

Apresentação



❖ Temas

- Introdução: banco de dados;
- Modelo relacional;
- Modelo de entidades-relacionamentos;
- SQL (Structured Query Language);
- Modelo relacional + MER + SQL;
- Práticas com MER + SQL;
- Criação de tabelas, consultas simples, consultas com maior complexidade, views, stored procedures, triggers.

Apresentação



❖ Temas

- Software: XAMPP (MySQL);

❖ Bibliografia:

- KORTH, Henry F.; SILBERSCHATZ, Abraham; S. Sudarshan. **Sistema de Banco de Dado**. Rio de Janeiro: Campus, 2006.
- DATE, C. **Introdução a sistemas de bancos de dados**. Rio de Janeiro: Campus, 2000.

Definição



❖ Banco de dados

- Coleção de dados relacionados.
- Fatos conhecidos que podem ser registrados e possuem significado implícito.
- Representa algum aspecto do mundo real.
- Coleção logicamente coerente de dados com algum significado inerente.
- Construído para uma finalidade específica.

Algumas aplicações



- ❖ Aplicações de banco de dados tradicionais
 - Armazena informações textuais ou numéricas.
- ❖ Bancos de dados de multimídia
 - Armazena imagens, clipes de áudio e streams de vídeo digitalmente.
- ❖ Sistemas de informações geográficas (GIS)
 - Armazena e analisa mapas, dados sobre o clima e imagens de satélite.

Algumas aplicações (cont.)



- ❖ Sistemas de data warehousing e de processamento analítico on-line (OLAP)
 - Extrair e analisar informações comerciais úteis de bancos de dados muito grandes.
 - Ajuda na tomada de decisão.
- ❖ Tecnologia de tempo real e banco de dados ativo
 - Controla processos industriais e de manufatura.

Um pouco de história



- ❖ O primeiro banco de dados teve início nas décadas de 60 e 70 na IBM, com pesquisas em automação de tarefas de escritório.
 - Naquela época, tinha-se que ter muita gente (\$\$\$) para armazenar e indexar os arquivos.
- ❖ Em 1970 um pesquisador da IBM - Ted Codd publicou o primeiro artigo sobre bancos de dados relacionais. Este artigo tratava sobre uma forma de consultar os dados (em tabelas) por meio de comandos.

Um pouco de história



- ❖ Inicialmente, por ser muito complexo, não foi aceito nem implantado.
- ❖ A IBM criou um grupo de pesquisa chamado System R, com o objetivo de desenvolver um sistema de BD para ser comercializado.
- ❖ System R introduziu uma linguagem chamada Structured Query Language (SQL).

Um pouco de história



- ❖ Essa linguagem tornou-se padrão internacional para BD relacional.
- ❖ O System R passou a se chamar SQL/DS.
- ❖ SQL/DS evoluiu para DB2, que é o banco de dados comercial da IBM.

Um pouco de história



- ❖ O DB2 é um banco de dados relacional que é amplamente utilizado em aplicativos empresariais e é compatível com diversas plataformas, incluindo Linux, Unix e Windows;
- ❖ A IBM investe continuamente no desenvolvimento do DB2, adicionando novas funcionalidades e melhorias de desempenho para manter a sua relevância no mercado de bancos de dados empresariais.

