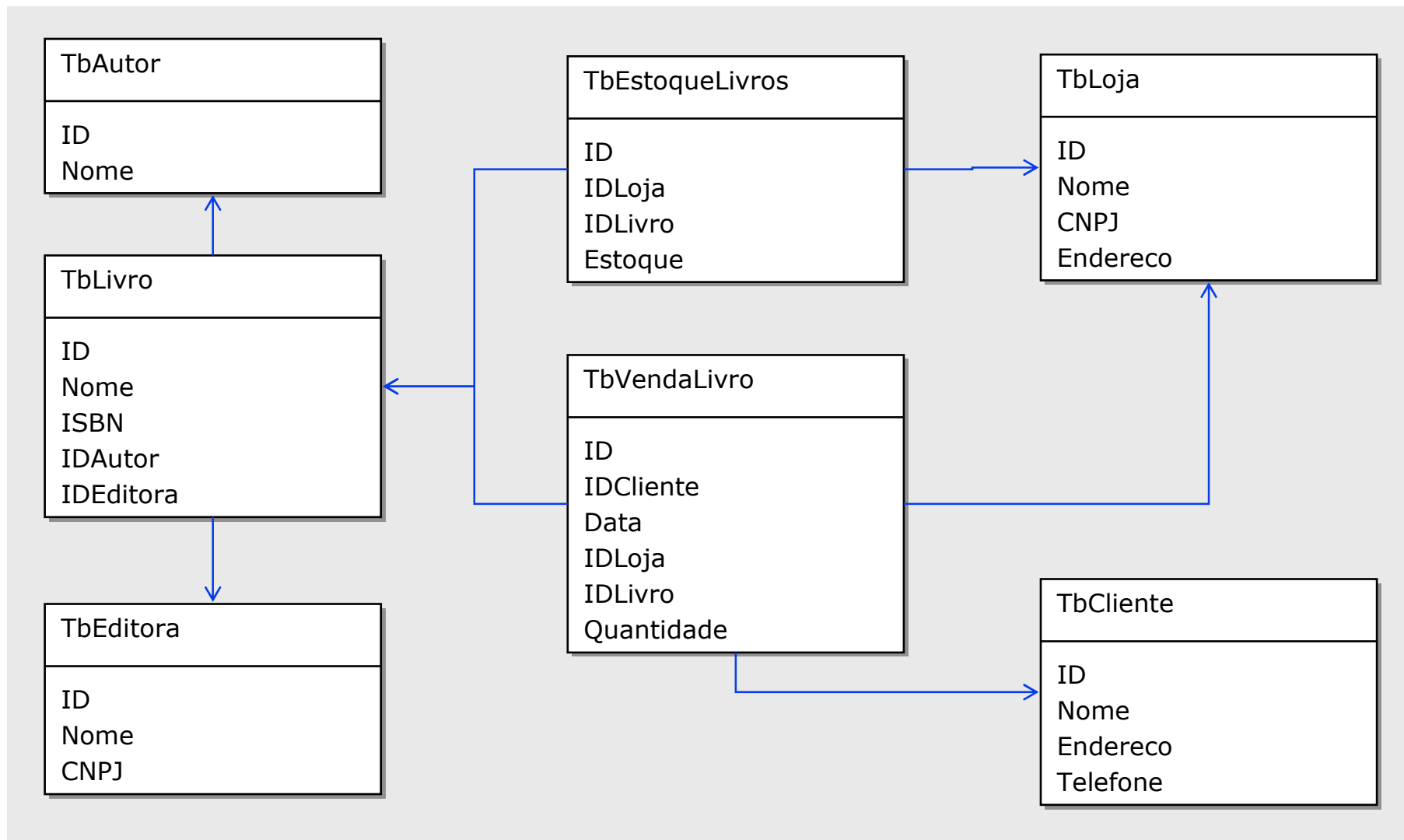

Processamento de Transações

Prof. Rodrigo Salvador Monteiro

salvador@ic.uff.br

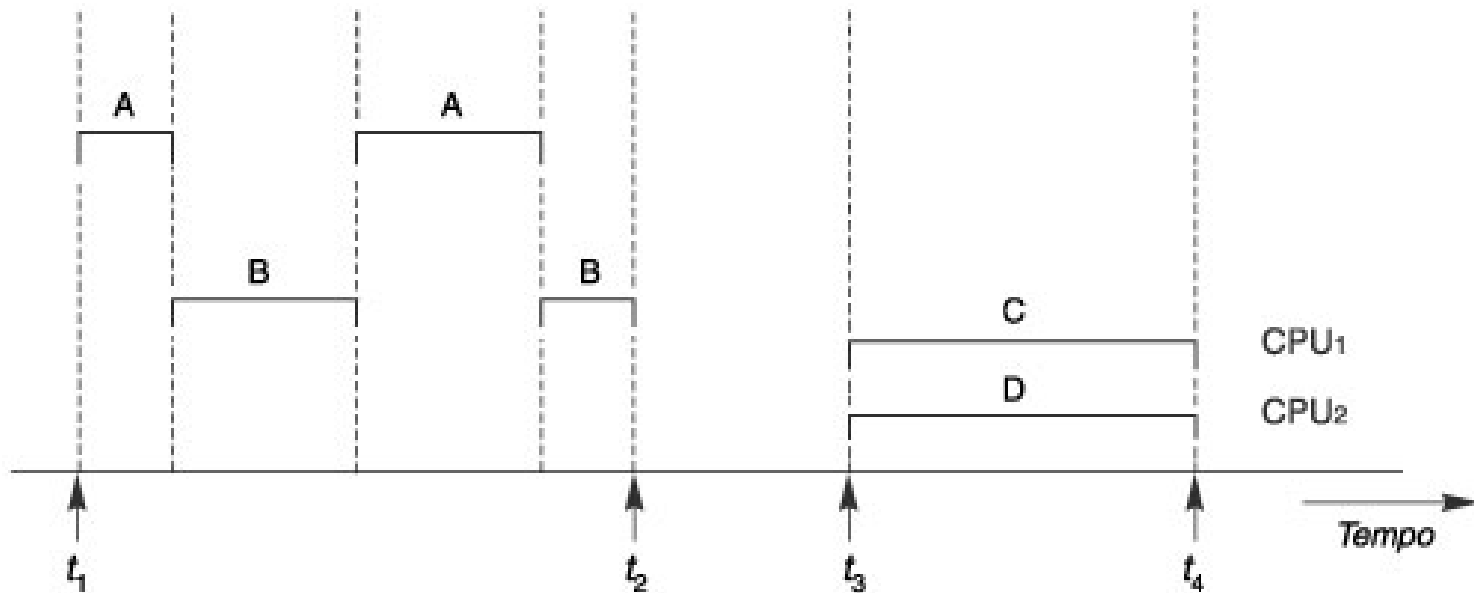
SQL – Tabelas Exemplo



Processamento de Transações

- ❑ Sistemas de processamento de transações
 - Sistemas que envolvem **grandes bancos de dados** e **centenas de usuários** executando transações **concorrentemente**
 - De acordo com o número de usuários acessando um SGBD, este pode ser classificado como **monousuário** ou **multiusuário**
 - SGBD multiusuários existem em função da **multiprogramação**, através da qual um sistema operacional pode executar diversos processos (aplicações) de forma **concorrente**
 - O SGBD deve estar preparado para receber comandos das diversas aplicações concorrentes, sem que um comando de uma aplicação afete os comandos de outras aplicações
 - » Execução paralela
 - » Execução entrelaçada

Processamento de transações concorrentes intercalado *versus* paralelo



Transações - Conceito

❑ Definição

- unidade lógica de trabalho, que trata diversos comandos isolados (CRUD) como uma única operação
- Embutidas em uma aplicação ou de forma interativa (SQL)
- Podem ser somente de leitura

Transações - Propriedades

A - C - I - D

- Atomicidade: uma transação é atômica, indivisível
 - » todos os seus comandos são executados com sucesso ou nenhum comando é executado
 - » Se algum dos comandos da transação gerar um erro, todos os comandos são desfeitos e o banco de dados é restaurado para sua condição inicial
- Consistência: Uma transação sempre leva o SGBD de um estado consistente para outro estado consistente
 - » o banco de dados nunca fica em um estado intermediário de execução (inconsistente)
 - » Os estados intermediários de execução não são visíveis para outras transações e não afetam os dados armazenados
- Isolamento: Cada transação é tratada como se estivesse sendo executada de forma isolada, não concorrente
 - » A ordem de execução das transações determinada pelo SGBD não deve interferir no resultado
- Durabilidade: as alterações de uma transação sobre o estado de um banco de dados devem ser permanentes, duráveis, persistentes

