

[MAC0439] Laboratório de Bancos de Dados  
Aula 1  
Introdução aos Sistemas de Bancos de Dados

Kelly Rosa Braghetto

DCC-IME-USP

12 de agosto de 2015

# Bancos de Dados – Introdução

# O que é um **banco de dados**?

- ▶ **Banco de dados** – coleção de dados relacionados
- ▶ **Dados** – fatos conhecidos que podem ser registrados e que possuem significado implícito

Problema: essa definição é genérica demais!

# Propriedades implícitas de um banco de dados (BD)

1. Representar (geralmente!) algum aspecto do mundo real = **minimundo** ou **UoD** (Universo de Discurso)  
As mudanças no minimundo são refletidas no BD
2. Ser uma **coleção lógica e coerente de dados** com algum **significado inerente**  
Uma coleção “aleatória” de dados não é um BD!
3. Ser projetado, construído e povoado com dados que possuem um objetivo específico  
Um BD deve possuir um grupo provável de usuários e algumas aplicações pré-concebidas, nas quais esses usuários estão interessados

# Propriedades implícitas de um banco de dados (BD)

**Resumindo:** um BD possui alguma **fonte** (de onde os dados são derivados), algum grau de **interação com eventos do mundo real** e um **público** que está ativamente interessado no conteúdo do BD

## Outras características:

- ▶ BDs têm complexidade e tamanho variáveis
- ▶ BDs podem ser informatizados ou mantidos manualmente

# Exemplo da dimensão que um BD pode assumir

## Facebook (dados de abril de 2014)

- ▶ *Data warehouse* com mais de 300 PB (petabytes)
- ▶ Diariamente, cerca de 600 TB (terabytes) de novos dados
- ▶ Mais de 1 bilhão de usuários ativos
- ▶ Grande variedade de aplicações: desde do tradicional processamento em lotes até a análise de grafos (redes), aprendizagem de máquina e análise interativa em tempo real.
- ▶ Em 2013, o quantidade de dados armazenados no *data warehouse* **triplicou**

**1 petabyte = 1.000 terabytes = 1 quadrilhão de bytes  
(~ 210.000 DVDs)**

Fonte:

<https://code.facebook.com/posts/229861827208629/scaling-the-facebook-data-warehouse-to-300-pb/>

# Softwares para a manutenção de bancos de dados

Um BD informatizado pode ser criado e mantido por:

- ▶ um grupo de programas de aplicação (criados especificamente para essa tarefa)
- ▶ um **Sistema de Gerenciamento de Banco de Dados (SGBD)**

Sistema de software de propósito geral que facilita o processo de **definição, construção, manipulação e compartilhamento** de BDs entre vários usuários e aplicações

# Sistema de Gerenciamento de Banco de Dados (SGBD)

## Apoia o ciclo de vida de um BD:

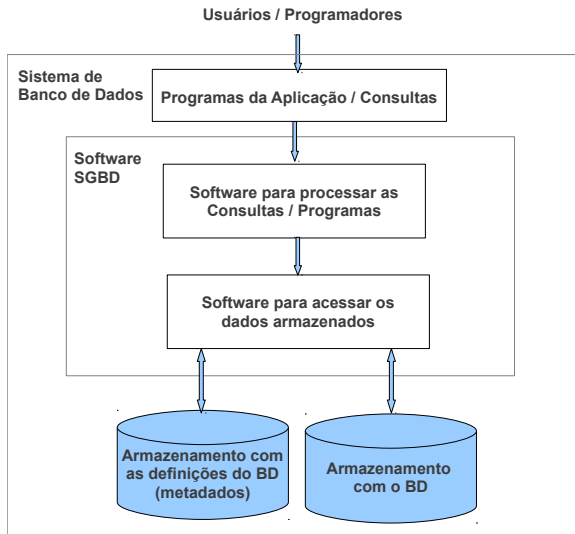
- ▶ **Definir um BD**  $\Rightarrow$  especificar os tipos, as estruturas e as restrições para os dados que serão armazenados no BD
- ▶ **Construir um BD**  $\Rightarrow$  gravar os dados em algum meio de armazenamento (controlado pelo SGBD)
- ▶ **Manipular um BD**  $\Rightarrow$  realizar funções como consultas ao BD para recuperar dados específicos, atualizar o BD para refletir mudanças no minimundo, etc.
- ▶ **Compartilhar um BD**  $\Rightarrow$  permitir que múltiplos usuários e programas acessem-no simultaneamente