Sistema de Arquivos vs. Banco de Dados

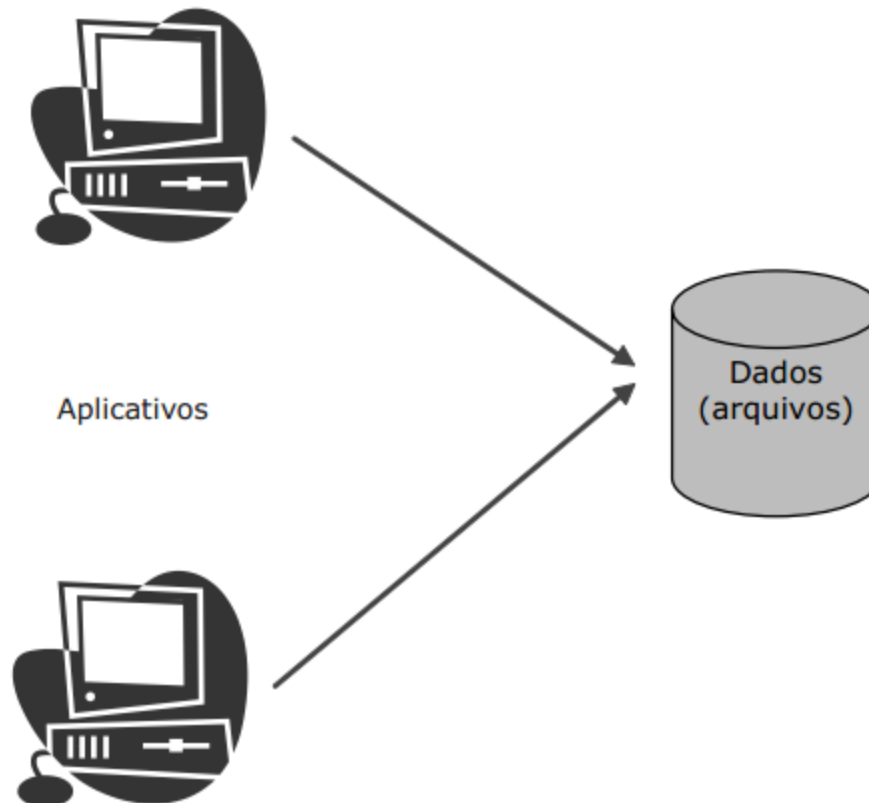


- ❖ Alguns Sistemas de Arquivos:
 - FAT32, NTFS, Ext2, Ext3, Ext4, ReiserFS....
- ❖ Alguns Bancos de Dados:
 - MySQL, SQL Server, Firebird, Postegre, DB2, Oracle...
- ❖ Ambos tem um objetivo em comum:
 - Armazenar e gerenciar dados em sistemas de computação.

Sistema de Arquivos vs. Banco de Dados



❖ Sistema de Arquivos



Sistema de Arquivos vs. Banco de Dados



- ❖ Sistemas de arquivos e bancos de dados são dois tipos de sistemas utilizados para armazenar e gerenciar dados. Embora compartilhem algumas semelhanças, há também muitas diferenças importantes entre eles;
- ❖ Ambos permitem que os dados sejam armazenados de forma organizada e estruturada.
- ❖ Ambos possibilitam o acesso rápido aos dados por meio de consultas ou buscas.
- ❖ Ambos oferecem mecanismos para proteger a integridade dos dados, como backups e recuperação de dados.

Sistema de Arquivos vs. Banco de Dados



- ❖ Há, também, algumas diferenças importantes;
- ❖ Finalidade: o objetivo principal de um sistema de arquivos é fornecer uma estrutura de armazenamento de arquivos simples e de fácil acesso. Já o objetivo de um banco de dados é proporcionar um sistema avançado para gerenciar e armazenar grandes quantidades de dados complexos, bem como fornecer mecanismos robustos para garantir a integridade dos dados e a segurança no acesso;
- ❖ Estruturação de dados: sistemas de arquivos utilizam uma estrutura de arquivos e pastas simples para armazenar dados. Já bancos de dados possuem uma estrutura de dados mais complexa, como tabelas, relações e índices;

Sistema de Arquivos vs. Banco de Dados



- ❖ Escalabilidade: um sistema de arquivos pode tornar-se rapidamente ineficiente e complexo à medida que a quantidade de dados aumenta, enquanto um banco de dados é projetado para lidar com grandes quantidades de dados e pode ser escalado horizontalmente adicionando mais servidores ao sistema;
- ❖ Compartilhamento de dados: um sistema de arquivos geralmente permite apenas o acesso a arquivos por um único usuário de cada vez, enquanto um banco de dados pode ser acessado por muitos usuários simultaneamente;
- ❖ Banco de dados geralmente fornecem mecanismos para gerenciar o acesso simultâneo a dados por muitos usuários. Eles usam transações para garantir que operações de dados sejam completadas ou revertidas, e normalmente oferecem controle de concorrência para gerenciar conflitos de acesso a dados. Isso torna o banco de dados uma escolha mais adequada para aplicações que requerem compartilhamento de dados por muitos usuários.

