

Bancos de Dados

Profa. Patrícia R. Oliveira
EACH / USP

Introdução a Bancos de Dados



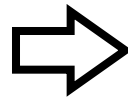
Introdução

- Sistemas de Banco de Dados (SBD's) são componentes essenciais no cotidiano da sociedade moderna.
- Exemplos de atividades do dia-a-dia que envolvem interação com bancos de dados:
 - transações de depósito ou retirada de dinheiro em caixas eletrônicos;
 - efetuação de reservas em hotéis ou em vôos;
 - consultas ao catálogo de uma biblioteca informatizada;
 - compra de produtos via web.

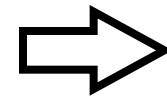


Introdução

Dados “úteis”
para o gerenciamento
da informação



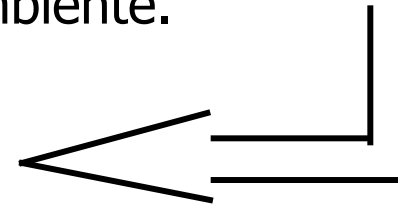
**COLETADOS E
ARMAZENADOS**



**BANCO DE
DADOS**

- A informação contida no BD deve representar um instante (estado) de uma determinada aplicação. Cada mudança no BD é um reflexo de um evento (ou sequência) que ocorre nesse ambiente.

***Modelo de uma parte do Mundo Real
(Universo de discurso ou Mini-mundo)***





Introdução

Sistema de Banco
de Dados →

- Sistema de informação computacional cujo objetivo principal é registrar e manter dados.
- Lida com BDs

- Um Sistema de Banco de Dados (SBD) possui 4 componentes:

 "Software"

 Usuários

 Dados

 "Hardware"