## Outras funções importantes de um SGBD

- ▶ **Proteger os dados** – contra falhas de hardware ou software e contra acessos não autorizados ou maliciosos
- ▶ **Manter os dados por um longo período de tempo** – permitindo que o sistema evolua acompanhando as mudanças dos requisitos ao longo do tempo

# Sistema de Banco de Dados = Banco de Dados + SGBD



## O acesso a um BD

Um **programa de aplicação** é um programa que acessa um banco de dados enviando **consultas** ou **transações** para o SGBD.

- ▶ **Consulta** – comando que recupera dados do BD
- ▶ **Transação** – comando que lê ou escreve dados do/no BD

## Banco de dados × processamento de arquivos tradicional

- ▶ No **processamento de arquivos tradicional**: cada usuário define e implementa os arquivos necessários para uma aplicação específica (como parte da aplicação)

**Problema: possível redundância na definição e armazenamento de dados** ⇒ desperdício de espaço e trabalho redundante na manutenção de dados comuns a mais de uma aplicação

- ▶ Nos **bancos de dados**: um único repositório de dados é mantido

Uma vez definido, o repositório passa a ser acessado por diversos usuários e aplicações

## Outras características importantes de BDs mantidos em SGBDs tradicionais

- ▶ **Natureza autodescritiva**  
BDs são mantidos com uma descrição completa de sua estrutura e restrições (metadados)
- ▶ **Isolamento entre programas e dados (por meio de **abstração de dados**)**  
Um SGBD oferece uma representação conceitual dos dados que não inclui muitos detalhes sobre como eles são armazenados fisicamente

## Outras características importantes de BDs mantidos em SGBDs tradicionais

- ▶ Suporte a visões múltiplas dos dados  
Uma visão pode ser um subconjunto do BD ou conter dados que são derivados dos dados armazenados
- ▶ Compartilhamento de dados e processamento de transações multiusuários  
O SGBD faz controle de concorrência, para garantir que o resultado de tentativas de atualizações simultâneas sobre um mesmo dado resultem em algo correto

## Vantagens do uso de um SGBD

1. Controle de redundâncias
2. Restrição do acesso não autorizado
3. Armazenamento persistente para objetos de programas e estruturas de dados
4. Estruturas de armazenamento e técnicas de busca para o processamento eficiente de consultas
5. Mecanismos de *backup* e recuperação
6. Múltiplas interfaces de usuário
7. Capacidade de representação de relacionamentos complexos entre dados
8. Imposição de restrições de integridade
9. Possibilidade de deduzir dados e executar ações por meio de regras

# Vantagens do uso de um SGBD

## Implicações adicionais:

1. Potencial para garantir padrões
2. Redução no tempo de desenvolvimento de aplicações
3. Flexibilidade
4. Disponibilidade de informações atualizadas
5. Economias de escala



# Analisando a viabilidade do uso de um SGBD

Os custos (de tempo e dinheiro) envolvidos no uso de um SGBD se relacionam a:

1. Os altos investimentos iniciais em hardware, software e treinamento
2. A generalidade que o SGBD fornece para a definição e o processamento de dados
3. O esforço adicional necessário para prover segurança, controle de concorrência, recuperação e integridade dos dados

## Quando é melhor **não** usar um SGBD [convencional]

O uso direto de arquivos ou de SGBDs “não-convencionais” é mais aconselhado que o uso de SGBDs tradicionais nas seguintes situações:

