



Fundamentos de Bancos de Dados

**Introdução aos
Sistemas de Bancos de Dados
Aula 1**





- Objetivo
 - Apresentar uma perspectiva histórica do surgimento dos SGBD, sua importância para as empresas e os principais conceitos e características envolvidas na área de Banco de Dados.
- Principais tópicos
 - Importância dos Bancos de Dados
 - Uma perspectiva histórica
 - Arquivos versus SGBD's
 - Quando usar e quando não usar SGBD
 - Principais Características dos SGBDs





- Principais tópicos (*continuação*)
 - Arquitetura “Three-schema”
 - Modelagem de Dados
 - Modelos de Dados (Conceituais, Lógicos e Físicos)
 - Matriz de Classificação de SGBDs
 - Síntese dos conceitos
 - Questões de Estudo



- A competitividade das empresas depende de dados precisos e atualizados.
- Conforme a empresa cresce, aumenta a sua dependência por dados abundantes e complexos.
- Assim, ferramentas de gerenciamento, extração rápida e precisa de informações é fundamental.
- Solução: Sistema Gerenciador de Banco de Dados, ou SGBD.





- No início da computação, programas tinham o único objetivo de armazenar e manipular dados.
- Esses programas gravavam seus dados em disco, segundo estruturas próprias.
- Somente programas que conheciam a estrutura dos dados podiam utilizar esses dados.





- Logo, se vários programas precisassem compartilhar os mesmos dados, eles teriam que conhecer e manipular as mesmas estruturas.





- Se algum programa precisasse realizar alguma mudança na estrutura de dados:
 - Todos os programas que acessavam esse mesmo arquivo tinham que ser alterados, mesmo que a alteração ocorresse em dados não eram manipulados por alguns dos programas.
- Isso gerava um grande problema:
 - Garantir a unicidade das estruturas de dados entre os diversos programas devido à existência de redundâncias.

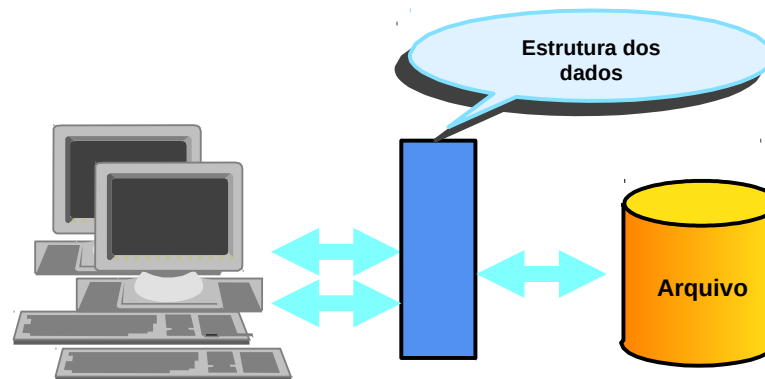


Redundância gera
inconsistência!





- Para evitar esse problema, colocou-se um sistema intermediário:
 - Que conhece a estrutura de dados do arquivo.
 - Fornece apenas dados que cada programa precisa.
 - Armazena adequadamente os dados de cada programa.