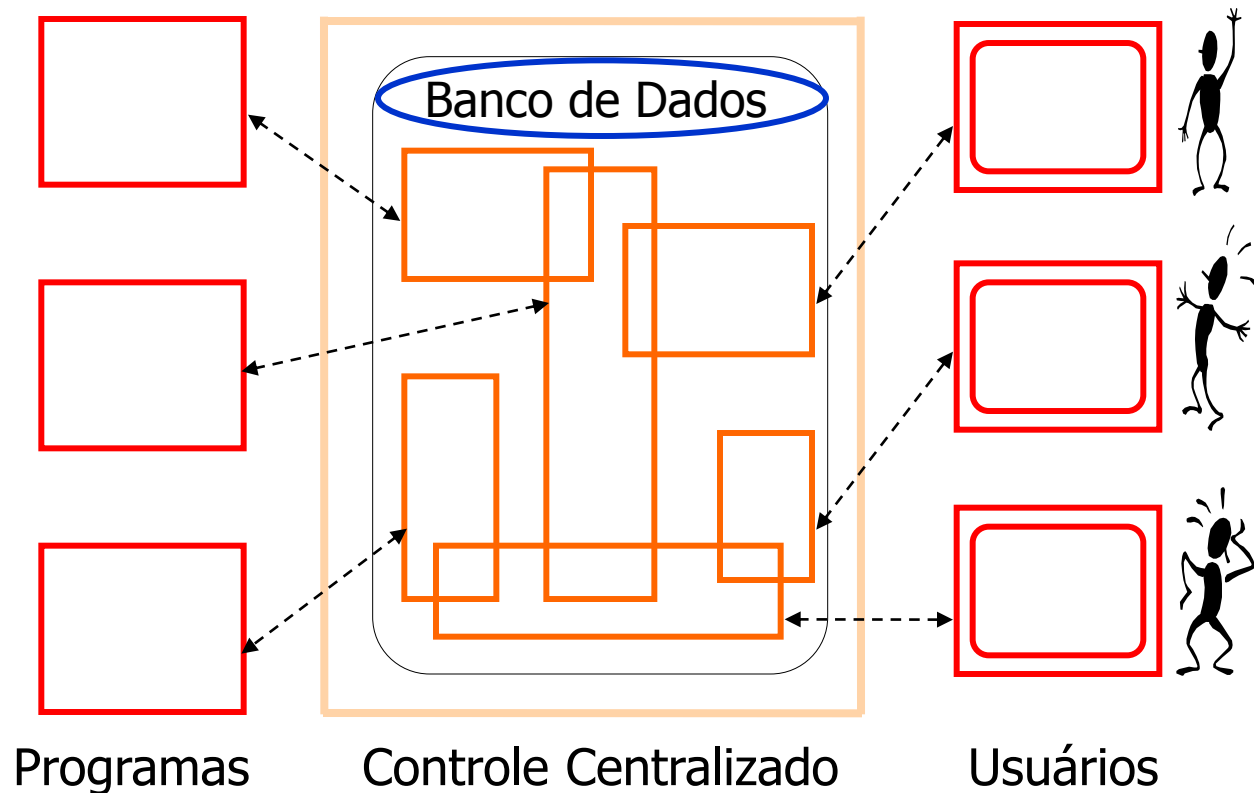


Importância dos Bancos de Dados

- A competitividade das empresas depende de dados precisos e atualizados.
- Conforme a empresa cresce, aumenta a sua dependência por dados abundantes e complexos.
- **Problema:** ferramentas de gerenciamento, extração rápida e precisa de informações tornam-se fundamentais.
- **Solução:** Sistema Gerenciador de Banco de Dados (SGBD).

Introdução

Sistema Gerenciador de Banco de Dados (SGBD)





Sistema Gerenciador de Bancos de Dados (SGBD)

- É uma coleção de programas que permite a criação e manutenção de um banco de dados.
- Facilita os processos de definição, construção, manipulação e compartilhamento de banco de dados entre vários usuários e aplicações.
- A **definição** de um banco de dados implica em especificar os tipos de dados, as estruturas e as restrições para os dados a serem armazenados.



Sistema Gerenciador de Bancos de Dados (SGBD)

- A **construção** de um banco de dados é o processo de armazenar os dados em alguma mídia apropriada, controlada pelo SGBD.
- A **manipulação** inclui algumas funções como:
 - pesquisas para recuperar um dado específico;
 - atualização do banco para refletir mudanças no mini-mundo;
 - geração de relatórios.



Sistema Gerenciador de Bancos de Dados (SGBD)

- O **compartilhamento** permite aos múltiplos usuários e programas acessar de forma concorrente o banco de dados.
- Outras funções importantes de um SGBD são:
 - a **proteção do sistema** quanto o mau funcionamento ou falhas (crashes) no hardware ou software;
 - a **segurança** contra acessos não autorizados;
 - a **manutenção** no sistema de banco de dados, permitindo a evolução dos requisitos ao longo do tempo.

