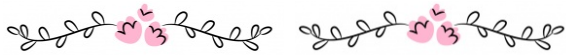


Processamento de Transações

Introdução

- ↳ Transação é uma unidade de execução de um programa que altera os dados de um BD
- ↳ Sequencia de operações de escrita e leitura
- ↳ É delimitada por declarações da forma início da transação e fim da transação



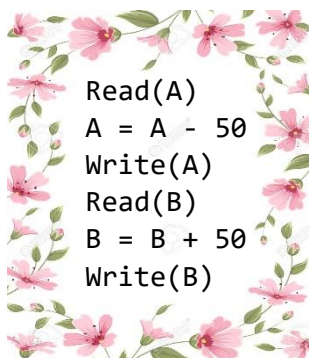
Observação:

- Os dados do BD estão armazenados em memória secundária
- As operações que não sejam leitura/escrita não apresentam efeito para o BD

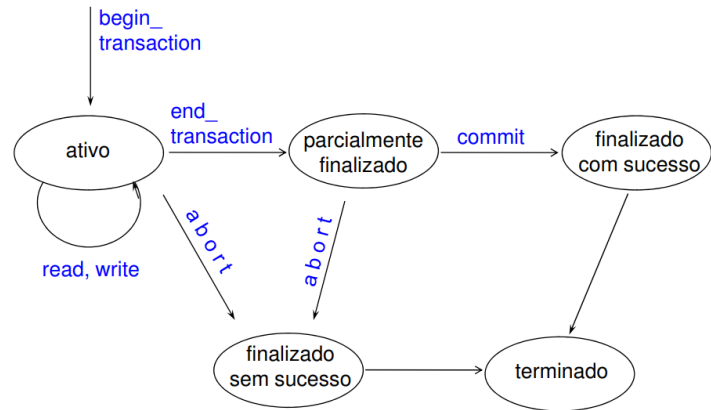


Operações das Transações

- ↳ **Begin_Transaction**: início da execução de uma transação
- ↳ **Read** ou **Write**: operações de leitura e escrita nos dados do BD
- ↳ **End_Transaction**: Final da execução de uma transação. Deve ser verificado se executará commit ou abort
- ↳ **Commit**:
 - Transação finalizada com sucesso
 - Alterações tornam-se permanentes
- ↳ **Rollback**:
 - Transação finalizada sem sucesso
 - Alterações são descartadas



Transição de estado



ACID

- ↳ **Atomicidade**: todas as operações devem ser efetivadas, caso contrário os dados são revertidos.
- ↳ **Consistência**: A transação cria um novo estado válido dos dados ou em caso de falha retorna todos os dados ao seu estado antes que a transação foi iniciada
- ↳ **Isolamento**: Uma transação em andamento, mas ainda não validada deve permanecer isolada de qualquer outra operação, ou seja, garantimos que a transação não será interferida por nenhuma outra transação concorrente.
- ↳ **Durabilidade**: Dados validados são registrados pelo sistema de tal forma que mesmo no caso de uma falha e/ou reinício do sistema, os dados estão disponíveis em seu estado correto.

Controle de Concorrência

↪ Execução Serial (sequencial):

- diversas transações executadas em sequência deixa a base de dados em estado correto e consistente
- estado inicial correto e consistente ⇒ estado final correto e consistente

↪ Execução Intercalada:

- comandos de diversas transações são intercalados
- pode levar a inconsistências
- Toda execução serial é consistente
- Mas uma execução intercalada só é consistente se for igual ao resultado de uma execução em sequência (em ordem conhecida)
- Esta execução é dita serializável

Níveis de Isolamento

Existem problemas da execução intercalada:

↪ **Dirty Read:** Suponhamos que a transação "A" modifique algum campo da tabela, porém que ainda não o tenha commitado. Se uma transação "B" efetua um select neste campo e vê o valor modificado pela transação "A" sem ter o commit efetuado, essa é uma leitura suja

↪ **Leitura não repetitiva:** Ocorre quando um select(leitura) reproduz resultados diferentes quando ela é repetida posteriormente na mesma transação. Ou seja, tentou ler um dado que foi alterado, impedindo que uma operação consistente fosse concluída

1. transação T lê um dado
2. esse dado é modificado por uma transação T' que começou depois de T
3. T é efetivada
4. se T' tentar reler o mesmo dado, obterá valores diferentes (nonrepeatable read)

↪ **Leitura Fantasma:** teve alterado o conjunto de tuplas envolvidas em uma seleção fazendo com que uma operação tenha resultados diferentes em momentos diferentes da transação

1. transação T lê um conjunto de tuplas que atendam a uma condição de consulta
2. transação T' insere/remove/atualiza uma tupla que atenderia a essa condição e é efetivada
3. se T refizer a mesma consulta, obterá um conjunto diferente de tuplas (phantom read)



↪ **Repeatable read:** lê valores diferentes de um mesmo dado que ainda está lá, mas foi alterado.

↪ **Read Committed:** Permite que a transação leia/manipule os dados já commitados por outras transações. Caso alguma transação tenha alterado algum dado porém sem efetuar um commit, estes não serão vistos.

↪ **Read Uncommitted:** Este nível permite que uma transação possa ver e manipular valores não commitados por outras transações, ficando aberto para leituras sujas e Leitura não repetitiva, facilitando também casos de Leituras fantasmas.

↪ **Serializable:** Semelhante ao Repeatable Read, porém com a restrição adicional de que as linhas selecionadas por uma transação não podem ser alteradas ou lidas por outra transação, até que a primeira transação seja concluída