Transações - Granularidade

- Transações realizam operações de leitura e escrita nos “itens de dados nomeados”
 - » Granularidade de um item de dado é variável: um campo de um registro, um registro, um bloco de disco, ...
 - » Read_item(X), write_item(X)
- O conjunto de dados lidos por uma transação é chamado de read-set, enquanto o conjunto de dados alterados é o write-set
- Outros comandos
 - » Begin_transaction, end_transaction, commit_transaction, abort (ou rollback)

Transações - exemplo

(a) T_1

ler_item (X);
 $X := X - N;$
escrever_item (X);
ler_item (Y);
 $Y := Y + N;$
escrever_item (Y);

(b) T_2

escrever_item (X);
 $X := X + M;$
escrever_item (X);

Transações – Cenários exemplo

- ❑ Manutenção de uma tabela de estoque a partir de uma venda
 - Considere a venda de 3 unidades de um livro (IDLivro=267) em uma loja (IDLoja=12)
 - Após a inserção do registro da venda, o estoque do livro na loja deve ser atualizado
 - Se o registro da venda não for inserido...
o estoque não deve ser atualizado e vice-versa

```
BEGIN_TRANSACTION;
```

```
INSERT INTO TbVendaLivro (IDCliente, Data, IDLoja, IDLivro, Quantidade)  
VALUES (343, '2004-10-03', 12, 267, 3);
```

```
UPDATE TbEstoqueLivros  
SET Estoque = Estoque - 3  
WHERE IDLoja = 12 AND IDLivro = 267;
```

```
COMMIT_TRANSACTION;
```

Transações e Concorrência

- ❑ Considere um cenário hipotético ...
 - Você consulta sobre a existência de um livro e o vendedor indica que existe apenas uma cópia
 - Enquanto você pensa se vai comprar o livro, um cliente consulta outro vendedor sobre o mesmo livro
 - Você decide comprar o livro e seu nome é associado à última unidade disponível do livro
 - O outro cliente também decide comprar e o vendedor, de acordo com o resultado da consulta anterior, associa o nome do cliente com a mesma unidade do livro
 - A última atualização sobreescreve seu nome ...

- ❑ Quem fica com o livro ?
 - Não sei, mas o banco de dados não deveria ter permitido esta operação ...

 - Esta situação é conhecida como “Problema da Perda de Atualização”

Transações e Concorrência

- ❑ Outro problema... "Leitura suja"

```
Update college Set enrollment = enrollment + 1000  
where cName = 'Stanford'
```

Concorrente com ...

```
Select Avg(enrollment) From college
```

Transações - Finalidades

- ❑ Finalidades de transações em ambientes multiusuário
 - Controle de concorrência
 - » fazer com que os comandos emitidos por um usuário não afetem os demais usuários
 - Recuperação de falha
 - » permitir que o banco de dados seja trazido para um estado consistente após uma falha

Execuções Concorrentes

- ❑ A questão chave é garantir consistência mesmo com a execução concorrente de transações.
- ❑ O problema é que mesmo que cada transação individual esteja correta, na execução concorrente a consistência pode não ser preservada.
- ❑ Escalonamento
 - descreve uma sequência de execução, ou seja, a ordem cronológica em que as instruções são executadas.
- ❑ Escalonamentos serial
 - Sem concorrência
- ❑ Escalonamento não serial (entrelaçado)
 - Quando diversas transações são executadas de forma concorrente
- ❑ O problema é que nem todos escalonamentos não serial levam o banco de dados a um estado consistente.

