

Transação

Transação é um bloco composto de uma unidade lógica de trabalho, ou seja, todas as instruções que compõem o bloco são efetivadas ou nenhuma delas é efetivada.

É um conjunto de procedimentos executados num banco de dados, que para o usuário é visto como uma única ação.

Do ponto de vista do SGBD, uma transação é uma sequência de operações que são tratadas como um bloco único e indivisível (atômico) no que se refere à sua recuperação.

Esse conceito garante: consistência e integridade dos dados.

CATEGORIA SQL DTL

DTL: Linguagem de Transação de Dados/
Data Transaction Language

ou

TPL: Linguagem de Processamento de Transações

ou ainda

TCL: Linguagem de Controle de Transações

Transação Monousuária

- ✓ Permite agrupar diversas operações em uma só.
- ✓ Se uma operação falhar, as demais são canceladas.

Comandos básicos:

- **BEGIN** – início (start no MySQL)
- **COMMIT** - final
- **ROLLBACK** - desfaz a transação corrente

Aumenta a velocidade!

Transações se dão sem acesso ao disco para atualizar o banco de dados (update). As mudanças são armazenadas na memória RAM, até o Commit, quando são gravadas. Assim, há acesso ao disco em uma única vez.

Esquema do Código SQL

```
START TRANSACTION
```

```
UPDATE CONTA A;
```

```
UPDATE CONTA B;
```

```
IF TRANSAÇÃO OK
```

```
    THEN COMMIT;
```

```
    ELSE ROLLBACK;
```

```
END IF;
```

Término da Transação

É indicado pelos comandos:

Commit: Termina com sucesso uma a transação.

Rollback: Termina uma transação indicando que houve um erro.

Se uma transação terminar com **commit**, todas as modificações realizadas dentro dela, serão efetivadas.

Se uma transação terminar com **rollback**, todas as modificações realizadas dentro dela, serão desfeitas. O banco de dados ficará no mesmo estado que estava antes da transação ser iniciada.

O comando **rollback** pode ser emitido explicitamente pelo programador ou automaticamente pelo SGBD quando ocorre um erro.

Garantia de Isolamento por Transação

COMMIT faz as atualizações efetuadas na transação se tornarem visíveis a outras transações e com garantia de não serem canceladas (durabilidade).

Por outro lado, ROLLBACK faz com que todas as atualizações feitas durante a transação sejam canceladas (desfeitas) como se a transação nunca tivesse existido (sido executada).

Nenhuma outra transação influenciará no resultado da transação em processo.

START

.....

SAVEPOINT P1;

.....

ROLLBACK TO SAVEPOINT P1;

Ponto de Salvamento

Ao contrário da instrução **COMMIT TRANSACTION**, a instrução **ROLLBACK TRANSACTION** termina uma transação que por algum motivo não pode ser completada com sucesso.

A instrução **SAVE TRANSACTION** estabelece um ponto de salvamento na transação. Este ponto é marcado de forma que todas as atualizações seguintes podem ser canceladas sem o cancelamento de toda a transação.

Propriedade das Transações

As transações devem ser executadas pelos SGBDs de maneira a evitar que problemas ocorram. As ações que compõem uma transação devem possuir um conjunto de propriedades que é normalmente referido como propriedades ACID:

- Atomicidade
- Consistência
- Isolamento
- Durabilidade

Atomicidade

Também chamado de Princípio do "Tudo ou Nada".

Evita que falha ocorridas, possam deixar o banco de dados inconsistentes.

Uma transação não pode ser executada pela metade, ou é executada por inteiro ou é cancelada por inteiro, retornando ao estado anterior ao seu início.

Consistência

Uma transação só é executada, se o estado do banco de dados permanecer consistente após o seu término.

Possui dois aspectos: A consistência do banco dados e a consistência da própria transação.

Uma transação deve ter uma programação correta e suas ações não devem resultar em violações das restrições de integridade definidas para o banco de dados.

Isolamento

As transações são independentes.

Em execuções concorrentes, uma transação não pode gerar um estado inconsistente, com a intercalação de diversas transações relacionadas.

Significa que, mesmo no caso de transações executadas concorrentemente, o resultado final é igual ao obtido com a execução isolada de cada uma delas.

Graus de Isolamento

O SQL Server implementa protocolos de controle de concorrência usando bloqueio.

O protocolo utilizado pode ser configurado pelo usuário de acordo com suas necessidades. O usuário pode, por exemplo, indicar o grau de isolamento desejado entre transações, definindo como será feito o bloqueio dos dados:

```
SET TRANSACTION ISOLATION LEVEL { SERIALIZABLE | REPEATABLE  
READ | READ COMMITTED | READ UNCOMMITTED }
```

Graus de isolamento também podem ser definidos para cada comando de uma transação, usando a cláusula WITH ao final do comando. Os valores aceitos são:

```
<comando SQL> WITH( { SERIALIZABLE | REPEATABLE | READ  
READCOMMITTED | READUNCOMMITTED } )
```

GRAUS DE ISOLAMENTO

SERIALIZABLE: PERMITE APENAS ESCALAS SERIALIZÁVEIS SEREM EXECUTADAS. BLOQUEIA OS DADOS ATÉ O FINAL DA TRANSAÇÃO E IMPEDE A INSERÇÃO DE NOVAS TUPLAS NAS TABELAS EM USO. MODO MAIS RESTRITO, QUE RESULTA EM MENOR CONCORRÊNCIA.

REPEATABLE READ: BLOQUEIA OS DADOS ATÉ O FINAL DA TRANSAÇÃO. GARANTE QUE OS VALORES DOS DADOS ACESSADOS NÃO SEJAM MODIFICADOS POR OUTRAS TRANSAÇÕES, MAS PERMITE INSERIR TUPLAS.

READ COMMITTED: OS VALORES DOS DADOS ACESSADOS NÃO PODEM SER MODIFICADOS POR OUTRAS TRANSAÇÕES. NÃO HÁ GARANTIA DE VALORES EM UMA PRÓXIMA LEITURA. OPÇÃO DEFAULT.

READ UNCOMMITTED: NÃO USA BLOQUEIO (outro usuário vai ler sem levar em conta a transação). É O MODO MENOS RESTRITO. DEVE SER USADO QUANDO NÃO SE TEM NENHUM COMPROMISSO COM CONSISTÊNCIA.

Durabilidade

A durabilidade garante, por algum mecanismo, a recuperação das informações perdidas.

Significa que o resultado de uma transação, caso ela seja concluída com sucesso, deve ser **persistente**. Mesmo se depois houver falha no sistema computacional envolvido.

Quando, em um banco de dados, ocorrer falha após a execução com sucesso de uma transação, é garantida a possibilidade de recuperação das informações e da integridade dos dados.

Abrangência

As transações são mais usuais com instruções **DML**, mas também funcionam com instruções **DDL**.

Elas ocorrem em processamentos isolados (monousuário) e concorrentes (multiusuário).

Observações

- Não confundir transação de processo com transação de Banco de Dados.
- Uma transação de BD ocorre sempre em tempo real, sem paradas.
- As transações isoladas precisam existir para manter a integridade do BD, independente de um DB monousuário ou multi.