MATHEUS PEIXOTO RIBEIRO VIEIRA - 22.1.4104

VITOR OLIVEIRA DINIZ - 22.1.4107

Lista de Exercícios III – Recuperação de Falhas

5) O que significam os termos steal / no-steal e force / no-force em relação ao gerenciamento dos buffers de memória para processamento de transações?

Os termos indicam a maneira pela qual o SGBD trabalhará a escrita dos dados em disco com relação ao commit

- Steal: A página que foi modificada pode ser escrita em disco antes mesmo que o commit ocorra, como se “roubasse” os dados para o disco
- No-Steal: Indica que a página só pode ser escrita em disco após a realização do commit
- Force: Após o commit, a página é imediatamente escrita em disco
- No-Force: Mesmo ocorrendo o commit, a página pode, ou não, ser escrita em disco de forma imediata

6) O que são checkpoints e porque são importantes?

Checkpoints são momentos onde as modificações confirmadas em buffers são forçadas a serem salvas em disco. Estes salvamentos são feitos de forma períodica e automática pelo próprio SGBD, gerando um novo registro no log indicando o acontecimento de um checkpoint.

Supondo um grande escalonamento entre diferentes transações, caso o sistema venha a falhar, todas aquelas que tiveram o commit antes do checkpoint não precisarão ser refeitas, economizando tempo e processamento.

8) Discuta a técnica de recuperação com atualização postergada. Porque é chamada de método NO UNDO / REDO?

Durante a execução da transação, as modificações somente são registradas no log e na cache, e só se escreve em disco quando ocorre o commit (no steal), mas isso não precisa ser de imediato.

Caso uma falha de sistema ocorra antes do commit, nada precisará ser desfeito, pois o banco de dados em si não foi modificado, por isso a técnica é NO UNDO.

Quando a falha ocorre, as operações que possuem um write_item precisam ser refeitas, dessa forma, o log é percorrido na ordem normal executando-as, por isso o método é REDO.

9) Discuta a técnica de recuperação com atualização imediata. Porque pode ser chamada de método UNDO / NO-REDO ou método UNDO / REDO?

Na transação imediata, as modificações são permitidas a serem gravadas em banco diretamente, sem a necessidade de esperar pelo commit (steal), sendo que, antes de serem escritas fisicamente, elas devem ser escritas no log.

O sistema mantém duas listas de transações, uma com transações confirmadas desde o último checkpoint (lista de commit) e uma com as transações ativas (lista ativa).

Caso o algoritmo não permita que modificações no disco sejam feitas após o commit, então ele deverá ser do tipo UNDO/NO-REDO, pois caso ocorra uma falha, já sabemos que tudo está no banco, então nada precisará ser refeito, somente desfeito.

Caso o algoritmo permita que as modificações no disco sejam feitas após o commit, então ele deverá ser do tipo UNDO/REDO, pois caso ocorra uma falha, não se sabe se todas as operações estarão em disco, logo deverão ser refeitas, e algumas operações precisarão ser desfeitas

Quando ocorre uma falha, o UNDO irá desfazer as operações de escrita das transações ativas e o REDO irá refazer as operações na ordem que foram gravadas no log.

11) O log abaixo apresenta uma sequência de operações, relativas ao escalonamento de 4 transações, até o momento de uma falha.

[start_transaction, T1]
[read_item, T1, A]
[read_item, T1, D]
[write_item, T1, D, 20, 25]
[commit, T1]
[checkpoint]
[start_transaction, T2]
[read_item, T2, B]
[write_item, T2, B, 12, 18]
[start_transaction, T4]
[read_item, T4, D]
[write_item, T4, D, 25, 15]
[start_transaction, T3]
[write_item, T3, C, 30, 40]
[read_item, T4, A]
[write_item, T4, A, 30, 20]
[commit, T4]
[read_item, T2, D]
[write_item, T2, D, 15, 25]
<<< Falha >>>

a) Descreva o processo de recuperação de falha por meio do protocolo de atualização imediata com checkpoint, especificando quais transações devem ser revertidas (rollback), quais operações do log devem ser refeitas e quais devem ser desfeitas, e se ocorre algum rollback em cascata.

A transação 1 tem o seu commit antes do checkpoint, então nada deverá ser feito, pois suas alterações já foram escritas no banco.

A transação 2 não foi confirmada e, como o seu início está após o checkpoint, ela sofrerá um rollback e deverá ser desfeita com o UNDO e será re-submetida. O mesmo ocorre para a transação 3. Dessa forma, as operações a serem desfeitas serão as apresentadas a seguir na seguinte ordem (Ordem inversa do que aparecem no log):

- [write_item, T2, D, 15, 25]
- [write_item, T2, B, 12, 18]
- [write_item, T3, C, 30, 40]

A transação 4 tem seu início e fim com o commit entre o checkpoint e a falha, dessa forma deverá ser feito um REDO de suas operações no disco. Dessa forma, as operações que deverão ser refeitas na ordem que aparecem no log serão:

- [write_item, T4, D, 25, 15]
- [write_item, T4, A, 30, 20]

Como naão ocorreram leituras sujas, não ocorrem rollbacks em cascata.

b) Suponha que o protocolo usado seja o de atualização postergada. Mostre como seria o log correspondente. A partir de tal log modificado, descreva o processo de recuperação de falha, especificando quais transações devem ser revertidas (rollback), quais operações do log devem ser refeitas e quais devem ser desfeitas, e se ocorre algum rollback em cascata.

[start_transaction, T1]
[read_item, T1, A]
[read_item, T1, D]
[write_item, T1, D, 25]
[commit, T1]
[checkpoint]
[start_transaction, T2]
[read_item, T2, B]
[write_item, T2, B, 18]
[start_transaction, T4]
[read_item, T4, D]
[write_item, T4, D, 15]
[start_transaction, T3]
[write_item, T3, C, 40]
[read_item, T4, A]
[write_item, T4, A, 20]
[commit, T4]
[read_item, T2, D]
[write_item, T2, D, 25]
<<< Falha >>>

Como a transação 1 atingiu o commit e, logo em seguida, ocorreu o checkpoint, suas modificações serão salvas no disco. Dessa forma, nada precisará ser feito.

A transação T2 e T3 iniciaram e falharam sem o commit. Dessa forma, nenhuma alteração no banco será feita, pois nada foi, de fato, escrito.

A transação 4 possui um commit entre o checkpoint e a falha, dessa forma, os seus comando de write_item precisarão serem refeitos pelo REDO na ordem em que aparecem no log.

Não ocorreram rollbacks em cascata.