Sistema de Arquivos vs. Banco de Dados



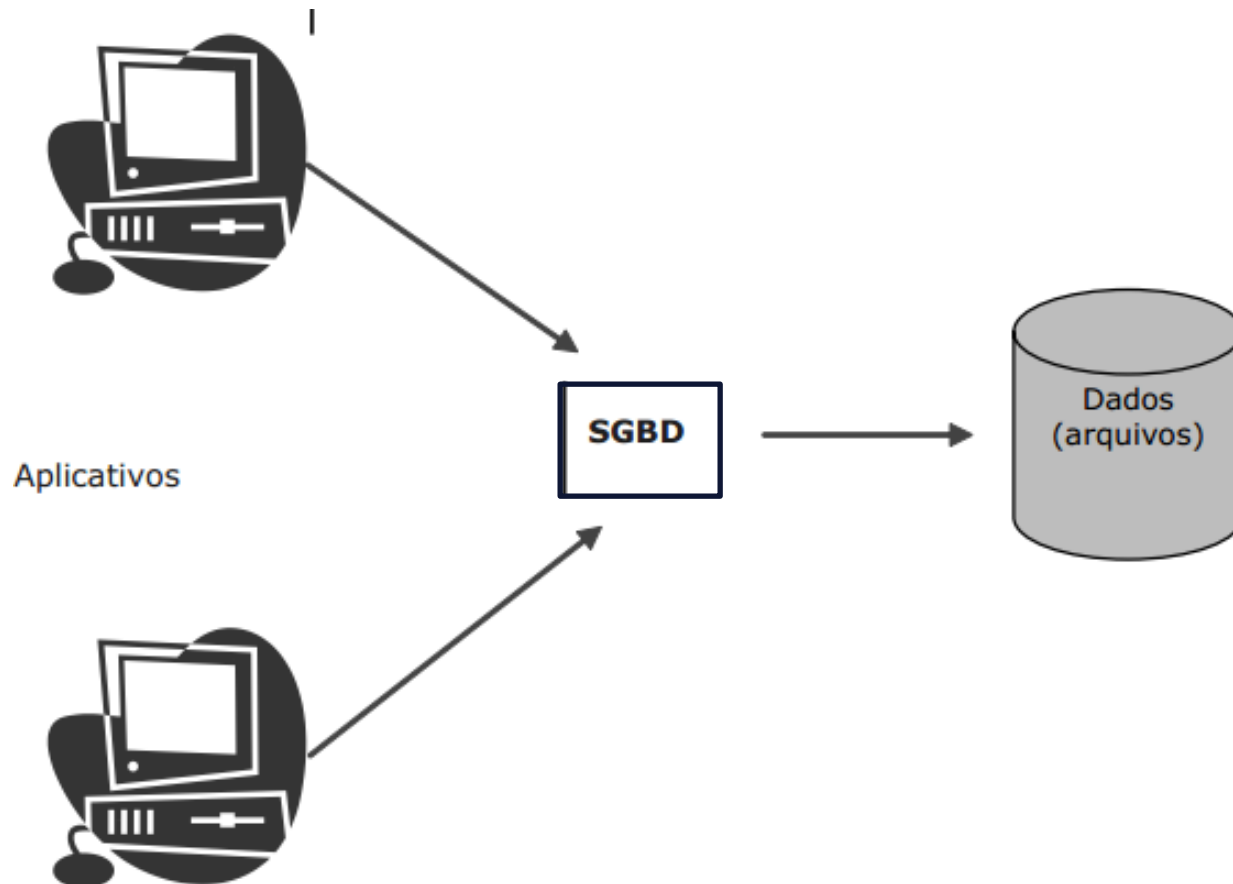
❖ Sistema de Arquivos

- A manutenção é prejudicada pois a estrutura de arquivos é definida e padronizada no próprio código do aplicativo (Cobol, Clipper, etc);
- O compartilhamento de um arquivo por vários programas apresenta dificuldades para gerenciar o acesso a esses arquivos e seu controle;
- O desenvolvimento de arquivos e programas de um mesmo SO é realizado isoladamente por programadores e linguagens diferentes, causando incompatibilidades no sistema;
- A falta de gerenciamento de acessos concorrentes aos dados e recuperação de dados.

Sistema de Arquivos vs. Banco de Dados



❖ Banco de Dados



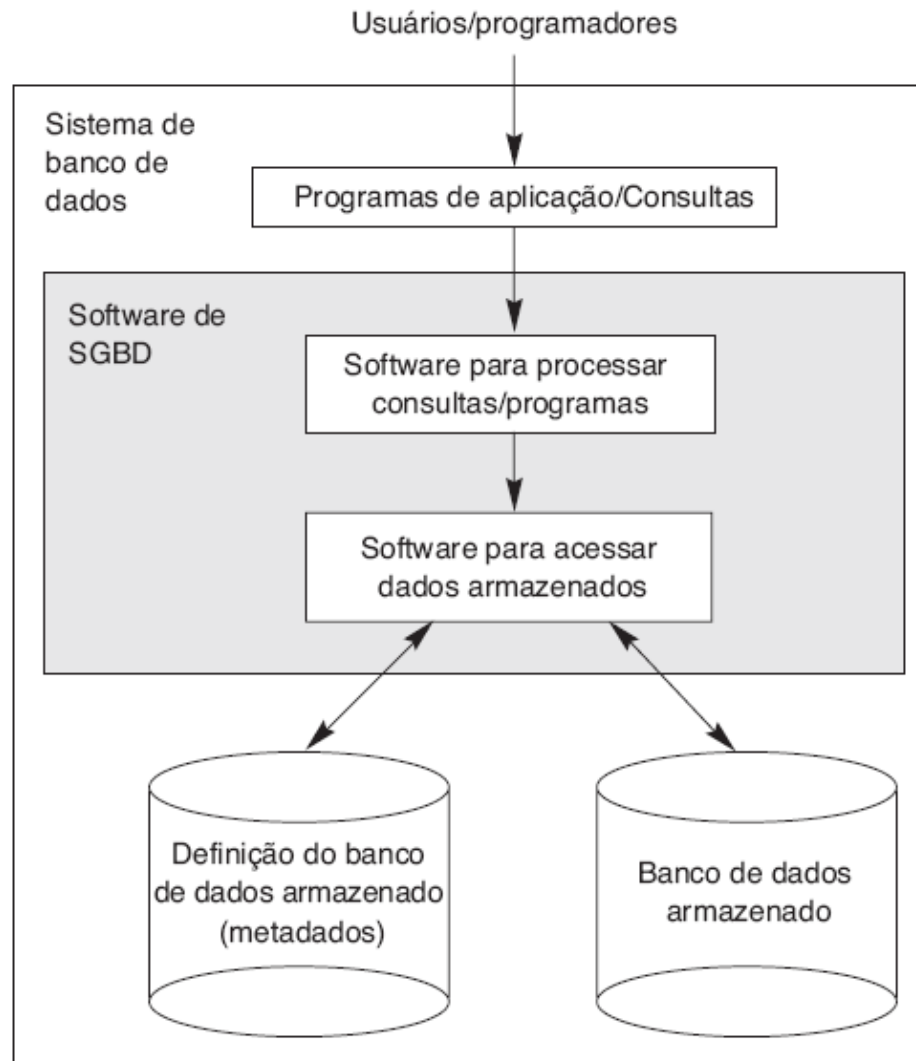


Figura 1.1

Diagrama simplificado de um ambiente de sistema de banco de dados.

