

HECHOS POR PROFESOR

2. Indica el mecanismo de recuperación de un sistema ante las siguientes situaciones:

Si ocurre un fallo en el punto (2) . Todo lo que aparece por debajo de l punto 2 no pasas lo que hay después. Durante un fallo de recuperación se garantizar que todos los datos se vuelcan a disco. En T2 que empieza buscamos hacia abajo pero en el espacio entre la tabla modificaciones y el 2 no hay un commit. Luego no terminó entonces ¿Qué hacemos con la transacción? Tenemos que deshacer la transacción. Hacer una operación UNDO. Asigna a todos los átomos el valor antiguo de la transacción 2.

- Fallo en (2)

- Fallo en (1) con savepoint en (2)

El punto 2 hace un savepoint en estado. Esto significa que tenemos algo relacionado con un checkpoint que es: El volcado obligatorio al disco. Se fuerza su volcado al disco de forma completamente obligatoria. En un checkpoint se escribe el redolog al disco volcar las transacciones al disco escribir en el redolog diciendo que hemos hecho un savepoint y volver a meterlo a disco. El savepoint garantiza que se han volcado los bloques de la T1 luego no tenemos que volver a escribirlo. La T2 empieza pero su finalización ocurre detrás del savepoint. No podemos garantizar la escritura del T2 así que tenemos que hacer un REDO de T2. Pero en el caso de T4 tenemos que hacer un UNDO por que pasa después. LA CLAVE ES EL COMMIT . Si empiezan y terminan después del save point se jod del todo lo que termine después del save point REDO todo lo que empiece pero no termine en ningún lado hay que hacerlo un UNDO.

HECHOS POR MÍ

6. Indica cómo sería la ejecución de las siguientes transacciones sobre el átomo a según el algoritmo de ordenación parcial modificado, teniendo en cuenta que cada escritura multiplica por 2 el valor anterior y que $a=2$ inicialmente. lee (T2, a), escribe(T3, a), lee (T4, a), escribe(T2, a), escribe(T4, a), escribe(T5, a), lee(T6, a), escribe(T7, a), escribe(T6, a)

Si abortamos metemos esa transacción al final.

$RR(A) = 0$ $WR(A) = 0$

Lectura contra escritura. Escritura contra lectura y escritura

1.Lee(T2,a) → Se lee sin problema. $RR(A) = 2$. T2: $A = 2$

2.Escribe(T3,a) → Se escribe sin problemas $RR(A) = 2$ $WR(A)=3$. T3: $A = 4$ (valor que queramos realmente)

3.Lee(T4,a) → Se lee sin problemas. $RR(A) = 4$ $WR(A) = 3$. T4: $A = 4$

4.Escribe(T2,a) → Falla porque $RR(A)=4 \leq 2$. Aborta la transacción 2 (el 4 intenta escribir pero hemos leído anteriormente, como el átomo de lectura anterior es la 4, la 2 no puede escribir, el plan debe de reescribirse poniendo de nuevo todas las sentencias que a).

5.Escribe(T4,a) → Se escribe sin problemas. $WR(A) = 4$. $RR(A) = 4$. $A = 8$

6.Escribe(T5,a) → Se escribe sin problemas. $WR(A) = 5$. $A = 16$

7.Lee(T6,a) → Se lee sin problemas $RR(A) = 6$. $A = 16$.

8.Escribe(T7,a) → Se escribe sin problemas $WR(A) = 7$. $A = 32$ (escribe 32 por que he querido yo, al T7 no leer A da igual)

9.Escribe(T6,a) → Cuidado: no hace nada, no escribimos en los controladores. No aborta la transacción. Se debe a la particularidad del algoritmo parcial modificado.

— Metidos a posterior —

No existe inicialmente

10. Lee(T8,a) ningún problema (Es T8 por que las quitamos y las ponemos al final luego su marca de tiempo es superior al de las demás)

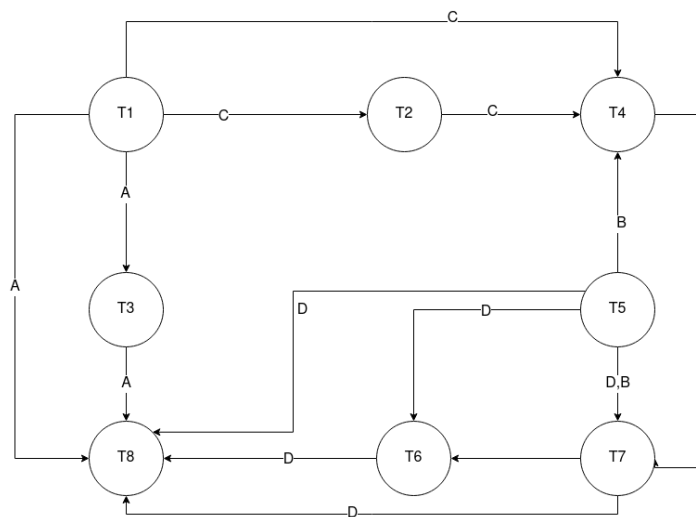
11. Escribe(T8,a)

El orden de ejecución de las transacciones sería : T3,T4,T5,T7,T6,T2 → (T8)

7. Para el siguiente plan de ejecución, construye el grafo de precedencia e indica si la ejecución es o no serializable. En caso afirmativo, dar alguna secuencia de transacciones equivalente.
Componer el grafo de forma ordenada junto con leemos las transacciones.

Los grafos de precedencia dependen del acceso no de si es lectura o escritura. Así que hay que entender sólo al acceso.

lee(T1,C),
escribe(T1,A),
lee(T2,C),
escribe(T2,C),
lee(T3,A),
escribe(T1,C),
lee(T4,C),
lee(T5,B),
escribe(T4,B),
escribe(T5,D),
escribe(T8,A),
lee(T7,D),
lee(T6,D),
escribe(T7,B),
lee(T8,D)



Es serializable ya que no existe ningún ciclo. Una secuencia de transacciones equivalentes sería empezar por T5.

Secuencia de transacciones: T1, T3, T2, T5, T4, T7, T6, T8

La secuencia no es única hay más opciones.