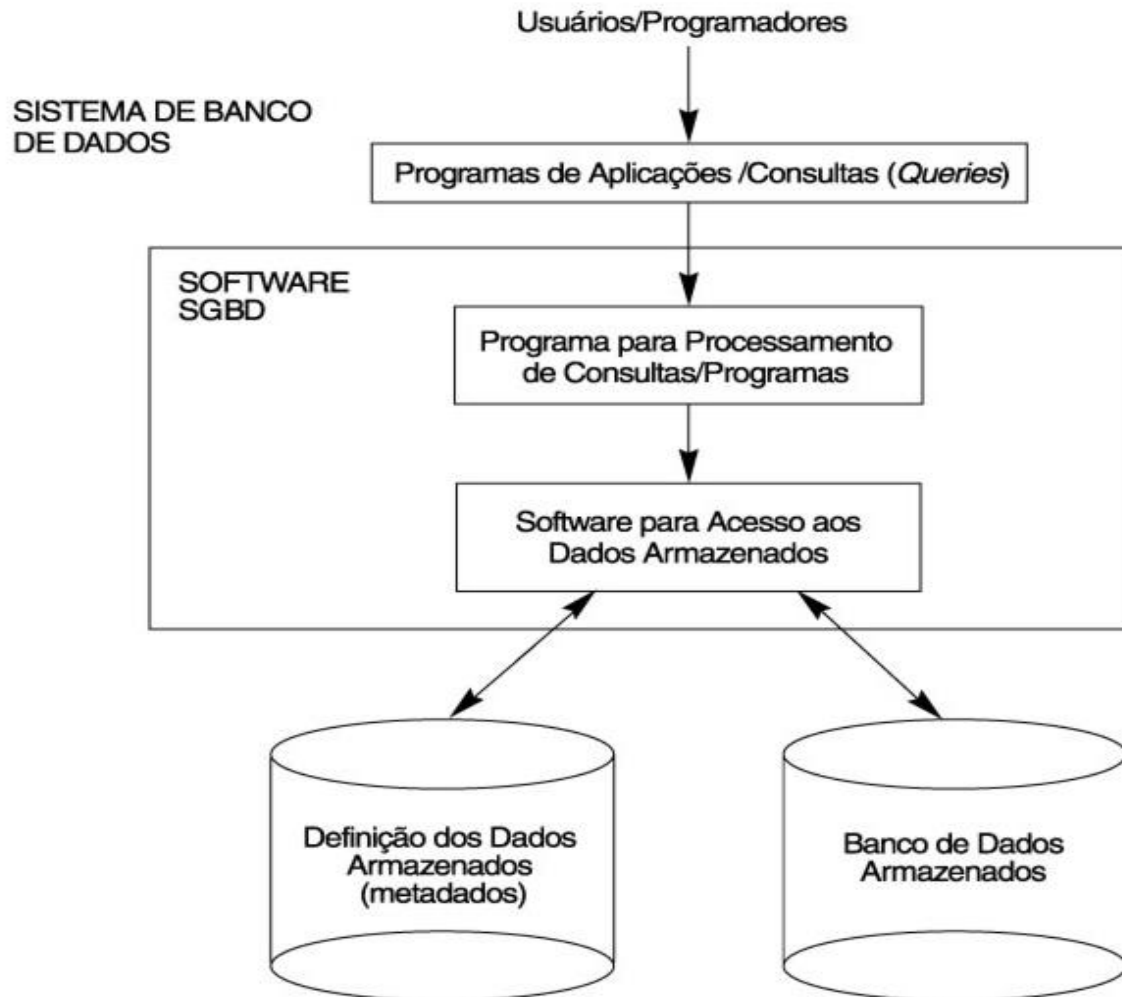


Sistema de Bancos de Dados (SBD)

- Definição: chama-se o banco de dados e o software SGBD, juntos, de **Sistema de Banco de Dados**.
- Em outras palavras,
Banco de Dados + SGBD = Sistema de Banco de Dados

Sistema de Bancos de Dados (SBD)

- Configuração



Uma perspectiva histórica

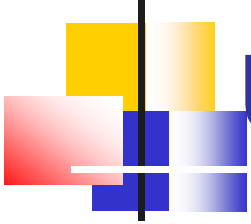
- No início da computação, programas tinham o único objetivo de armazenar e manipular dados.
- Esses programas gravavam dados em discos, usando estruturas próprias.
- Programas que não conhecessem as estruturas dos dados não poderiam utilizar esses dados.



Uma perspectiva histórica

- Se vários programas precisassem compartilhar os dados de um mesmo arquivo:
 - todos os programas teriam que conhecer e manipular as mesmas estruturas.



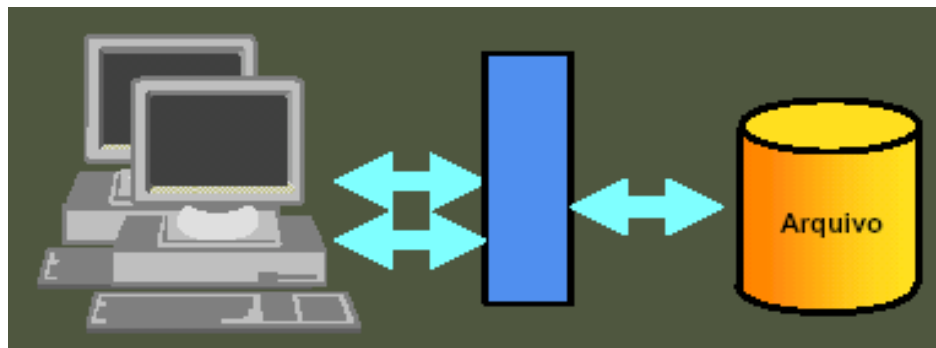


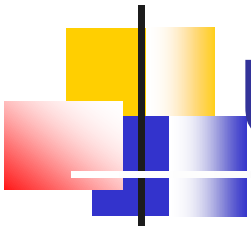
Uma perspectiva histórica

- Se algum programa precisasse realizar alguma mudança nas estruturas de dados:
 - todos os programas que acessam esse mesmo arquivo tinham que ser alterados;
- Problema: como garantir a unicidade das estruturas de dados entre os diversos programas devido à existência de redundância?

Uma perspectiva histórica

- Solução: criar um sistema intermediário que:
 - conheça as estruturas de dados do arquivo;
 - forneça apenas os dados que os programas precisam;
 - armazene adequadamente os dados de cada programa.



