



**BANCO DE DADOS**

**CONCORRÊNCIA**

RICARDO SONAGLIO ALBANO



## Concorrência

- Tudo seria perfeito, se não existisse o fato de que várias transações podem ser executadas no mesmo momento e estarem utilizando o mesmo conjunto de dados;
- Devido à este fato, podem acontecer problemas durante a execução de uma transação;



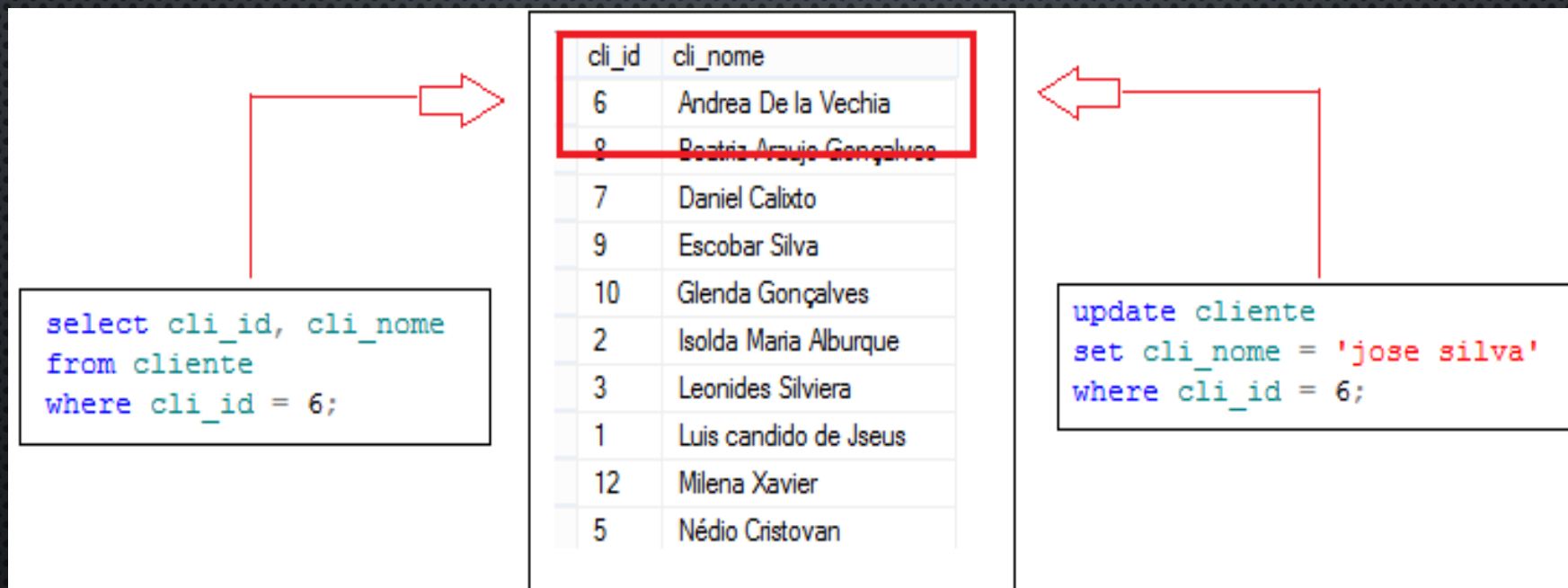
## Concorrência

- Problemas da concorrência:
  - Dirty Read (Leitura Suja);
  - Nonrepeatable Read (Leitura Não repetida);
  - Phantom Reads (Leitura Fantasma);
  - Lost Update (Atualização Perdida);