Controle de Concorrência

- ❑ Problemas que podem ocorrer quando transações são executadas de forma concorrente sem controle
 1. Problema da atualização perdida
 2. Problema da atualização temporária (ou leitura suja)
 3. Problema do sumário incorreto
 4. Problema da leitura não repetida

Problema da atualização perdida

- duas transações acessam um mesmo item no SGBD têm suas operações entrelaçadas.
 - a atualização realizada por uma transação pode ser perdida em decorrência de uma escrita da segunda transação
- » R1 (X), R2 (X), U1 (X), W1 (X), U2 (X), W2 (X)

Problema da atualização temporária (ou leitura suja)

- uma transação atualiza e escreve um item do SGBD, falhando em seguida (abort).
 - uma segunda transação lê o item atualizado pela primeira antes que seu valor original seja restaurado
- » R1 (X), U1 (X), W1 (X), R2 (X), A1, U2 (X), W2 (X)

Problema do sumário incorreto

- uma transação calcula uma função agregadora, tal como uma soma ou média, enquanto outras transações atualizam itens que participam da agregação.
- Ao fim da operação de agregação, seu resultado não reflete os valores atuais do SGBD

(c)

T_1	T_3
	$sum:=0;$ $ler_item(A);$ $sum:=sum+A;$ \vdots
$ler_item(X);$ $X:=X-N;$ $escrever_item(X);$	
	$ler_item(X);$ $sum:=sum+X;$ $ler_item(Y);$ $sum:=sum+Y;$
$ler_item(Y);$ $Y:=Y+N;$ $escrever_item(Y);$	

