CENTRO PAULA SOUZA

FATEC MAUÁ

Tecnólogo em Desenvolvimento de Software Multiplataforma

JOÃO PEDRO SASSI GRANADO

ACID – CONCEITOS DE DB

Mauá

2023

JOÃO PEDRO SASSI GRANADO

ACID – CONCEITOS DE DB

Trabalho sobre os conceitos de ACID apresentado no Tecnólogo de Desenvolvimento de Software Multiplataforma da Fatec Mauá, orientado pelo professor Prof. Carlos Alberto Vaz da disciplina Banco de Dados – Relacional.

Mauá

2023

ÍNDICE DE ILUSTRAÇÕES

[Figura 1: Exemplo Database 4](#_Toc137414433)

[Figura 2: Exemplo Write-Ahead Logging 5](#_Toc137414434)

[Figura 3: Exemplo Locking 7](#_Toc137414435)

[Figura 4: Exemplo ACID 9](#_Toc137414436)

SUMÁRIO

[1. INTRODUÇÃO 4](#_Toc137414490)

[2. ATOMICIDADE 5](#_Toc137414491)

[3. CONSISTÊNCIA 6](#_Toc137414492)

[4. ISOLAMENTO 7](#_Toc137414493)

[5. DURABILIDADE 8](#_Toc137414494)

[6. CONCLUSÃO 9](#_Toc137414495)

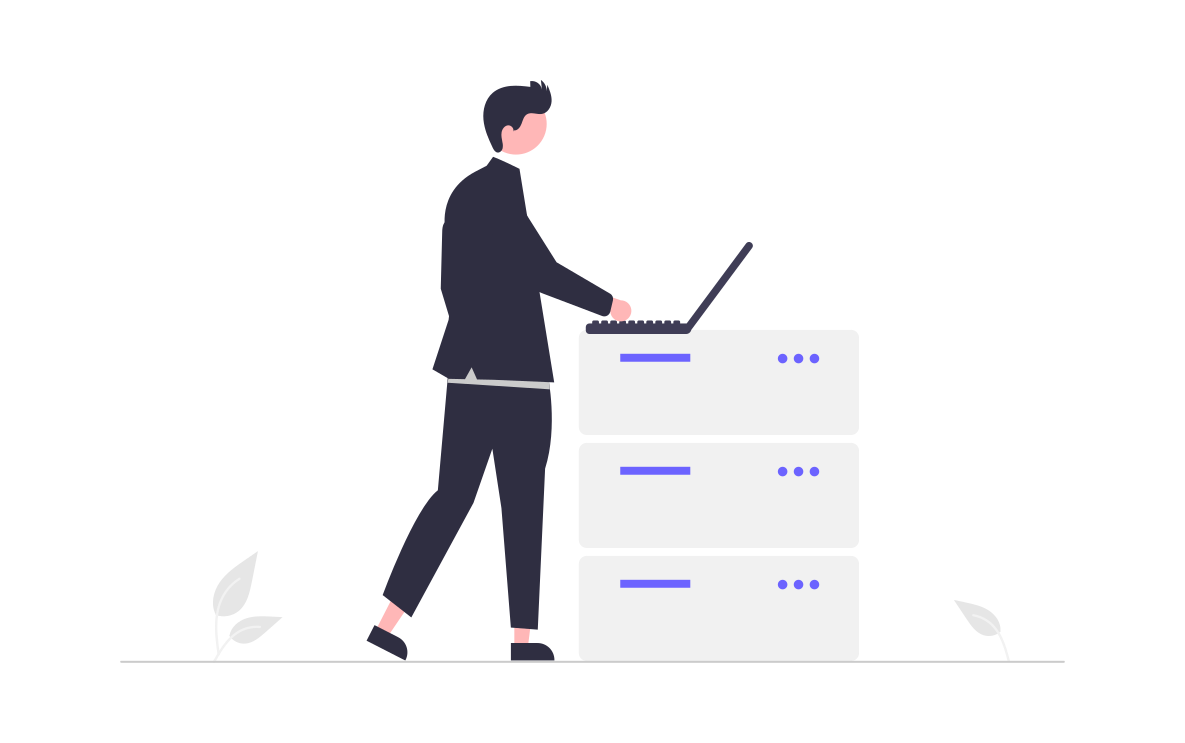
[BIBLIOGRAFIA 10](#_Toc137414496)

# INTRODUÇÃO

ACID é um acrônimo que representa quatro propriedades que garantem a confiabilidade e a consistência das transações em um banco de dados: atomicidade, consistência, isolamento e durabilidade.

Essas propriedades são essenciais para evitar erros, inconsistências e perdas de dados que podem comprometer a integridade do banco de dados. Neste trabalho, irei explicar cada uma dessas propriedades e como elas são implementadas em diferentes sistemas de gerenciamento de banco de dados, ilustrando com exemplos práticos.