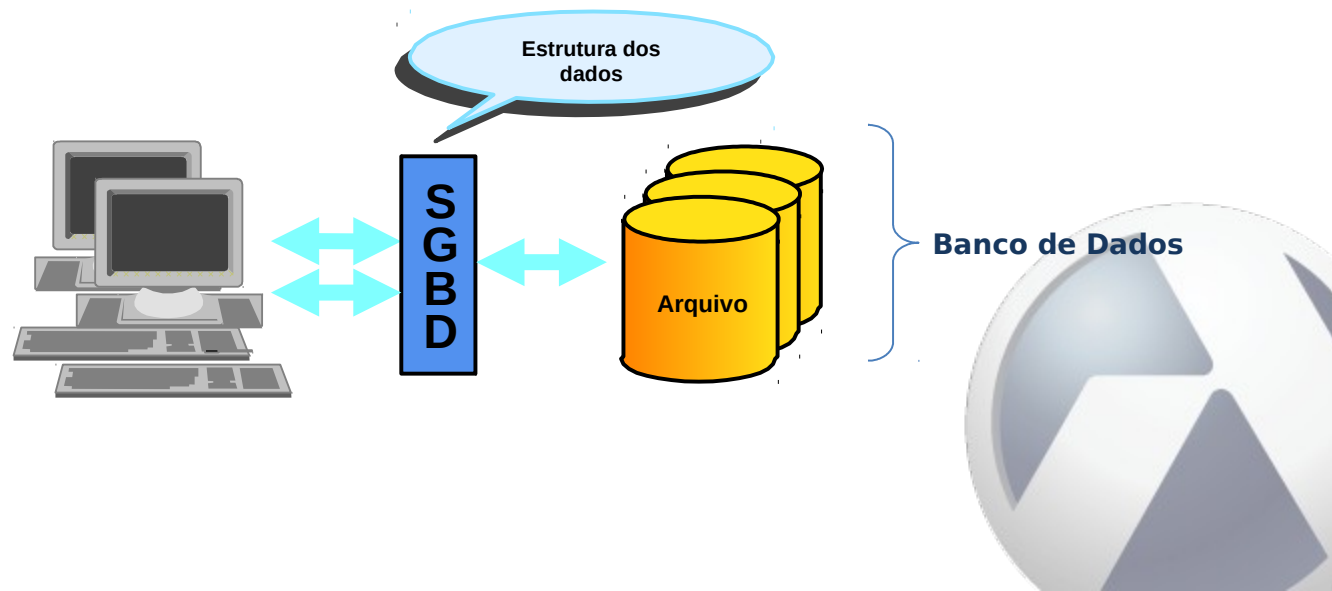


- Agora, com esse sistema intermediário:
 - Os programas "verão" apenas os dados que lhes interessam.
 - Os programas não precisam conhecer os detalhes de como seus dados estão gravados fisicamente.
 - Os programas não precisarão ser modificados se a estrutura de dados que utilizam não for modificada.
 - As alterações ficam concentradas nesse sistema intermediário.



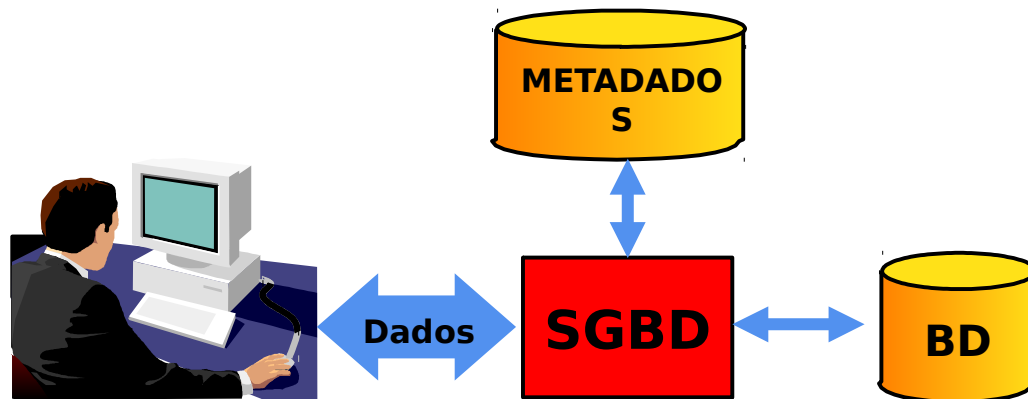


- Com o tempo, esse sistema intermediário passou a gerenciar vários arquivos.
- À essa coleção de arquivos foi dado o nome de Banco de Dados e o sistema intermediário recebeu o nome de Sistema Gerenciador de Banco de Dados (SGBD).





- O primeiro SGBD comercial surgiu em 1960.
- Com o tempo, surgiram padrões para descrever as Estruturas de Dados: os modelos de dados.
- A Estrutura de Dados do BD, segundo um modelo de dados é chamada de metadados.





- Hoje, um banco de dados:
 - É uma coleção de dados coerente e logicamente relacionados, com algum significado associado para atender a um propósito e audiência específicos.
 - Representa algum aspecto do mundo real, chamado de **minimundo**.





Processamento tradicional de Arquivos	SGBD	Vantagens do SGBD
Definição dos dados é parte do código de programas de aplicação	Meta Dados	eliminação de redundâncias
Dependência entre aplicação e dados	Independência entre aplicações e dados	eliminação de redundâncias
		facilidade de manutenção
Representação de dados em nível físico	Representação conceitual através de dados e programas	facilidade de manutenção
Cada visão é implementada por módulos específicos	Permite múltiplas visões	facilidade de consultas



Quando usar SGBD

- ❑ Controle redundância
- ❑ Controle consistência e integridade
- ❑ Acesso multiusuário
- ❑ Compartilhamento de dados
- ❑ Controle acesso e segurança
- ❑ Controle de recuperação e restauração
- ❑ Consultas eficientes

Quando não Usar SGBD

- ❑ Dados e aplicações simples e estáveis
- ❑ Requisitos de tempo-real não puderem ser atendidos



Arquitetura
**“Three-
schema”**

- Apoio a múltiplas visões de dados (nível externo)
- Capacidade de abstração de dados (nível conceitual)
- Capacidade de descrever a estrutura de armazenamento físico dos dados (nível interno)
- Compartilhamento de dados e processamento de transações.