1. O BD e suas aplicações são simples, bem definidos e sem previsão de mudanças
2. A sobrecarga do SGBD pode impedir que requisitos de desempenho (como em programas de tempo-real) sejam atendidos
3. O acesso de múltiplos usuários aos dados não é necessário
4. O dados não cabem em uma única máquina

# Modelos de Dados

# Abstração de dados

Oferecer abstração de dados é uma característica fundamental dos bancos de dados, ocultando detalhes sobre a organização e armazenamento dos dados (detalhes esses que são desnecessários para a maioria dos usuários de BDs)

A abstração é feita por meio de modelos de dados:

- ▶ **Modelo de dados** – é um conjunto de conceitos que podem ser usados para descrever a *estrutura* de um banco de dados
  - ▶ Modelos também podem incluir *operações* básicas para a recuperação e atualização de dados do banco
- ▶ **Estrutura de um banco de dados** – define os tipos de dados, relacionamentos e restrições que se aplicam aos dados

# Um “parênteses” sobre a especificação de operações em BDs

- ▶ Além das operações básicas, agora é comum que modelos de dados ofereçam mecanismos para a especificação do aspecto dinâmico (= comportamento) de uma aplicação de BD
- ▶ O projetista do BD pode especificar operações definidas pelos usuários, permitidas sobre os objetos do BD
- ▶ Esses mecanismos são particularmente importantes para os modelos de dados de objetos, mas também vêm sendo incorporados aos modelos de dados mais tradicionais

**Tendência atual:** combinar as atividades de projeto de banco de dados às de projeto de software

# Categorias de modelos de dados

- ▶ **Modelos de dados conceituais** (ou **de alto nível**) – possuem conceitos que descrevem os dados como os usuários os percebem
- ▶ **Modelos de dados físicos** (ou **de baixo nível**) – possuem conceitos que descrevem os detalhes de como os dados estão armazenados no computador, com informações como o formato de um registro, as ordenações dos registros e os caminhos de acesso <sup>1</sup>
- ▶ **Modelos de dados representativos** (ou **de implementação**) – estão a um meio termo dos dois anteriores, pois podem ser entendidos pelos usuários finais, ao mesmo tempo em que não estão tão distantes da forma como os dados são organizados e armazenados no computador

---

<sup>1</sup>Estrutura que torna eficiente a busca por registros em um BD particular

# Modelos de dados conceituais

Possuem conceitos tais como:

- ▶ **Entidade:** representa um objeto ou conceito do mundo real (ex.: aluno, disciplina, turma, ...)
- ▶ **Atributo:** representa alguma propriedade que ajuda a descrever uma entidade (ex.: nome do aluno, número USP, código da disciplina, ...)
- ▶ **Relacionamento:** mostra uma associação entre duas ou mais entidades (ex.: aluno pertence a uma turma)

⇒ Exemplo de modelo conceitual bastante popular: **modelo entidade-relacionamento**

# Modelos de dados representativos

- ▶ Os modelos representativos são os mais usados nos SGBDs comerciais tradicionais
- ▶ Exemplos: o **modelo de dados relacional**, o **modelo de redes** e o **modelo hierárquico**  
Esses modelos são denominados de **modelos de dados baseados em registros** (pois mostram os dados usando estruturas de registros)
- ▶ O **modelo de dados de objeto** pode ser considerado como um modelo representativo de mais alto nível, mas também é bastante usado como modelo conceitual (especialmente na área de Engenharia de Software).



# Descrição do BD × BD de fato

## Esquema do BD

⇒ É a descrição do BD

- ▶ É definido durante a fase de projeto e espera-se que não seja alterado com frequência
- ▶ Geralmente, é representado por um **diagrama**, que mostra somente alguns aspectos do esquema (nome dos registros e itens de dados, alguns tipos de restrições)