Figura : Exemplo Database



Fonte: Portal Undraw, 2023

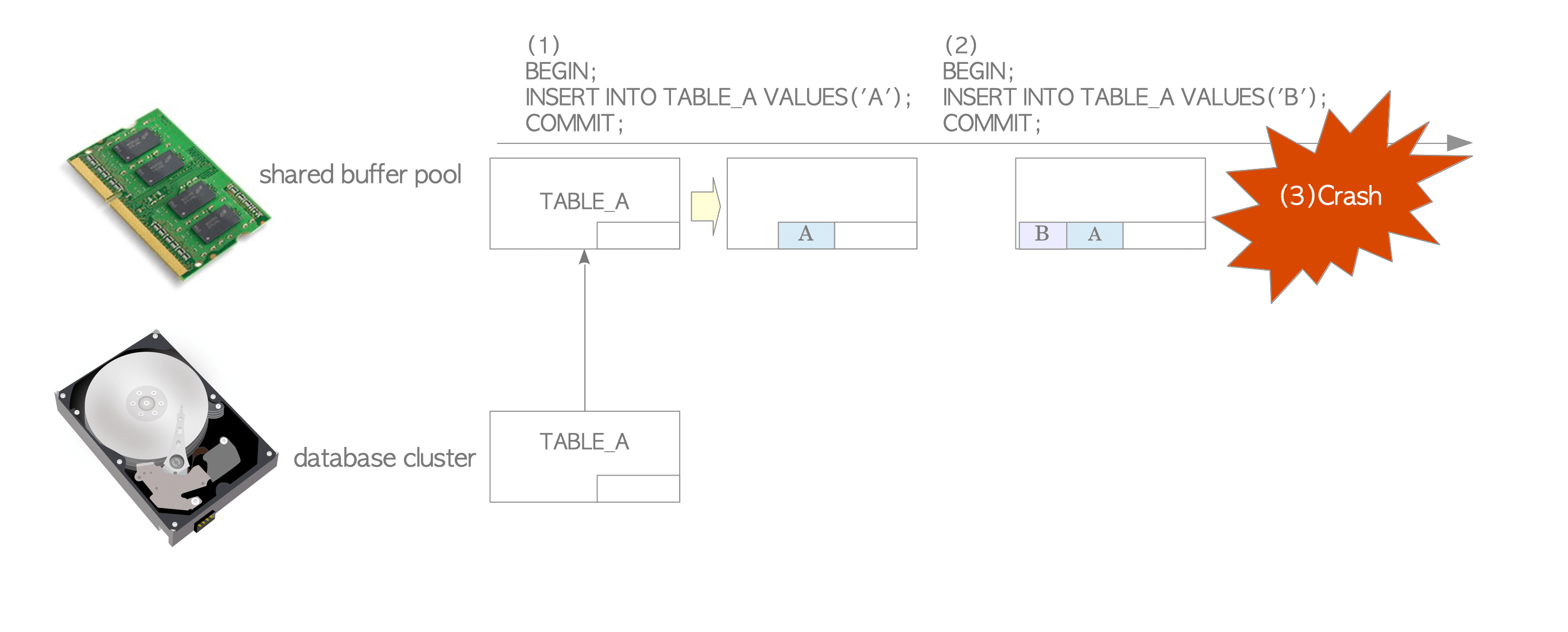
# ATOMICIDADE

Atomicidade significa que uma transação é uma unidade indivisível de operação, ou seja, ou ela é executada por completo ou não é executada. Isso evita que uma transação parcialmente concluída deixe o banco de dados em um estado inválido.

Por exemplo, se uma transação envolve transferir dinheiro de uma conta para outra, a atomicidade garante que o dinheiro seja debitado de uma conta e creditado na outra simultaneamente, sem que haja a possibilidade de ocorrer apenas uma das operações.

Um exemplo de sistema que implementa a atomicidade é o PostgreSQL, que usa um mecanismo chamado Write-Ahead Logging (WAL) para registrar as operações das transações em um arquivo de log antes de aplicá-las no banco de dados. Assim, se ocorrer uma falha durante a execução de uma transação, o sistema pode usar o arquivo de log para desfazer ou refazer as operações e restaurar o estado consistente do banco de dados.

Figura 2: Exemplo Write-Ahead Logging



#### Fonte: Portal InterDB, 2023

# CONSISTÊNCIA

Consistência significa que uma transação deve preservar as regras e os invariantes definidos pelo banco de dados, ou seja, ela deve levar o banco de dados de um estado válido para outro estado válido. Isso evita que uma transação viole as restrições de integridade do banco de dados, como chaves primárias, chaves estrangeiras, domínios e triggers.

