

## **BCC441 - Lista de Exercícios III: Recuperação a falhas**

**Alunos:** Mateus Lana e Thiago Santana

**Matrículas:** 15.1.4340 e 15.1.4313

### **1) Discuta os diferentes tipos de falhas em transações: falha catastrófica e falha não catastrófica.**

As possíveis falhas são categorizadas em:

**Falha catastrófica:** recarrega a cópia (backup) mais recente do banco de dados e reconstrói um estado consistente do mesmo refazendo operações das transações já confirmadas a partir do arquivo de log.

**Falha não catastrófica:** reverte as alterações que causaram a inconsistência desfazendo operações de transações não confirmadas ou refazendo operações de transações confirmadas.

### **6) O que são checkpoints e porque são importantes ?**

Periodicamente, o sistema operacional força a gravação em disco de todos os buffers do SGBD que tenham sido alterados. Quando isso ocorre, um registro chamado checkpoint (ponto de verificação) é gravado em log. Ou seja, checkpoints são pontos de gravação no banco de dados.

Eles são importantes pois todas as transações que registraram um [commit, T] no log antes de um checkpoint não precisam ser refeitas em caso de colapso do sistema.

### **8) Discuta a técnica de recuperação com atualização postergada. Por que é chamada de método NO-UNDO/REDO ?**

A técnica de recuperação baseada na atualização postergada adia qualquer atualização no banco de dados até que a transação complete sua execução com êxito e atinja seu ponto de confirmação (abordagem no-steal).

Durante a execução da transação, as atualizações são registradas somente no log e nos buffers do cache. Se uma transação falha antes de atingir seu ponto de confirmação, não há necessidade de desfazer nenhuma operação, porque a transação não afetou, de modo algum, o banco de dados armazenado em disco. Depois que a transação atinge seu ponto de confirmação e o log é forçosamente gravado no disco, as atualizações são registradas no banco de dados (registro adiantado em log).

É chamada de NO-UNDO/REDO pois não é necessário desfazer(UNDO) porque o banco de dados não é atualizado antes do commit. No entanto, pode ser necessário refazer(REDO) se o sistema falhar depois do commit, porém antes que todas as suas alterações tenham sido registradas no banco de dados.

**11) O log apresentado na questão demonstra uma sequência de operações, relativas ao escalonamento de 4 transações, até o momento de uma 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.**

Utilizando o protocolo de atualização imediata com algoritmo de recuperação UNDO/REDO, não haverá necessidade de refazer a transação T1 visto que esta já foi confirmada antes do checkpoint.

As operações de escrita da transação T4 serão refeitas respeitando a ordem em que foram executadas na transação.

As operações de escrita das transações T2 e T3 serão desfeitas na ordem inversa em relação a ordem que foram executadas nas transações.

Não haverá rollback 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.**

Se o protocolo utilizado fosse o de atualização postergada, o log correspondente, a cada operação de escrita, não armazenaria o valor antigo do item a ser escrito, e sim somente o novo valor. Isso se deve ao fato deste protocolo não desfazer as transações, ele somente refaz.

Utilizando o protocolo de atualização postergada com algoritmo de recuperação NO-UNDO/REDO, não haverá necessidade de refazer a transação T1 visto que esta já foi confirmada antes do checkpoint.

As operações de escrita da transação T4 serão refeitas respeitando a ordem em que foram executadas na transação.

As transações T2 e T3 são ignoradas (nenhuma de suas operações foi registrada no banco de dados) e devem ser resubmetidas.

Não haverá rollback em cascata.