Sistema de Arquivos vs. Banco de Dados



❖ Banco de Dados

- Rapidez no acesso às informações presentes no Banco de Dados;
- Redução de problemas de integridade e redundância;
- Diminuição do esforço humano no desenvolvimento;
- Utilização dos dados e controle integrado de informações distribuídas fisicamente.
- Descreve uma coleção lógica e coerente de dados com algum significado inerente. Uma organização randômica de dados não pode ser considerada um Banco de Dados;
- Constrói em atendimento a uma proposta específica.

Sistema de Arquivos vs. Banco de Dados



- ❖ Processamento de arquivo tradicional
 - Cada usuário define e implementa os arquivos necessários para uma aplicação de software específica.
- ❖ Abordagem de Banco de Dados
 - Um único repositório mantém dados que são definidos uma vez e depois acessados por vários usuários.

O que é um SGBD?



- ❖ **Sistema Gerenciador de Banco de Dados** é um conjunto de programas e ferramentas utilizadas para configurar, atualizar e manter um banco de dados.
 - Recursos para administrar usuários/permisões.
 - Recursos para criar/alterar tabelas e banco de dados.
 - Recursos para backup e restauração de dados.
 - Recursos para otimizar a performance do banco.

O que é um SGBD?



- ❖ É um software especializado que ajuda a gerenciar, armazenar e recuperar informações em um banco de dados. O objetivo principal de um SGBD é oferecer uma camada de abstração entre os dados armazenados e as aplicações que precisam acessá-los, tornando mais fácil e seguro para as aplicações trabalhar com dados;
- ❖ Os SGBDs fornecem uma ampla variedade de recursos e funcionalidades, incluindo:
- ❖ Estruturação de dados: permite a definição e manutenção da estrutura de dados, como tabelas, colunas, tipos de dados, chaves primárias e estrangeiras, índices, etc.
- ❖ Segurança: oferece mecanismos para controlar o acesso aos dados, incluindo autenticação, autorização e criptografia;

O que é um SGBD?



- ❖ Backup e recuperação: permite criar cópias de segurança dos dados e recuperá-los em caso de falhas.
- ❖ Otimização de desempenho: oferece mecanismos para otimizar o acesso aos dados, incluindo índices, otimização de consultas e cache de dados.
- ❖ Integridade de dados: garante a consistência e integridade dos dados armazenados, usando mecanismos como transações, restrições e regras de negócio.

Alguns SGBDs

The Oracle logo, featuring the word "ORACLE" in a bold, red, sans-serif font.

www.oracle.com

The IBM logo, consisting of the letters "IBM" in a blue, striped, sans-serif font.The DB2 logo, featuring the letters "DB2" in white, bold, sans-serif font on a green rectangular background.

www-01.ibm.com/software/data/db2/



www.firebirdsql.org/

The Sybase logo, featuring the word "SYBASE" in a large, black, serif font, followed by a vertical line, the text "An SAP Company", and the SAP logo.

www.sybase.com.br/



www.microsoft.com/sqlserver/en/us/default.aspx

The PostgreSQL logo, featuring a stylized elephant icon next to the text "PostgreSQL" in a white, sans-serif font on a blue background.

www.postgresql.org/

The MySQL logo, featuring the word "MySQL" in a blue and orange, sans-serif font, with a stylized fish icon above the "SQL" part.

www.mysql.com/

Principais características de banco de dados



1. Natureza de autodescrição de um sistema de banco de dados.
2. Isolamento entre programas e dados, e abstração de dados.
3. Suporte de múltiplas visões dos dados.
4. Compartilhamento de dados e processamento de transação multiusuário.

1. Natureza de autodescrição de um sistema de banco de dados



- ❖ O sistema de banco de dados contém definição completa de sua estrutura e restrições.
 - Metadados, que descreve a estrutura do banco de dados.
- ❖ O catálogo é usado pelo:
 - Software de SGBD
 - Usuários do banco de dados que precisam de informações sobre a estrutura do banco de dados.