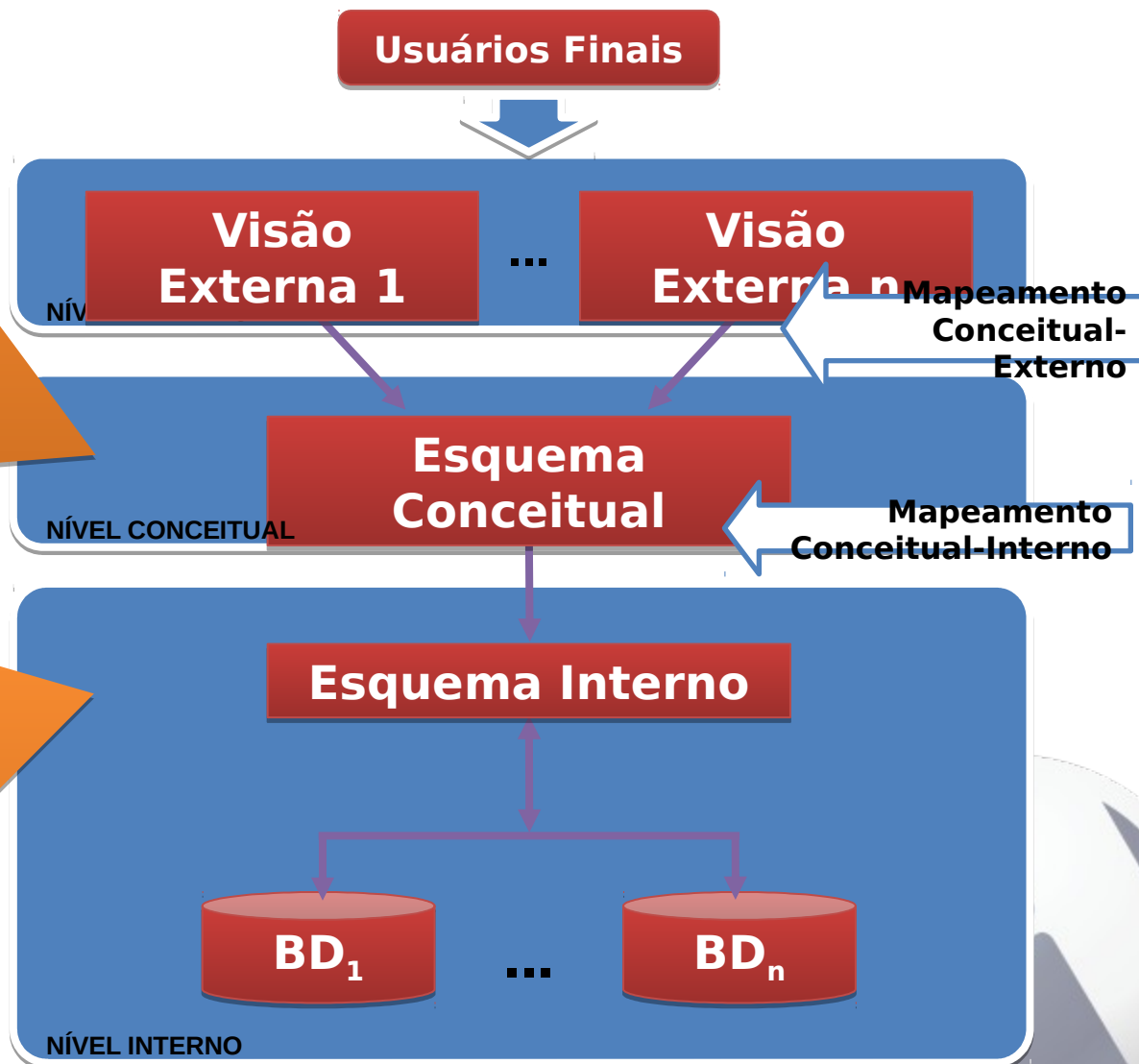
Arquitetura “Three-schema”

Independência Lógica de Dados:

É a capacidade de alterar o esquema conceitual sem ter que mudar os esquemas externos ou programas de aplicação.