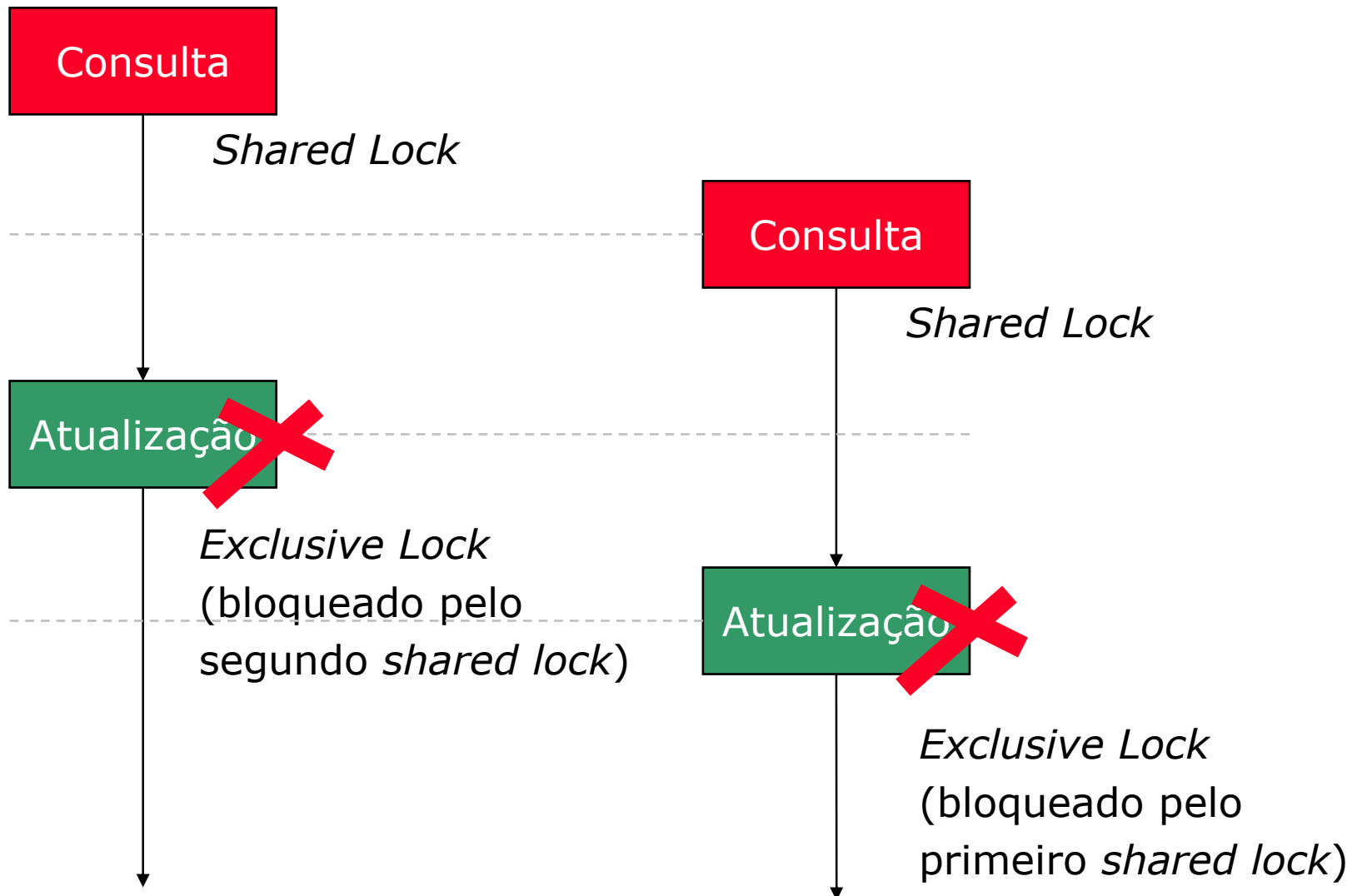
Problema da leitura não repetida

- uma transação lê duas vezes um item do SGBD e seu valor difere entre as duas leituras em decorrência de alterações promovidas por outras transações
- » R1 (X), R2 (X), U2 (X), W2 (X), R1 (X).

Implementação de Isolamento

- Diversos protocolos possíveis
 - » Bloqueios, timestamp, multiversão, etc...
- Bloqueio
 - » Bloqueio binário (locked/unlocked)
 - » Compartilhado/exclusivo
 - Exclusivo: somente um usuário pode acessar os dados dos registros afetados, utilizado em inserções, remoções e atualizações
 - compartilhado (shared): previne outras transações de aplicarem um locking exclusivo sobre um conjunto de dados, utilizados em consultas

Transações e Concorrência



Transações e Recuperação

- ❑ Conclusão da transação
 - Uma vez concluída uma transação, seus dados são registrados no SGBD de forma definitiva
 - Em sistemas modernos, os dados não serão perdidos mesmo que ocorra uma falha de disco ou memória em seguida
- ❑ Estratégia (informal) resumida para recuperação em um SGBD
 - Se houve falha catastrófica (ex: crash de disco) que causou grande perda no banco de dados então
 - » Recupere cópia anterior do banco (backup)
 - » A partir do backup do log, *refaça* as operações das transações confirmadas (committed) até o momento da falha
 - Se houve falha não catastrófica que causou inconsistência no banco de dados (sem perda física de dados) então
 - » A partir do log (on-line), *desfaça* operações das transações até o momento da falha

Log do banco de dados

- ❑ Para se recuperar de falhas, o sistema mantém um log
 - O log é utilizado para acompanhar as operações que afetam os itens do SGBD
 - O log também pode ser utilizado para fins de auditoria (controle de que usuário realizou determinada consulta ou atualização)
 - O log é mantido em disco para que uma falha de memória não afete o registro das operações
 - » Periodicamente copiado para backup (geralmente fita)
 - Cada entrada do log contém um ID da transação, o tipo de operação (escrita, leitura, abertura de transação, commit ou rollback) e seus parâmetros
 - » [start_transaction, T]
 - » [write_item, T, X, old_value, new_value]
 - » [read_item, T, X]
 - » [commit, T]
 - » [abort, T]

Transações – Recuperação

- ❑ As operações de uma transação podem não executar com sucesso devido a uma diversidade de falhas:
 - Falha do sistema computacional
 - » erros de hardware, sistema operacional, conexão de rede ou erros de acesso a mídia
 - Falha por operação inválida na transação
 - » erros internos nas operações (como divisão por zero ou overflow), devido a parâmetros errôneos ou erros de programação na transação
 - Exceções na transação
 - » ocorre devido a erros na lógica de negócio relacionada com a transação. Ex: saldo insuficiente em uma conta para realização de um saque
 - Interrupção da transação pelo controle de concorrência
 - » os mecanismos de controle de transações podem decidir abortar uma transação por detectar deadlock
 - Falhas no disco e problemas físicos
 - » falta de energia, roubo, sabotagem, incêndio, ...

Exercício

- ❑ Considere as seguintes sequências de operações sobre itens de dados em um SGBD, e faça o que se pede:

$R1(X) ; R3(X) ; W1(X) ; R2(X) ; W3(X)$

$R1(X) ; R2(X) ; R3(Y) ; W1(Y) ; W3(X)$

Quais problemas podem ocorrer em cada sequência de operações, caso não haja um controle de transações implementado?

- ❑ Considerando o SGBD PostgreSQL, o que é nível de isolamento de transações? Descreva e explique quais são os níveis de isolamento existentes neste SGBD.