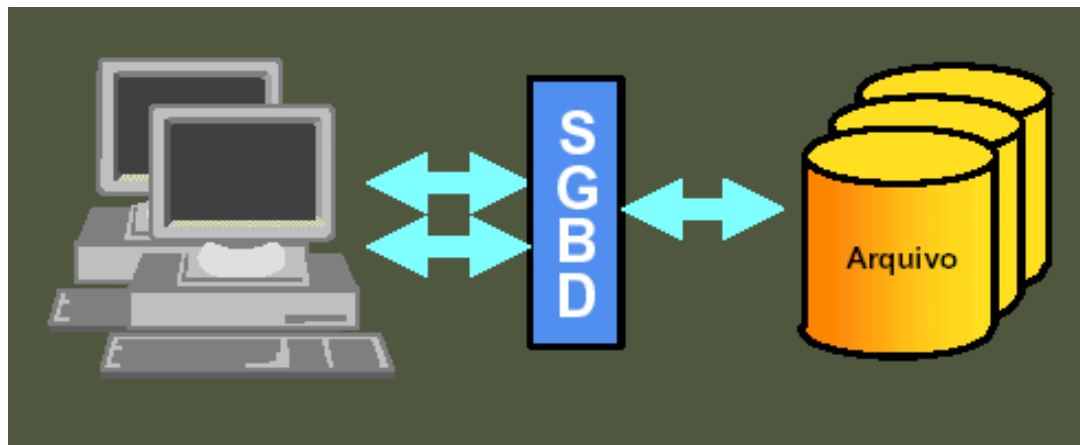


Uma perspectiva histórica

- Com esse sistema intermediário:
 - os programas “verão” somente os dados que lhes interessam;
 - os programas não precisam conhecer os detalhes de como seus dados estão gravados fisicamente;
 - os programas não precisam ser modificados se as estruturas de dados que manipulam o arquivo forem modificadas;
 - as alterações ficam concentradas no sistema intermediário.

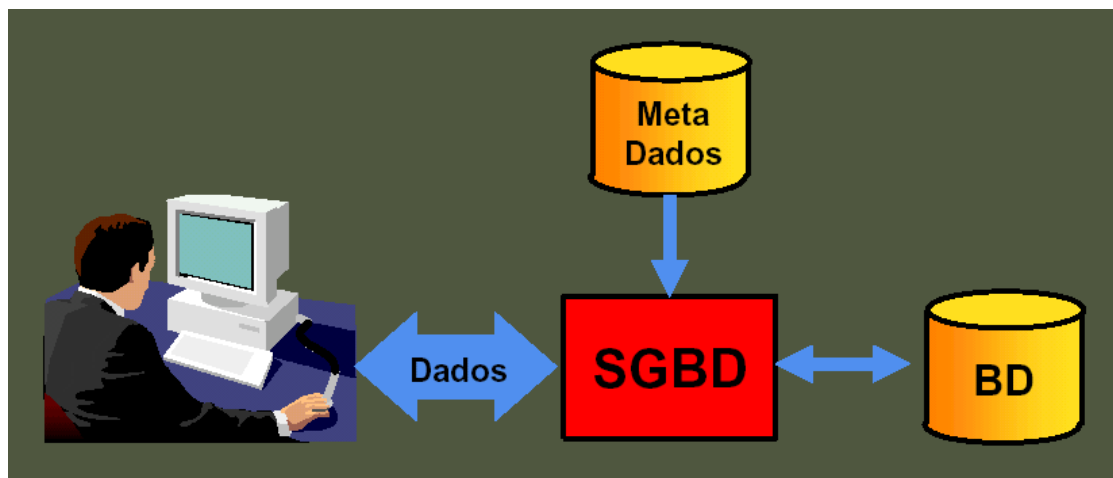
Uma perspectiva histórica

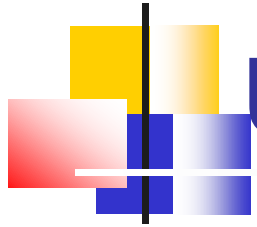
- Com o tempo, esse sistema intermediário passou a gerenciar vários arquivos;
- A essa coleção de arquivos foi dado o nome **Banco de Dados** e o sistema intermediário recebeu o nome de **Sistema Gerenciador de Banco de Dados (SGBD)**.



Uma perspectiva histórica

- O primeiro SGBD comercial surgiu em 1960
 - *Integrated Data Store* (IDS), desenvolvido pela GE.
- Com o tempo, surgiram padrões para descrever as estruturas de dados: os **modelos de dados**;
- A descrição de um banco de dados, segundo um modelo de dados é chamada de meta dados.





Uma perspectiva histórica

- Hoje, um banco de dados:
 - é uma coleção de dados coerentes e logicamente relacionados com algum significado associado;
 - é projetado, construído e constituído por dados que atendem a um propósito específico;
 - representa um aspecto do mundo real (mini-mundo).