Independência Física de Dados:

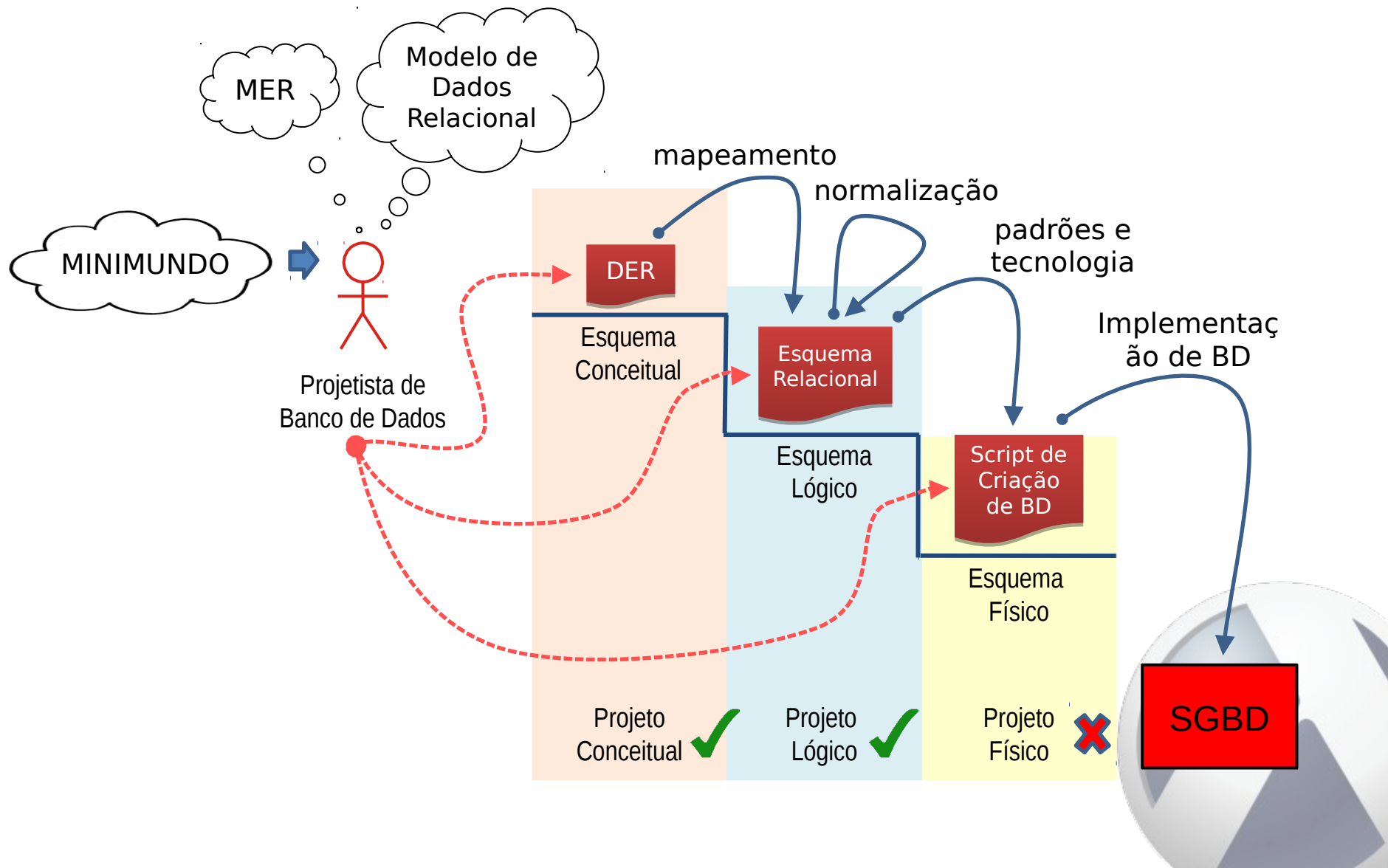
É a capacidade de alterar o esquema interno sem ter que alterar o esquema conceitual e externo.





- Existem modelos para diferentes níveis de abstração de representação de dados
 - modelos conceituais
 - modelos lógicos
 - modelos físicos
 - Referem-se:
 - organização dos arquivos de dados em disco
 - não são manipulados por usuários ou aplicações que acessam o BD
 - decisões de implementação são de cada SGBD





- Redes
- Hierárquico
- Relacional
- Entidade-Relacionamento
- ER - Estendido
- Objeto
- Objeto - Relacional





- Representação com alto nível de abstração
 - modelam de forma mais natural os fatos do mundo real, suas propriedades e seus relacionamentos
 - são independentes de BD
 - preocupam-se apenas com a semântica da aplicação
 - exemplo:
 - modelo entidade-relacionamento





