadas indican un cambio de valor respecto al instante de tiempo anterior.

Protocolo OMT				x		y		z		k	
tiempo	$T_1$	$T_2$	$T_3$	MT_L	MT_E	MT_L	MT_E	MT_L	MT_E	MT_L	MT_E
				q	q	q	q	q	q	q	q
$t_0$	$r_1(x)$			$t_0$	q	q	q	q	q	q	q
$t_1$		$r_2(x)$		$t_1$	q	q	q	q	q	q	q
$t_2$		$w_2(x)$		$t_1$	$t_1$	q	q	q	q	q	q
$t_3$		$r_2(y)$		$t_1$	$t_1$	$t_1$	q	q	q	q	q
$t_4$	$r_1(z)$			$t_1$	$t_1$	$t_1$	q	$t_0$	q	q	q
$t_5$			$w_3(y)$	$t_1$	$t_1$	$t_1$	$t_5$	$t_0$	q	q	q
$t_6$		$r_2(k)$		$t_1$	$t_1$	$t_1$	$t_5$	$t_0$	q	$t_1$	q
$t_7$		$c_2$		$t_1$	$t_1$	$t_1$	$t_5$	$t_0$	q	$t_1$	q
$t_8$			$r_3(k)$	$t_1$	$t_1$	$t_1$	$t_5$	$t_0$	q	$t_5$	q
$t_9$	$r_1(k)$			$t_1$	$t_1$	$t_1$	$t_5$	$t_0$	q	$t_5$	q
$t_{10}$	$c_1$			$t_1$	$t_1$	$t_1$	$t_5$	$t_0$	q	$t_5$	q
$t_{11}$			$w_3(z)$	$t_1$	$t_1$	$t_1$	$t_5$	$t_0$	$t_5$	$t_5$	q
$t_{15}$			$c_3$	$t_1$	$t_1$	$t_1$	$t_5$	$t_0$	$t_5$	$t_2$	q

El plan es aceptado por el protocolo OMT.

4)

- a) De las tres versiones de  $x$  que hay se busca la que tenga mayor marca de tiempo de escritura que cumpla que:  $MT_E(x_i) \leq MT(T)$ . En este caso esa versión es  $x_1$ . Dado que  $MT_L(x_1) > MT(T)$ , la operación es rechazada ya que  $x_1$  ha sido leída por una transacción más joven que  $T$ . Con esta condición se controla que la versión "anterior" al elemento de datos  $x$  no ha sido leída por ninguna transacción  $T^*$  posterior (cronológicamente) a  $T$ .
- b) Un plan recuperable es un plan en el que una transacción  $T$  no se puede confirmar antes de que se confirmen todas las transacciones que han actualizado (escrito) un elemento de datos leído por  $T$ .

$P = r_1(x), r_2(y), w_1(x), r_2(x), c_1, c_2$

$P' = r_1(x), r_2(y), w_1(x), r_2(x), c_2, c_1$

$P$  es un plan recuperable ya que  $T_2$  que lee un elemento de datos ( $x$ ) escrito por  $T_1$  pero se confirma después de la confirmación de  $T_1$ . Hay lectura sucia en la operación  $r_2(x)$  ya que el elemento de datos  $x$  ha sido escrito por  $T_1$  y aún no ha sido confirmada cuando  $T_2$  lee ese elemento.

$P'$  es un plan no recuperable ya que  $T_2$  que lee un elemento de datos escrito por  $T_1$  y se confirma antes de la confirmación de  $T_1$ .