## Concorrência

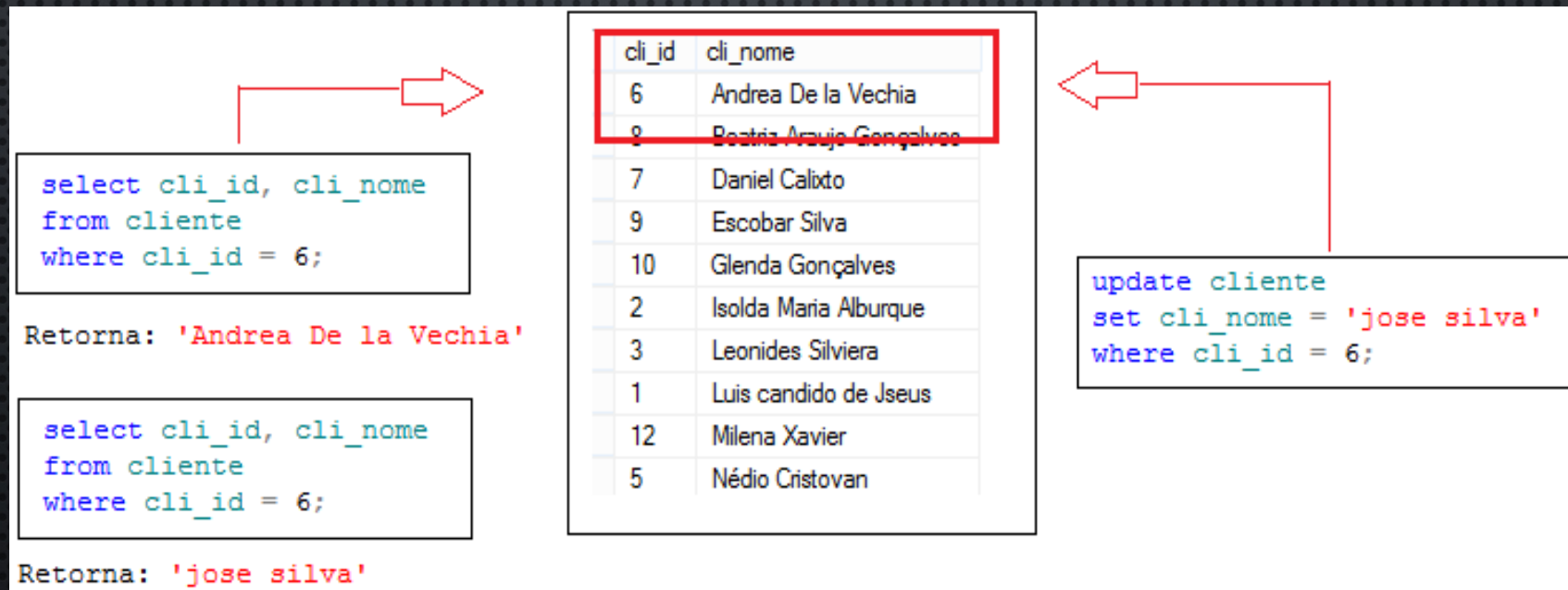
- Dirty Read (Leitura Suja):
  - Uma transação acessa dados que estão sendo atualizados por outra transação;