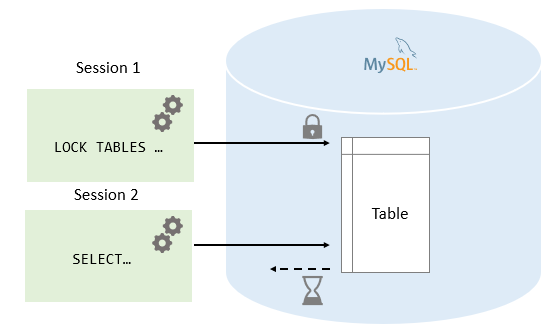
Por exemplo, se uma transação envolve inserir um novo registro em uma tabela, a consistência garante que o registro respeite as regras da tabela e não cause conflitos com outros registros existentes. Um exemplo de sistema que implementa a consistência é o Oracle Database, que usa um mecanismo chamado Constraint Validation para verificar se as operações das transações obedecem às restrições definidas pelo usuário ou pelo sistema. Se alguma restrição for violada, o sistema aborta a transação e retorna um erro ao usuário.

# ISOLAMENTO

Isolamento significa que uma transação deve ser executada de forma independente das outras transações concorrentes, ou seja, ela deve ter acesso exclusivo aos dados que manipula. Isso evita que uma transação interfira ou seja afetada por outras transações que estejam ocorrendo ao mesmo tempo.

Por exemplo, se duas transações envolvem ler e atualizar o mesmo dado, o isolamento garante que elas não leiam valores inconsistentes ou sobrescrevam os valores uns dos outros. Um exemplo de sistema que implementa o isolamento é o MySQL, que usa um mecanismo chamado Locking para controlar o acesso concorrente aos dados. O sistema pode usar diferentes tipos de bloqueios (locks), como compartilhados (shared), exclusivos (exclusive) ou intencionais (intention), para permitir ou impedir que as transações leiam ou atualizem os dados. O nível de isolamento determina quais tipos de bloqueios são usados e quais tipos de anomalias são evitadas ou permitidas.

Figura : Exemplo Locking



#### Fonte: Portal MySQL Tutorial, 2023

# DURABILIDADE

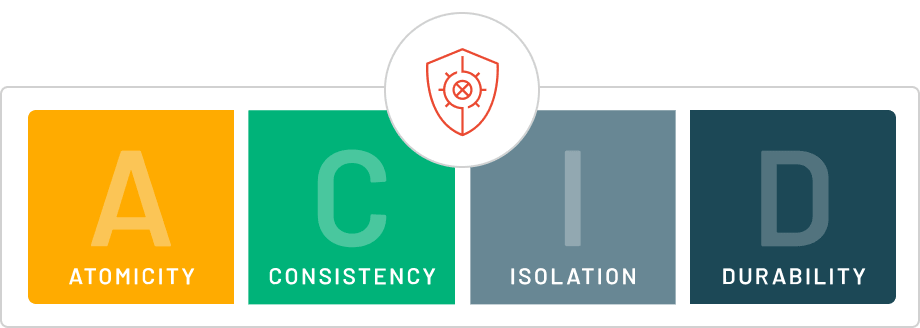
Durabilidade significa que uma transação deve ser persistente no banco de dados após ser concluída com sucesso, ou seja, ela deve resistir a falhas do sistema ou da mídia. Isso evita que uma transação perca os efeitos das suas operações em caso de interrupção do funcionamento do banco de dados.

Por exemplo, se uma transação envolve alterar o saldo de uma conta bancária, a durabilidade garante que o novo saldo seja gravado no disco e não seja perdido em caso de queda de energia ou pane no disco. Um exemplo de sistema que implementa a durabilidade é o MongoDB, que usa um mecanismo chamado Journaling para armazenar as operações das transações em um arquivo separado chamado journal. O journal é escrito no disco a cada 100 milissegundos, garantindo que as operações sejam preservadas em caso de falha. O sistema também usa um mecanismo chamado Checkpoints para sincronizar periodicamente os dados do journal com os dados do banco de dados, reduzindo o tempo de recuperação em caso de falha.

# CONCLUSÃO

Os conceitos de ACID são fundamentais para garantir a confiabilidade e a consistência dos bancos de dados. No entanto, eles também implicam em custos de desempenho e escalabilidade, pois exigem mecanismos de controle e sincronização das transações. Por isso, alguns sistemas de gerenciamento de banco de dados optam por relaxar algumas dessas propriedades em troca de maior eficiência e disponibilidade. Nesses casos, é preciso avaliar os requisitos e os trade-offs envolvidos na escolha do nível adequado de garantia ACID para cada aplicação.

Figura : Exemplo ACID



#### Fonte: Portal Databricks, 2023

# BIBLIOGRAFIA

REIS, Fábio Dos. **Conceitos de Bancos de Dados – O que significa ACID** http://www.bosontreinamentos.com.br/bancos-de-dados/conceitos-de-bancos-de-dados-o-que-significa-acid/. Acesso: 06/06/2023

Portal InterDB. **Write Ahead Logging — WAL**. Disponível em: https://www.interdb.jp/pg/pgsql09.html. Acesso em: 06/06/2023

Portal Databricks. **ACID Transactions**. Disponível em: https://www.databricks.com/glossary/acid-transactions. Acesso em: 06/06/2023

Portal 4Linux. **Banco de dados Relacional ACID e NoSQL BASE**. Disponível em: https://4linux.com.br/o-que-e-banco-de-dados-acid/. Acesso em: 06/06/2023