## Descrição do BD × BD de fato

### Estado do BD (ou *snapshot*, ou *conjunto de instâncias*)

⇒ São os dados no BD em um determinado momento no tempo

- ▶ Toda inserção ou remoção de um registro, ou alteração do valor de um item de dado gera uma mudança de estado no BD
- ▶ O SGBD é parcialmente responsável por garantir que cada novo estado do BD é um estado válido (= que respeita a estrutura e as restrições definidas no esquema)
- ▶ O SGBD armazena em seu catálogo as descrições das construções do esquema e suas restrições (= metadados) e, dessa forma, pode acessá-los sempre que necessário

## Exemplo de diagrama de esquema

### ALUNO

Nome	Numero_aluno	Tipo_aluno	Curso
------	--------------	------------	-------

### DISCIPLINA

Nome_disciplina	Numero_disciplina	Creditos	Departamento
-----------------	-------------------	----------	--------------

### PRE\_REQUISITO

Numero_disciplina	Numero_pre_requisito
-------------------	----------------------

### TURMA

ID_turma	Numero_disciplina	Semestre	Ano	Professor
----------	-------------------	----------	-----	-----------

### HISTORICO\_ESCOLAR

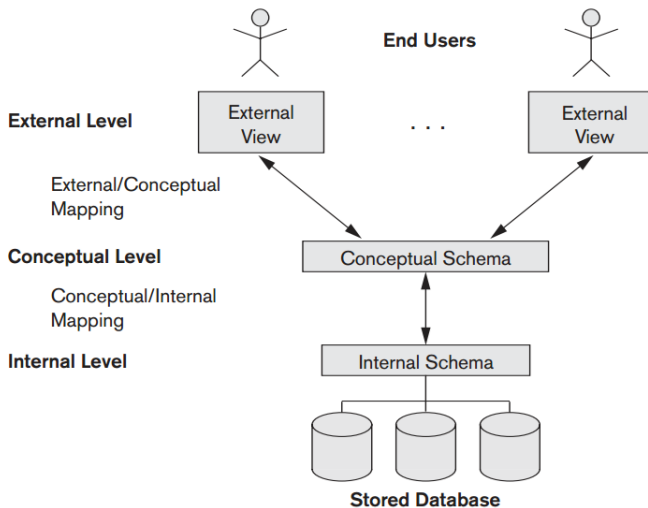
Numero_aluno	ID_turma	Nota
--------------	----------	------

# Um “parênteses” sobre a evolução do esquema de um BD

- ▶ Embora o esquema de um BD não deva mudar frequentemente, não é raro que ele sofra alterações ao longo do tempo
- ▶ Isso é chamado de **evolução do esquema**
- ▶ A maioria dos SGBDs modernos possui operações que permitem que um esquema seja evoluído ao mesmo tempo em que o BD está em funcionamento

# Arquitetura de Três Esquemas

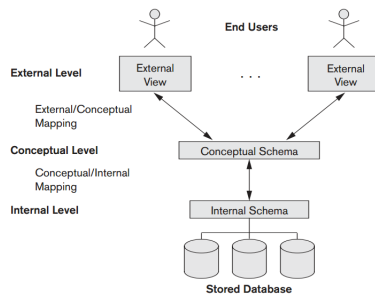
**Objetivo:** separar as aplicações de usuários do banco de dados físico



# Arquitetura de Três Esquemas

Essa arquitetura garante para um BD:

- ▶ Natureza autodescritiva (esquema armazenado em catálogo)
- ▶ Isolamento entre programas e dados, e abstração de dados
- ▶ Suporte a visões múltiplas dos dados



## Referências Bibliográficas

- ▶ *Sistemas de Bancos de Dados* (6ª edição), Elmasri e Navathe. Pearson, 2010.  
Capítulos 1 e 2
- ▶ *Database Systems – the complete book* (2ª edição), Garcia-Molina, Ullman e Widom. Prentice Hall, 2009.  
Capítulo 1
- ▶ *Introdução a Sistemas de Bancos de Dados* (8ª edição), Date. Campus, 2004.  
Capítulos 1 e 2

# Cenas dos próximos capítulos...

## Revisão sobre:

- ▶ Linguagens e arquiteturas dos SGBDs
- ▶ Projeto conceitual de BDs e o modelo entidade-relacionamento