## Concorrência

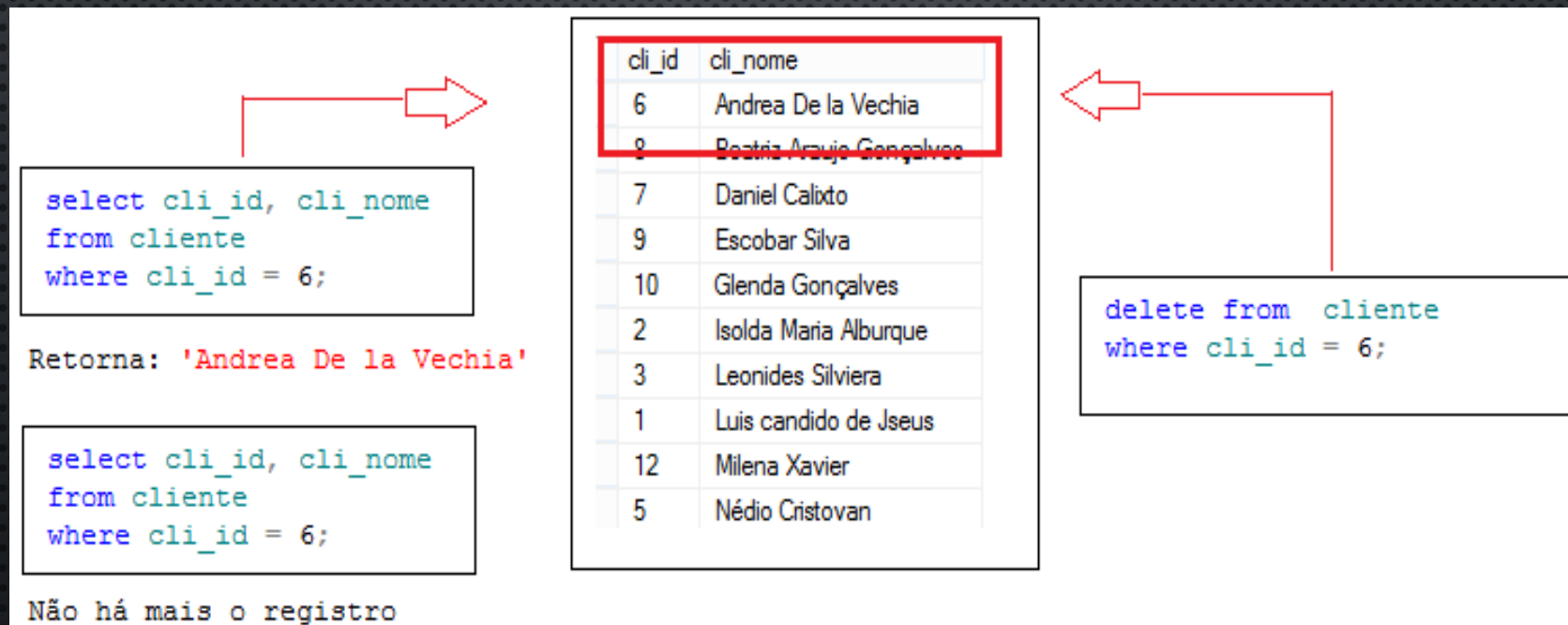
- Leitura não repetida (Nonrepeatable Read):
  - A transação acessa o mesmo registro várias vezes e obtém valores diferentes em cada vez;





## Concorrência

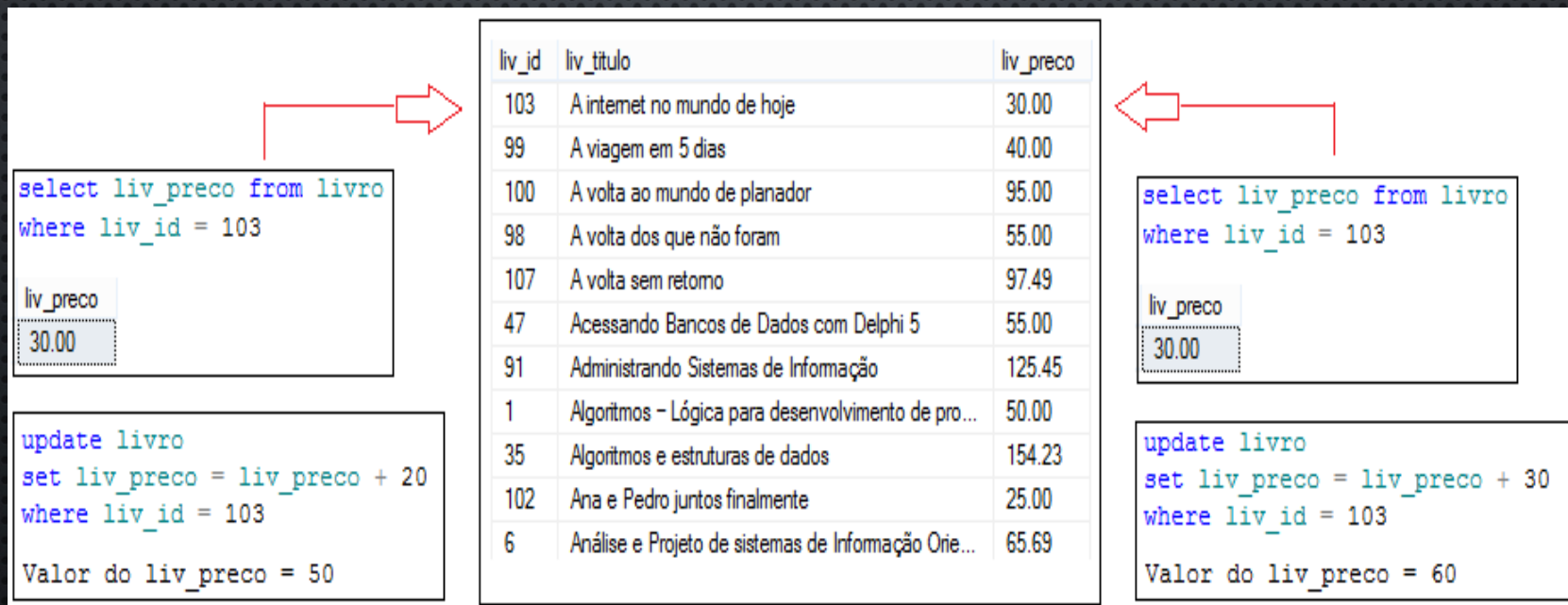
- Phantom Reads (Leitura Fantasma):
  - Revela-se através de diferenças no número de linhas obtidas entre cada leitura;