Um exemplo

Banco de dados de uma universidade:

ALUNO	Nome	Numero	Turma	Curso_Hab
	Smith	17	1	CC
	Brown	8	2	CC

CURSO	NomedoCurso	NumerodoCurso	Creditos	Departamento
	Introdução à Ciência da Computação	CC1310	4	CC
	Estruturas de dados	CC3320	4	CC
	Matemática Discreta	MAT2410	3	MATH
	Banco de dados	CC3380	3	CC

DISCIPLINA	IdentificadordeDisciplina	NumerodoCurso	Semestre	Ano	Instrutor
	85	MAT2410	Segundo Semestre	98	King
	92	CC1310	Segundo Semestre	98	Anderson
	102	CC3320	Primeiro Semestre	99	Knuth
	112	MAT2410	Segundo Semestre	99	Chang
	119	CC1310	Segundo Semestre	99	Anderson
	135	CC3380	Segundo Semestre	99	Stone

HISTORICO_ESCOLAR	NumerodoAluno	Identificador_Disciplinas	Nota
	17	112	B
	17	119	C
	8	85	A
	8	92	A
	8	102	B
	8	135	A

PRE_REQUISITO	NumerodoCurso	NumerodoPre_requisito
	CC3380	CC3320
	CC3380	MAT2410
	CC3320	CC1310



Um exemplo

- Arquivo **ALUNO**: conserva os dados de cada estudante na universidade:

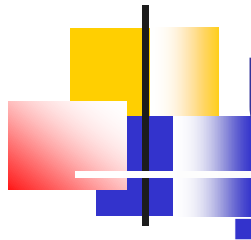
ALUNO	Nome	Numero	Turma	Curso_Hab
	Smith	17	1	CC
	Brown	8	2	CC



Um exemplo

- Arquivo **CURSO**: preserva os dados sobre cada curso:

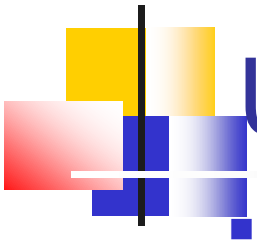
CURSO	NomedoCurso	NumerodoCurso	Creditos	Departamento
	Introdução à Ciência da Computação	CC1310	4	CC
	Estruturas de dados	CC3320	4	CC
	Matemática Discreta	MAT2410	3	MAT
	Bancos de Dados	CC3360	3	CC



Um exemplo

Arquivo **DISCIPLINA**: guarda os dados de cada disciplina do curso:

DISCIPLINA	Identificador deDisciplina	NumerodoCurso	Semestre	Ano	Instrutor
	85	MAT2410	Segundo Semestre	98	King
	92	CC1310	Segundo Semestre	98	Anderson
	102	CC3320	Primeiro Semestre	99	Knuth
	112	MAT2410	Segundo Semestre	99	Chang
	119	CC1310	Segundo Semestre	99	Anderson
	136	CC3380	Segundo Semestre	99	Stone



Um exemplo

Arquivo **HISTORICO_ESCOLAR**: mantém as notas recebidas por aluno nas diversas disciplinas cursadas:

HISTORICO_ESCOLAR	NumerodoAluno	Identificador_Disciplina	Nota
	17	112	B
	17	119	C
	8	85	A
	8	92	A
	8	102	B
	8	135	A



Um exemplo

- Arquivo **PRE_REQUISITO**: armazena os pré-requisitos de cada curso:

PRE_REQUISITO	NumerodoCurso	NumerodoPre_Requisito
	CC3380	CC3320
	CC3380	MAT2410
	CC3320	CC1310



Um exemplo

- Para definir esse banco de dados, devemos:
 - especificar a estrutura de cada registro, em cada arquivo;
 - considerar os diferentes tipos de elementos dos dados a serem armazenados em cada registro.



Um exemplo

- Cada registro ALUNO inclui os dados:
 - NomeDoAluno
 - NumeroDoAluno
 - Turma
 - calouro, ou 1
 - veterano, ou 2
 - ...
 - CursoHabilitação
 - Matemática, ou MAT
 - Ciência da Computação, ou CC



Um exemplo

- Cada registro CURSO apresenta dados como:
 - NomedoCurso
 - NumerodoCurso
 - Créditos
 - Departamento (que oferece o curso)



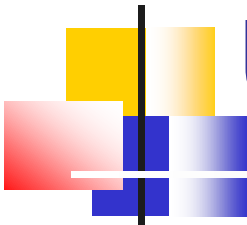
Um exemplo

- É necessário, ainda, especificar os tipos de dados para cada elemento de dados em um registro, como:
 - NomedoAluno em ALUNO é uma string (cadeia de caracteres alfabéticos);
 - NumerodoAluno em ALUNO é um inteiro;
 - Nota em HISTORICO_ESCOLAR é um caractere único no conjunto {A, B, C, D, E, F, I};



Um exemplo

- Para **construir** o banco de dados UNIVERSIDADE:
 - armazena-se os dados que representam cada aluno, curso, disciplina, relatório de notas e pré-requisitos em cada registro em seu arquivo apropriado.
- Obs: os registros em diferentes arquivos podem estar relacionados.