- Representa os dados em alguma estrutura (lógica) de armazenamento de dados
 - também chamados de modelos de BD
 - dependente de BD
 - exemplos
 - modelo relacional (tabelas)
 - modelos hierárquico



- Apoiam:
 - na especificação dos dados do modelo (DDL)
 - dados, seus domínios e restrições
 - na especificação de como manipular os dados (DML)





- Possuem foco na:
 - Indexação e estrutura de arquivos
 - Transações e controle de concorrência
 - Otimização
 - Recuperação em casos de falhas
 - Mecanismos de proteção (segurança)
 - Partição e agrupamento de dados





- Uma transação define uma unidade de execução que pode acessar e atualizar vários itens de dados.
- Uma transação executa vários comandos como se fossem apenas um comando indivisível (atômico).
- Os vários comandos são delimitados pelas declarações *begin transaction* e (*commit* ou *rollback*):

– `begin transaction(x)`

- `Update(a)`
- `Delete(b)`
- `Insert(c)`

– `commit(x)`

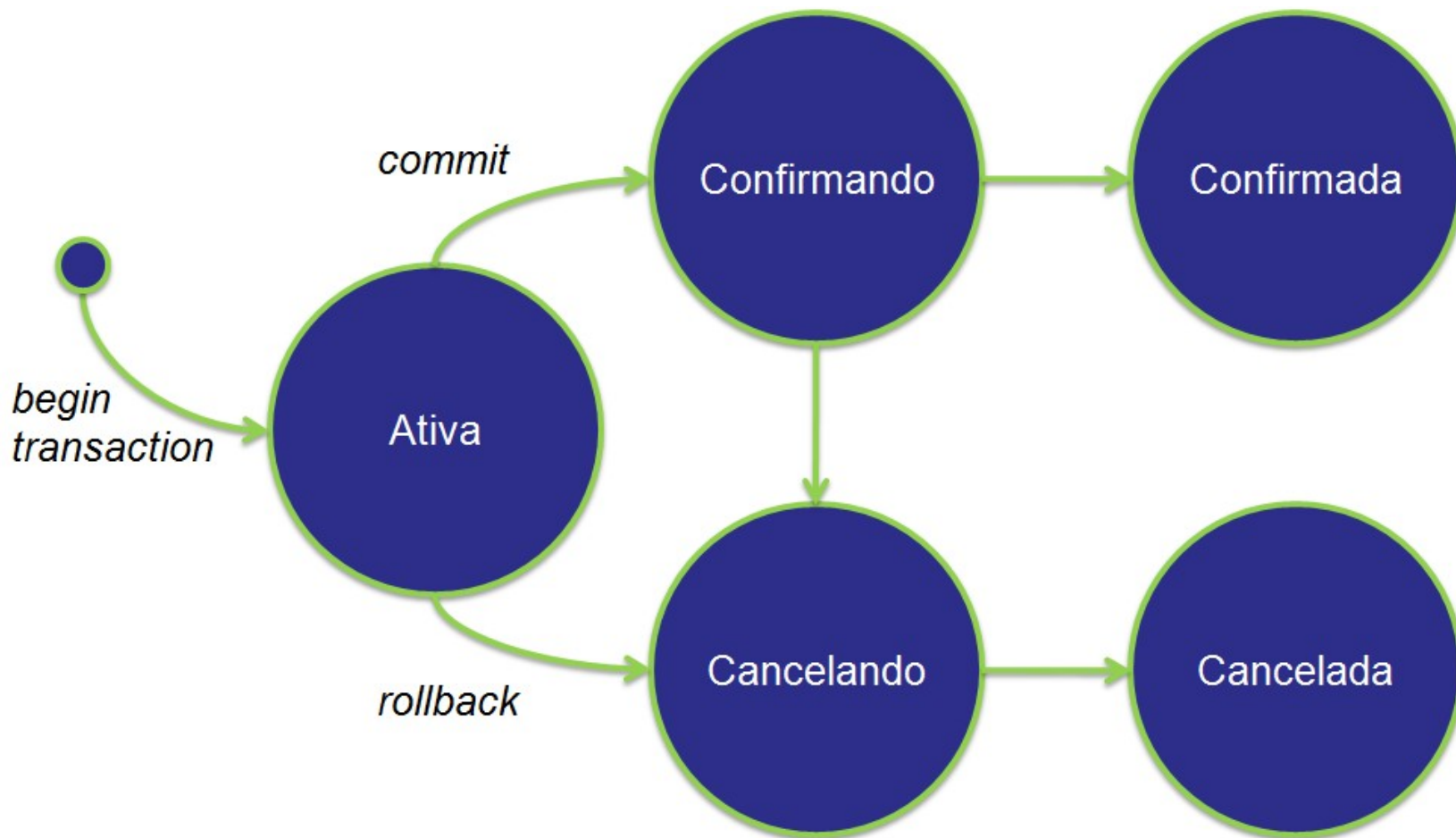
O SGBD considera este bloco como um único comando, atômico e indivisível