## Concorrência

- Lost Update (Atualização Perdida):
  - Duas ou mais transações consultam a mesma linha e a atualizam com base no valor original;





## Concorrência

- Níveis de isolamento da Concorrência:
  - Deve-se evitar o acesso simultâneo de dois ou mais usuários a um mesmo registro, essa situação pode quebrar a consistência dos dados.
  - O mecanismo no SQL Server é o bloqueio (LOCK).
  - Bloqueios são acionados automaticamente durante a transação.



## Concorrência

- Níveis de isolamento da Concorrência:
  - Os bloqueios diferem quanto à:
    - Durabilidade (se persistem até o término da transação);
    - Tipo (exclusivos ou compartilhados) e
    - Abrangência (por linha, página, tabela ou grupo de linhas);



## Concorrência

- Níveis de isolamento da Concorrência:
  - Níveis de bloqueios:
    - Read Uncommitted;
    - Read Committed;
    - Repeatable Read;
    - Serializable;



## Concorrência

- Read Uncommitted;
  - Permite a leitura de dados não atualizados;
  - Melhora a performance;
  - A adoção desse método deve ser criteriosamente avaliada, já que infringe uma regra básica: fornecer leituras consistentes;



## Concorrência

- Read Committed;
  - Somente dados já atualizados podem ser lidos;
  - Os registros manipulados por comandos INSERT, UPDATE e DELETE permanecem bloqueados para outras sessões até que a transação seja concluída;
  - É o padrão do SQL Server
  - Não impede Leituras não repetíveis ou dados fantasmas;



## Concorrência

- Repeatable Read:
  - Instruções não podem ler dados que foram modificados, mas que ainda não foram confirmados por outras transações e
  - Nenhuma outra transação pode modificar dados que foram lidos pela transação atual até que a transação atual seja concluída;



## Concorrência

- Serializable:
  - Nenhuma outra transação pode modificar dados lidos pela transação atual até que a transação atual seja concluída;
  - Outras transações não podem inserir linhas novas com valores chave que estejam no intervalo de chaves lido por qualquer instrução da transação atual até que esta seja concluída;



## Concorrência

- Níveis de Isolamento:

Nível de isolamento	Dirty read	Nonrepeatable read	Phantom read
Read uncommitted	Possível	Possível	Possível
Read committed	Impossível	Possível	Possível
Repeatable read	Impossível	Impossível	Possível
Serializable	Impossível	Impossível	Impossível



## Concorrência

- Níveis de Isolamento:
- Comandos:

```
set transaction isolation level { serializable | repeatable read | read committed |  
                                read uncommitted };
```

- Exemplos:

```
Set transaction isolation level read committed;
```

```
Set transaction isolation level repeatable read;
```



## Concorrência

- Níveis de Isolamento:
  - Podem Definidos para cada comando de uma transação, usando a cláusula WITH ao final do comando;
  - Valores aceitos:

<comando SQL> WITH( { serializable | repeatableread | readcommitted |  
read uncommitted } );



**Obrigado por sua atenção!**

Ricardo Sonaglio Albano