ALUNO

Nome	Numero_aluno	Tipo_aluno	Curso
Silva	17	1	CC
Braga	8	2	CC

DISCIPLINA

Nome_disciplina	Numero_disciplina	Creditos	Departamento
Introd. à ciência da computação	CC1310	4	CC
Estruturas de dados	CC3320	4	CC
Matemática discreta	MAT2410	3	MAT
Banco de dados	CC3380	3	CC

TURMA

Identificacao_turma	Numero_disciplina	Semestre	Ano	Professor
85	MAT2410	Segundo	07	Kleber
92	CC1310	Segundo	07	Anderson
102	CC3320	Primeiro	08	Carlos
112	MAT2410	Segundo	08	Chang
119	CC1310	Segundo	08	Anderson
135	CC3380	Segundo	08	Santos

HISTORICO_ESCOLAR

Numero_aluno	Identificacao_turma	Nota
17	112	B
17	119	C
8	85	A
8	92	A
8	102	B
8	135	A

PRE_REQUISITO

Numero_disciplina	Numero_pre_requisito
CC3380	CC3320
CC3380	MAT2410
CC3320	CC1310

Figura 1.2

Exemplo de banco de dados que armazena informações de aluno e disciplina.

RELACOES

Nome_relacao	Numero_de_colunas
ALUNO	4
DISCIPLINA	4
TURMA	5
HISTORICO_ESCOLAR	3
PRE_REQUISITO	2

COLUNAS

Nome_coluna	Tipo_de_dado	Pertence_a_relacao
Nome	Caractere (30)	ALUNO
Numero_aluno	Caractere (4)	ALUNO
Tipo_aluno	Inteiro (1)	ALUNO
Curso	Tipo_curso	ALUNO
Nome_disciplina	Caractere (10)	DISCIPLINA
Numero_disciplina	XXXXNNNN	DISCIPLINA
....
....
....
Numero_pre_requisito	XXXXNNNN	PRE-REQUISITO

Figura 1.3

Exemplo de um catálogo para o banco de dados na Figura 1.2.

2. Isolamento entre programas e dados



- ❖ Independência de dados do programa
 - A estrutura dos arquivos de dados é armazenada no catálogo do SGBD separadamente dos programas de acesso (cliente).
- ❖ Independência da operação do programa
 - Uma operação é especificada em duas partes:
 - A interface de uma operação inclui o nome da operação e os tipos de dados de seus argumentos.
 - A implementação da operação pode ser alterada sem afetar a interface.

3. Abstração de dados



❖ Abstração de dados

- Permite a independência de dados do programa e a independência da operação do programa.
- Se adicionar um campo na tabela, na próxima consulta aquele dado já estará disponível.

❖ Representação conceitual de dados

- Através do Diagrama MER.
- Não inclui detalhes de como os dados são armazenados ou como as operações são implementadas.

4. Suporte para múltiplas visões dos dados



❖ Visão

- Subconjunto do banco de dados.
- Contém dados virtuais derivado dos arquivos do banco de dados, mas que não estão armazenados da forma que são visualizados.

5. Compartilhamento de dados e processamento de transação multiusuário



- ❖ Permite que múltiplos usuários acessem o banco de dados ao mesmo tempo.
- ❖ Software de controle de concorrência
 - Garante que vários usuários tentando atualizar o mesmo dado faça isso de uma maneira controlada, resultado dessas atualizações seja correto.



5. Compartilhamento de dados e processamento de transação multiusuário (cont.)

❖ Transação

- Programa em execução ou processo que inclui um ou mais acessos ao banco de dados.
- Propriedade de isolamento:
 - Cada transação parece executar isoladamente das outras transações.
- Propriedade de atomicidade:
 - Todas as operações em uma transação são executadas ou nenhuma será.

Atores em cena



- ❖ Administrador de banco de dados (DBA) é responsável por:
 - Autorizar o acesso ao banco de dados
 - Coordenar e monitorar seu uso
 - Adquirir recursos de software e hardware
- ❖ Projetistas de banco de dados são responsáveis por:
 - Identificar os dados a serem armazenados
 - Escolher estruturas apropriadas para representar e armazenar esses dados

Atores em cena (cont.)



- ❖ Analistas de sistemas
 - Identificam as necessidades dos usuários finais.
- ❖ Programadores de aplicações
 - Implementam essas especificações como programas.
- ❖ Usuários finais
 - Pessoas cujas funções exigem acesso ao banco de dados.

Trabalhadores dos bastidores



- ❖ **Projetistas e implementadores de sistema de SGBD**
 - Projetam e implementam os módulos e as interfaces do SGBD como um pacote de software.
- ❖ **Desenvolvedores de ferramentas**
 - Projetam e implantam ferramentas.
- ❖ **Operadores e pessoal de manutenção**
 - Responsáveis pela execução e manutenção do ambiente de hardware e software para o sistema de banco de dados.

Vantagens de usar a abordagem de SGBD



- ❖ Controlando a redundância
 - Normalização de dados
- ❖ Restringindo o acesso não autorizado
 - Subsistema de segurança e autorização
 - Software privilegiado
- ❖ Oferecer armazenamento persistente para objetos do programa
 - Objeto complexo em C++ pode ser armazenado de forma permanente em um SGBD orientado a objeto

Vantagens de usar a abordagem de SGBD (cont.)