- Transações terminadas com *commit*, em caso de sucesso, efetivam todas as modificações realizadas dentro dela.
- Transação terminadas com *rollback*, desfazem todas as modificações realizadas dentro dela.
 - O banco de dados ficará no mesmo estado que estava antes do início da transação.
- O comando *rollback* pode ser chamado explicitamente pelo programador ou pelo SGBD quando ocorre algum erro.







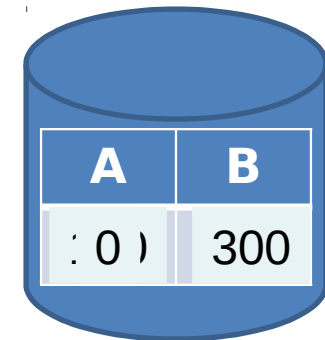
- Um SGBD deve controlar a execução concorrente de transações para assegurar que o estado do banco de dados permaneça consistente.
- A serialização é uma propriedade que garante que independente da ordem dos acessos aos dados feitos pelas transações, o resultado final será o mesmo.





- Execução das transações T1 e T2 em seqüência:

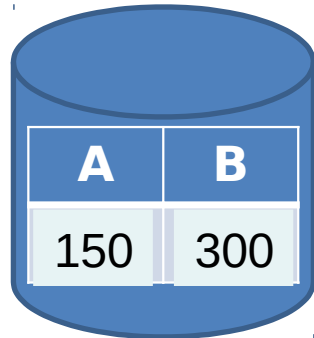
	A	B		A	B
	150	150		0	300
	T1			T2	
1	read (A);		7	read (A);	
2	A:=A-50;		8	A:=A-150;	
3	write(A);		9	write(A);	
4	read(B);		10	read(B);	
5	B:=B+50;		11	B:=B+150	
6	write (B);		12	write (B);	





- Problema que ocorre sem a serialização no controle de transação:

- troca de contexto antes do write (A)
- queda do banco antes de write (B)

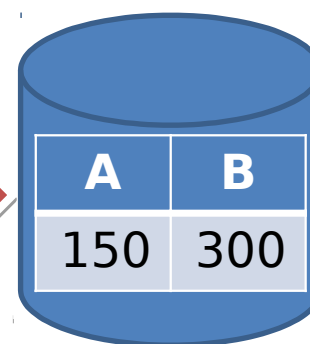
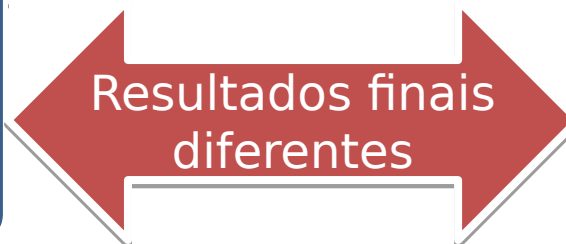
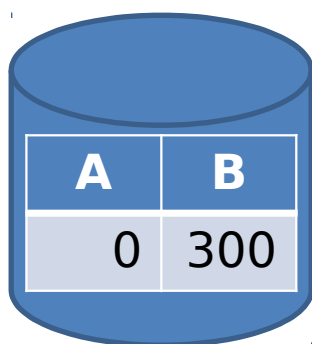


A		B	
150		150	
A		B	
50		300	
T1		T2	
1	read (A);	3	read (A);
2	A:=A-50;	4	A:=A-150;
6	write(A);	5	write(A);
7	read(B);	10	read(B);
8	B:=B+50;	11	B:=B+150;
9	write (B);	12	write (B);



	T1		T2
1	read (A);	7	read (A);
2	A:=A-50;	8	A:=A-150;
3	write(A);	9	write(A);
4	read(B);	10	read(B);
5	B:=B+50;	11	B:=B+150;
6	write (B);	12	write (B);

	T1		T2
1	read (A);	3	read (A);
2	A:=A-50;	4	A:=A-150;
6	write(A);	5	write(A);
7	read(B);	10	read(B);
8	B:=B+50;	11	B:=B+150;
9	write (B);	12	write (B);



- Transação devem possuir um conjunto de propriedades que é normalmente referido como propriedades ACID:
 - Atomicidade
 - Consistência
 - Isolamento
 - Durabilidade





- **Atomicidade**
 - Garante que todas as operações na transação serão executadas ou nenhuma será.
 - Isto evita que falha ocorridas, possam deixar o banco de dados inconsistentes.
- **Consistência**
 - Possui dois aspectos: A consistência do banco dados e a consistência da própria transação.
 - Uma transação não deve violar as restrições de integridade definidas para o banco de dados.