Um exemplo

ALUNO	Nome	Numero	Turma	Curso_Hab
	Smith	17	1	CC
	Brown	8	2	CC

HISTORICO_ESCOLAR	NumerodoAluno	Identificador_Disciplina	Nota
<p>O registro para "Smith" no arquivo ALUNO está relacionado a dois registros no arquivo HISTORICO_ESCOLAR</p>	17	112	B
	17	119	C
	8	85	A
	8	92	A
	8	102	B
	8	135	A



Um exemplo

- A manipulação do banco de dados envolve operações de:
 - consulta (*query*)
 - atualização
- Exemplos de consultas:
 - a recuperação do histórico escolar de 'Smith';
 - a relação dos nomes de alunos que cursaram a disciplina do curso de Banco de Dados no segundo semestre de 2009;



Um exemplo

- Exemplos de atualizações:
 - mudar a turma de 'Smith' para veteranos;
 - criar uma nova disciplina para o curso de Banco de Dados nesse semestre;
 - colocar a nota A para 'Smith' na disciplina do curso Banco de Dados no último semestre.



Processamento de Arquivos *versus* Banco de Dados

- No tradicional processamento de arquivos:
 - cada usuário define e implementa os arquivos necessários para uma aplicação específica;
 - os arquivos são parte do programa da aplicação.



Exemplo de aplicações com processamento de arquivos

- O **usuário 1**, a *secretaria de notas*, pode manter um arquivo para os alunos e suas notas.
 - os programas para imprimir o histórico de um aluno e colocar novas notas no arquivo são implementados como parte da aplicação.
- O **usuário 2**, o *departamento de contabilidade*, pode manter um arquivo para os alunos e suas mensalidades (pagamentos).



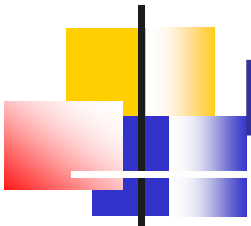
Exemplo de aplicações com processamento de arquivos

- **Conclusão:** apesar dos usuários 1 e 2 precisarem dos dados sobre os alunos, cada um deles:
 - mantém suas informações em arquivos separados;
 - implementam seus próprios programas para manipular os dados nos arquivos.



Processamento de Arquivos *versus* Banco de Dados

- Resultados indesejados da abordagem com arquivos:
 - redundâncias na definição e armazenamento dos dados;
 - desperdício de espaço de armazenamento;
 - esforços redundantes para manter os dados comuns atualizados.
- Na abordagem que utiliza um banco de dados:
 - Um único repositório de dados é definido uma única vez, mantido e, então, acessado por vários usuários.

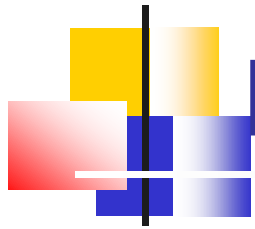


Por que utilizar SBD's?

- Principal argumento: um SBD proporciona à empresa um controle centralizado dos dados.
 - Em um sistema de arquivos, o controle dos dados pode tornar-se difícil.

Outras vantagens importantes:

- 1) Em um SBD, a redundância dos dados pode ser reduzida.
 - Em um sistema de arquivos, as informações podem estar dispersas, o que provoca a redundância.

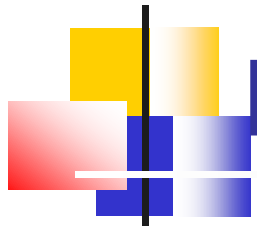


Por que utilizar SBD's?

2) Em um SBD, é possível evitar inconsistências nos dados, como consequência direta da redução de redundância.

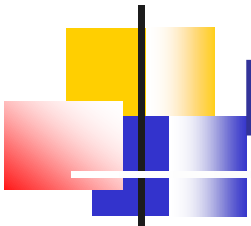
Ou seja,

Redundância controlada	→	Propagação das atualizações
Redundância não controlada	→	Inconsistência



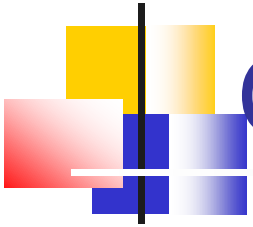
Por que utilizar SBD's?