- ❖ Oferecendo estruturas de armazenamento e técnicas de pesquisa para o processamento eficiente de consulta
 - Índices.
 - Buffering ou caching.
 - Processamento e otimização de consulta.

Vantagens de usar a abordagem de SGBD (cont.)



- ❖ Oferecendo backup e recuperação
 - Subsistema de backup e recuperação de SGBD é responsável pela recuperação.
- ❖ Oferecendo múltiplas interfaces do usuário
 - Interfaces gráficas do usuário (GUIs).
- ❖ Representando relacionamentos complexos entre dados
 - Pode incluir muitas variedades de dados que estão inter-relacionados de diversas maneiras.

Vantagens de usar a abordagem de SGBD (cont.)



❖ Impondo restrições de integridade

- Restrição de integridade referencial.
 - cada registro de turma deve estar relacionado a um registro de disciplina.
- Restrição de chave ou singularidade.
 - Cada registro de tabela deve ter um código único.
- Regras de negócio.

Vantagens de usar a abordagem de SGBD (cont.)



- ❖ Permitir ações usando regras
 - Gatilhos (Triggers)
 - Regra ativada por atualizações na tabela
 - Procedimentos armazenados (Stored Procedures)
 - Procedimentos mais elaborados para impor regras

Vantagens de usar a abordagem de SGBD (cont.)



- ❖ Implicações adicionais do uso da abordagem de banco de dados
 - Tempo reduzido para desenvolvimento de aplicação.
 - Flexibilidade (adequar às novas necessidades).
 - Disponibilidade de informações atualizadas.
 - Economias de escala.

Uma breve história das aplicações de banco de dados



- ❖ Antigas aplicações de banco de dados usando sistemas hierárquicos e de rede
 - Grande quantidade de registros com estrutura semelhante
 - Eram implantados em mainframes (\$\$\$) e usavam códigos complexos para consulta.
- ❖ Oferecer abstração de dados e flexibilidade de aplicação com bancos de dados relacionais
 - Separa o armazenamento físico dos dados de sua representação conceitual.
 - Utiliza uma linguagem mais simples e de alto nível (SQL).

Uma breve história das aplicações de banco de dados



- ❖ Aplicações orientadas a objeto e a necessidade de bancos de dados mais complexos
 - Inicialmente, era um potencial substituto dos bancos de dados relacionais.
 - Falta de padronização e complexidade contribuíram para este tipo de banco perder o mercado.
 - Hoje existem camadas de software que implementam a Orientação a Objetos no banco de dados relacional.
 - ADO Entity Framework, Hibernate, Nhibernate.

Uma breve história das aplicações de banco de dados



- ❖ Intercâmbio de dados na Web para comércio eletrônico usando XML
 - eXtended Markup Language (XML) é considerada o principal padrão para intercâmbio entre diversos tipos de bancos de dados e páginas Web

Uma breve história das aplicações de banco de dados



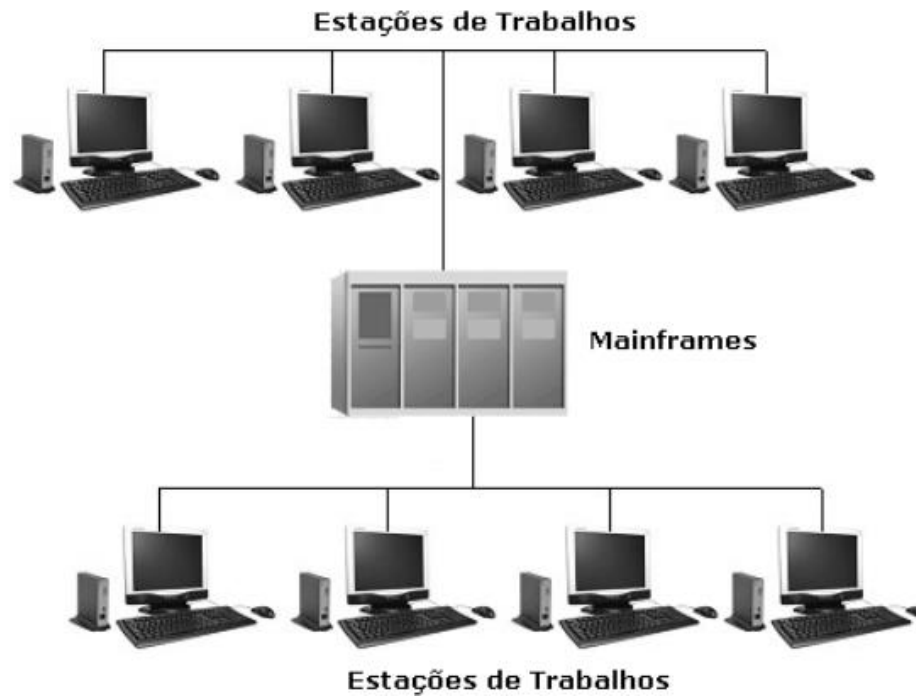
- ❖ Estendendo as capacidades do banco de dados para novas aplicações
 - Aplicações científicas.
 - Armazenamento de imagens.
 - Armazenamento de videos.
 - Data Mining (análisa e busca padrões em grandes massas de dados)

Arquiteturas de um SGBD



❖ Plataformas Centralizadas:

- Um computador de grande porte que é o hospedeiro do SGBD e o emulador para os aplicativos.
- Possui alto custo, e alto poder de processamento.