- Isolamento

- Significa que, mesmo no caso de transações executadas concorrentemente, o resultado final é igual ao obtido com a execução isolada de cada uma delas.
- A observância desta propriedade das transações pelos SGBDs impede a ocorrência dos problemas de acesso a dados.

- Durabilidade

- Significa que os resultados de uma transação, caso ela seja concluída com sucesso, devem ser persistentes. Mesmo se depois houver falha no sistema.





Transações externas

SGBD

Gerenciador de
Transações

Escalonador

Gerenciador de
Recuperação

Gerenciador
de Memória

Cache

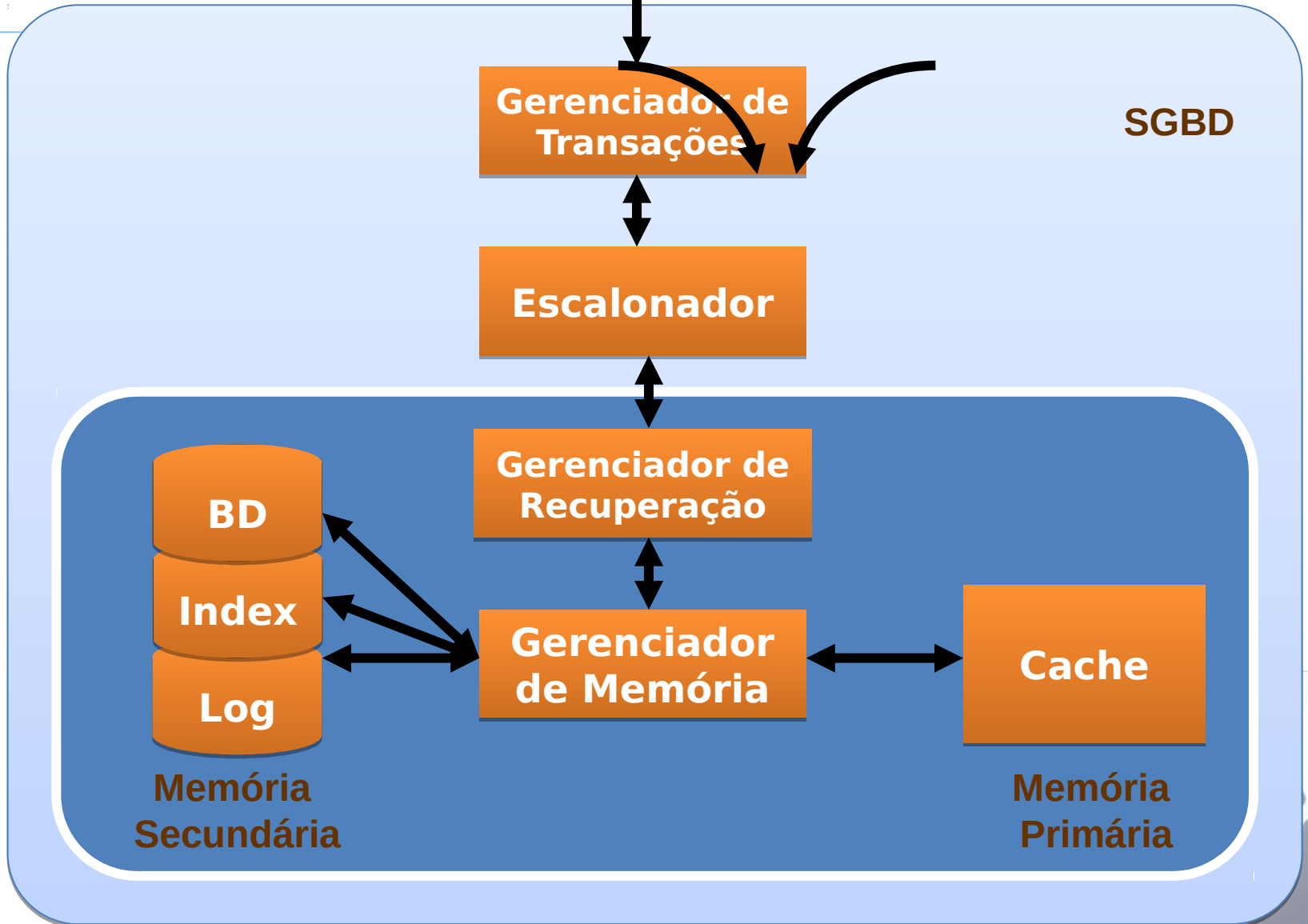
BD

Index

Log

Memória
Secundária

Memória
Primária





Consultas Complexas	RELACIONAL	OBJETO- RELACIONAL
	SISTEMA DE ARQUIVOS	LINGUAGEM DE PERSISTÊNCIA
Consultas Simples	Dados Simples	Dados Complexos