- 3) Em um SBD, os dados podem ser compartilhados por diferentes aplicações.
 - Novas aplicações podem ser implementadas sem que se tenha que criar novos arquivos.
- 4) É possível aplicar restrições de segurança a um SBD, das seguintes formas:
 - garantindo que as únicas vias de acesso sejam por meio dos canais adequados;
 - definindo a execução da verificação de acesso à tentativa de acessar dados sensíveis.



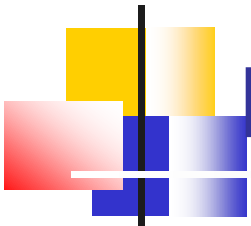
Limitações de um SBD

- As limitações de um SBD referem-se principalmente aos custos incrementais e vulnerabilidade do sistema:
 - vulnerabilidade → natureza centralizada do sistema;
 - o processo de recuperação de falhas é mais custoso;
 - generalidade que um SGBD fornece para definir e processar os dados;
 - os custos de hardware, software e programadores são mais altos.



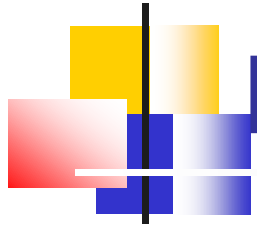
Quando utilizar arquivos?

- Quando o banco de dados e as aplicações forem simples, bem definidas e estáveis;
- Quando os requisitos de eficiência em tempo real forem altos;
- Quando os requisitos de acesso forem mono-usuário.



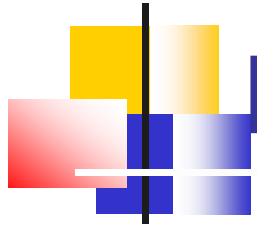
Principais características dos SBD's

- Principais características da abordagem de banco de dados:
 - natureza auto-descritiva do Sistema de Banco de Dados (SBD);
 - isolamento entre os programas e dados e a abstração dos dados;
 - suporte para múltiplas visões dos dados;
 - compartilhamento de dados e suporte a transações multi-usuários.



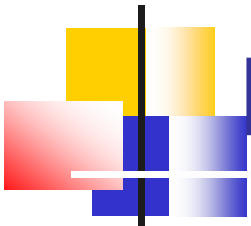
Natureza auto-descritiva do SBD

- O SBD não possui apenas o banco de dados, mas também:
 - uma completa definição ou descrição da estrutura do banco de dados;
 - restrições.



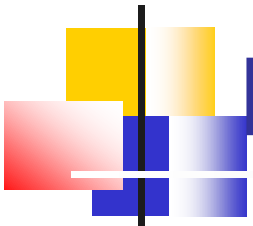
Natureza auto-descritiva do SBD

- A definição do banco de dados está armazenada no catálogo do SGBD, que contém:
 - a estrutura de cada arquivo;
 - o tipo e o formato de cada item de dado;
 - restrições sobre os dados.
- A informação armazenada no catálogo é chamada de metadados.



Natureza auto-descritiva do SBD

- O catálogo é usado:
 - pelo software SGBD;
 - pelos usuários
 - para buscar informações sobre a estrutura do banco de dados;
- Importância do catálogo:
 - O SGBD não é feito para uma aplicação específica
 - é necessário acessar o catálogo para conhecer os tipos e os formatos dos dados.



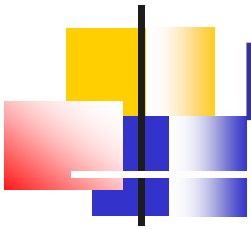
Natureza auto-descritiva do SBD

- Descrição dos dados no processamento tradicional de arquivos:
 - faz parte dos próprios programas da aplicação;
 - os programas estão restritos a trabalhar com um único banco de dados específico:
 - a estrutura do banco de dados declarada no programa da aplicação.
- Ex: Uma aplicação em C++ pode ter declarações de classes ou estruturas (structs).



Exemplo

- No banco de dados UNIVERSIDADE, o catálogo do SGBD armazena as definições de todos os arquivos:
 - ALUNO;
 - CURSO;
 - DISCIPLINA;
 - HISTORICO_ESCOLAR;
 - PRE_REQUISITO.