Arquiteturas de um SGBD



❖ Sistema de Computador Pessoal:

- O computador pessoal é o hospedeiro do SGBD e o cliente ao mesmo tempo.

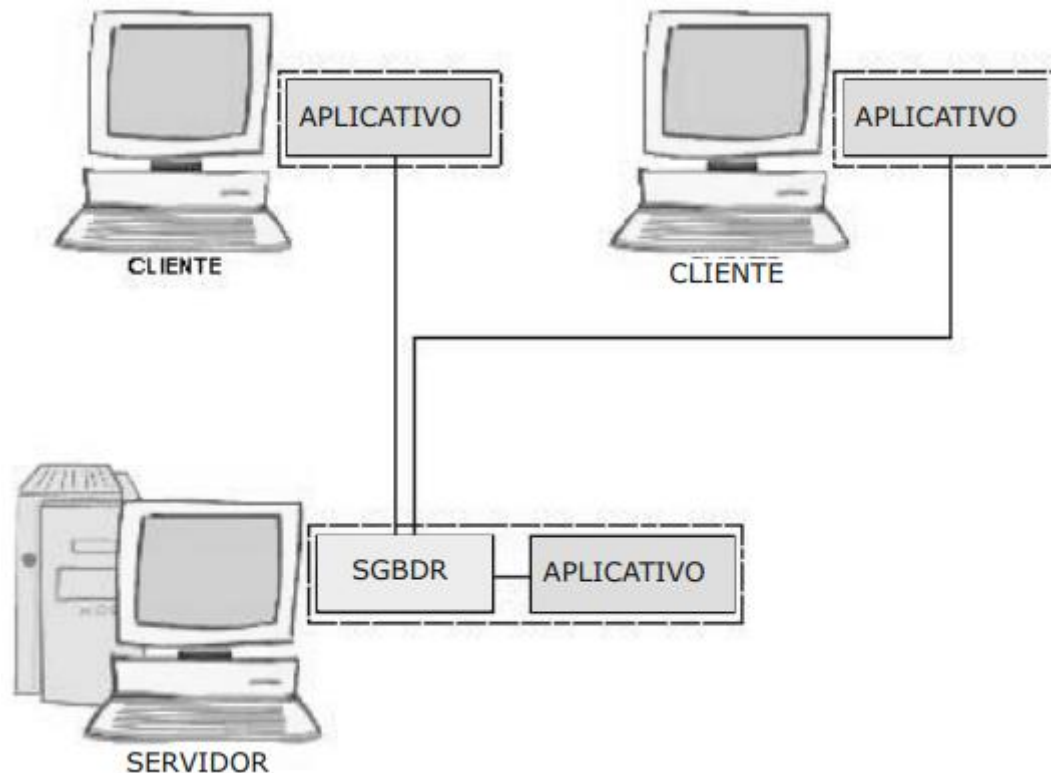


Arquiteturas de um SGBD



❖ Sistema de Cliente-Servidor:

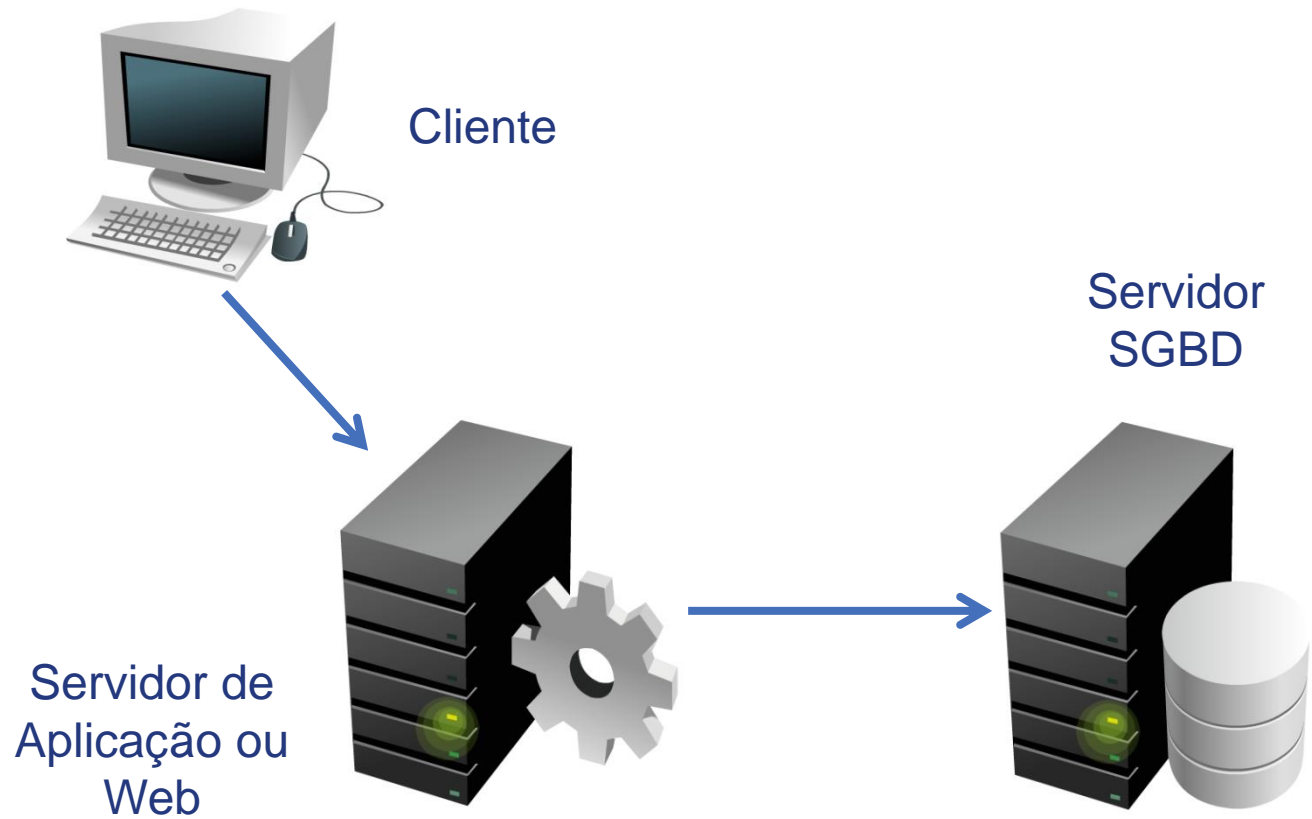
- O cliente executa as tarefas do aplicativo (interface gráfica), e o servidor executa o SGBD.