* Baseado no livro: Object Relational DBMS by Stonebraker and Moore, Morgan Kaufmann, 1996



- Banco de dados (BD):
 - conjunto de dados integrados que por objetivo atender a uma comunidade de usuários.
- Modelo de dados:
 - descrição formal das estruturas de dados para representação de um BD; com suas respectivas restrições e linguagem para criação e manipulação de dados.
- Sistema Gerenciador de banco de dados (SGBD):
 - software que incorpora as funções de definição, recuperação e alteração de dados em um BD.





- Modelagem de dados:
 - é a ação de representar/abstrair dados do minimundo com o objetivo de criar projetos conceituais e lógicos de um BD.
 - alguns autores incluem os projetos físicos como parte da modelagem de dados, pelo fato de que as otimizações são oriundas de análises do comportamento dinâmico do BD.





- **Projeto conceitual BD:**
 - ação que produz o esquema de dados abstratos que descreve a estrutura de um BD de forma independente de um SGBD (esquema conceitual).
- **Projeto lógico BD:**
 - ação que produz o esquema lógico de dados que representa a estrutura de dados de um BD em acordo com o modelo de dados subjacente a um SGBD.





- Projeto físico BD:
 - ação que produz o esquema físico de dados a partir do esquema de lógico de dados com a adição das estratégias de otimização para manipulação das estruturas de dados. As estratégias de otimização são dependentes dos fabricantes dos SGBDs e de suas versões.





Questões de Estudo





1. Quando faz sentido utilizar um SGBD ao invés de simplesmente utilizar o sistema de arquivos?
Quando não faz sentido utilizar um SGBD?
2. O que é independência lógica de dados e por que esse conceito é importante?
3. Explique as diferenças entre independência lógica de dados e independência física de dados.
4. Explique as diferenças entre esquemas externos, lógico e físico. Como esses conceitos se relacionam com os conceitos de independência de dados?
5. Quais são as responsabilidades de um Projetista de Banco de Dados e do DBA?





6. O Sr. Avarento quer guardar informações de seus funcionários (nome, endereço, momentos preocupantes). O volume de dados o forçou a decidir comprar um SGBD. Para economizar, ele quer comprar um que tenha apenas as características necessárias para executar uma aplicação *stand-alone* em seu PC. O Sr. Avarento não quer compartilhar essa lista com ninguém. Indique quais das seguintes características de SGBDs o Sr. Avarento necessita? Justifique.

- Segurança.
- Controle de concorrência.
- Recuperação após falhas.
- Mecanismos de visão.
- Linguagem de consulta.





7. Descreva os passos de um projeto de BD.
8. Quais dos seguintes itens exercem papel importante na representação de informações do mundo real num BD? Comente.
 - Linguagem de definição de dados.
 - Linguagem de manipulação de dados.
 - Cachê.
 - Modelo de dados.
9. O que é transação?
10. Por que o SGBD entrelaça as ações de diferentes transações, ao invés de executá-las seqüencialmente?





Referências





■ Referências Bibliográficas

1. Elmasri, R.; Navathe, S. B. [Trad.]. **Sistemas de bancos de dados.** Traduzido do original: FUNDAMENTALS OF DATABASE SYSTEMS. São Paulo: Pearson(Addison Wesley), 2005. 724 p. ISBN: 85-88639-17-3.
2. Korth, H.; Silberschatz, A. **Sistemas de Bancos de Dados.** 3a. Edição, Makron Books, 1998.
3. Raghu Ramakrishnan e Johannes Gehrke, **Database Management Systems**, Second Edition, McGraw-Hill, 2000.
4. Teorey, T.; Lightstone, S.; Nadeau, T. **Projeto e modelagem de bancos de dados.** Editora Campus, 2007.





Faculdade
IMPACTA
TECNOLOGIA

Obrigado!

Prof. Dr. Alexandre L. Rangel

alexandre.leite@faculdadeimpacta.edu.br

www.alexandrelrangel.blogspot.com.br