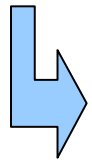
Natureza auto-descritiva do SBD

- As descrições dos arquivos em um SBD são:
 - especificadas pelo projetista de banco de dados;
 - e armazenadas no catálogo do SGBD.



Isolamento entre os programas e dados e abstração de dados

- Na abordagem tradicional de arquivos, mudanças na estrutura de um arquivo pode exigir alterações em todos os programas que acessam esse arquivo.
- Programas para acesso ao SGBD não exigem essas alterações.



■ Pergunta: Por que isso acontece?



Isolamento entre os programas e dados e abstração de dados

- Resposta: Em um SBD, a estrutura dos arquivos de dados são armazenadas no catálogo do SGBD, separadamente dos programas de aplicação.
- À essa propriedade de um SBD dá-se o nome de independência programa-dados.



Isolamento entre os programas e dados e abstração de dados

- A característica que permite a independência programa-dados é chamada de abstração de dados.
- Um SGBD oferece uma representação conceitual (modelo) dos dados
 - não inclui muitos detalhes de como os dados são armazenados.



Isolamento entre os programas e dados e abstração de dados

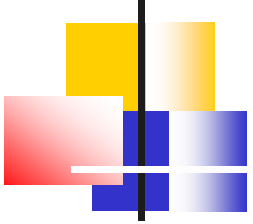
- O modelo de dados utiliza conceitos lógicos:
 - objetos;
 - propriedades dos objetos;
 - inter-relacionamentos entre objetos.
- Pergunta: por que o modelo de dados é importante?



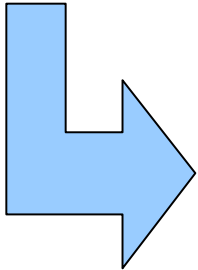
Isolamento entre os programas e dados e abstração de dados

- Resposta: é mais fácil para o usuário entender conceitos lógicos do que conceitos de armazenamento computacionais.
- O modelo de dados esconde detalhes de armazenamento e implementação.

Suporte para múltiplas visões dos dados



- Um banco de dados típico tem muitos usuários.



- Cada qual pode solicitar diferentes perspectivas, ou visões, do banco de dados.



Suporte para múltiplas visões dos dados

- Uma visão pode:
 - ser um subconjunto do banco de dados;
 - conter uma visão virtual dos dados:
 - dados derivados dos arquivos do banco de dados mas não explicitamente armazenados.



Exemplo

- Duas visões derivadas do banco de dados UNIVERSIDADE:

(a)

HISTORICO_ESCOLAR	Nome do Aluno	Histórico Escolar do Aluno				
		Número do Curso	Nota	Semestre	Ano	Id Disciplina
	Smith	CC1310	C	Outono	99	119
		MAT2410	B	Outono	99	112
	Brown	MAT2410	A	Outono	98	85
		CC1310	A	Outono	98	92
		CC3320	B	Primavera	99	102
		CC3380	A	Outono	99	135

(b)

PRE_REQUISITOS	Nome do Curso	Número do Curso	Pre_Requisitos
	Banco de Dados	CC3380	CC3320
			MAT2410
	Estruturas de Dados	CC3320	CC1310



Compartilhamento de dados e Processamento de transação multiusuários

- Vários usuários podem acessar o banco de dados ao mesmo tempo.
- O SGBD deve prover um controle de concorrência:
 - garante que vários usuários, ao tentar usar o mesmo dado, o façam de forma controlada;
 - assegurar resultados de atualizações corretos;



Exemplo

- Quando muitos atendentes tentam reservar um lugar em um voo, o SGBD deve:
 - garantir que cada assento possa ser acessado somente por um atendente de cada vez.



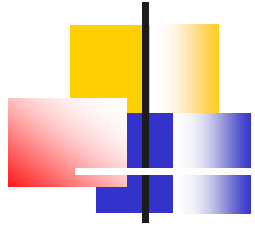
Compartilhamento de dados e Processamento de transação multiusuários

- Uma transação é um programa em execução ou processo que inclui um ou mais acessos ao banco de dados, como:
 - leitura de registros;
 - atualização de registros.



Compartilhamento de dados e Processamento de transação multiusuários

- Cada transação deve executar um acesso logicamente correto ao banco de dados
 - para isso, deve ser executada sem a interferência de outras transações.
- A propriedade de isolamento garante que cada transação possa ser efetuada de forma isolada de outras transações.



Leitura

- **Elmasri, R.; Navathe, S.B.** "Sistemas de Banco de Dados", Pearson Education do Brasil, 4a. Edição, 2005.
- Capítulo 1: Bancos de Dados e Usuários de banco de dados.