Arquiteturas de um SGBD



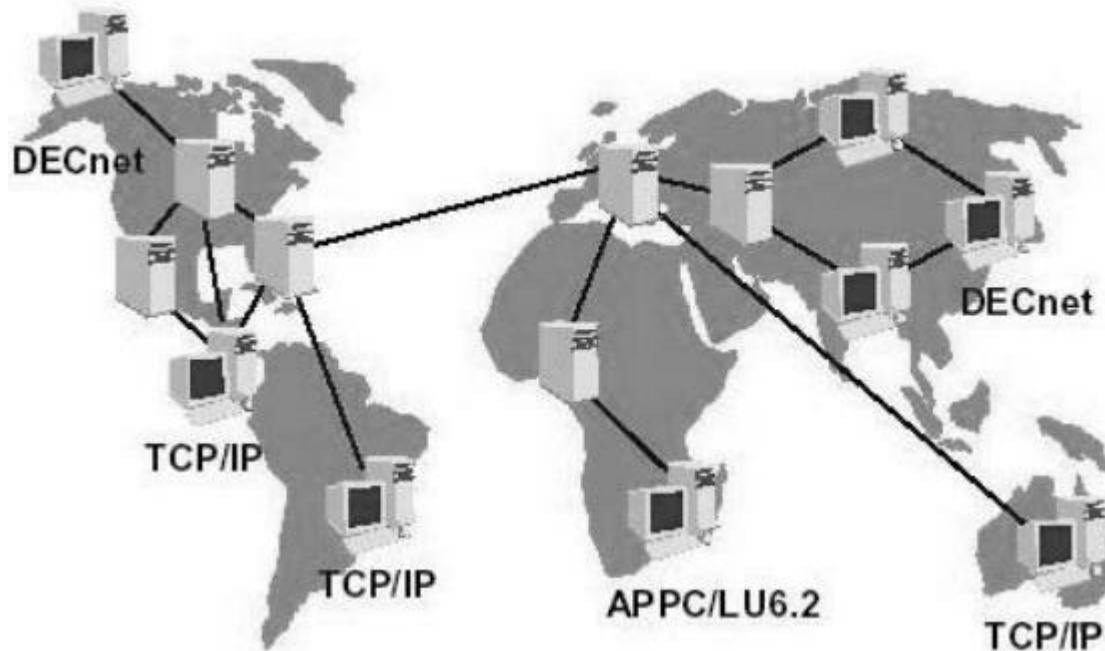
- ❖ Sistema de Cliente-Servidor N camadas:
 - Acrescenta uma camada (Aplicação) entre o cliente e o banco de dados.



Arquiteturas de um SGBD



- ❖ Banco de dados distribuídos:
 - A informação está distribuída em diversos servidores espalhados em locais diferentes.



Quando não usar um SGBD



- ❖ Mais desejável usar arquivos comuns sob as seguintes circunstâncias:
 - Aplicações de banco de dados simples e bem definidas, para as quais não se espera muitas mudanças.
 - Requisitos rigorosos, de tempo real, que podem não ser atendidos devido as operações extras executadas pelo SGBD (programas CAD, etc.).
 - Sistemas embarcados com capacidade de armazenamento limitada.
 - Nenhum acesso de múltiplos usuários aos dados.