

# Índice

Bases de Dados (2014/09/29 (v97))

Pedro Quaresma

Departamento de Matemática  
Universidade de Coimbra

2014/2015

1. Componente Teórica
  - 1.1 Introdução
  - 1.2 Modelo Entidade-Associação
  - 1.3 Modelo Relacional
  - 1.4 SQL
  - 1.5 Integridade e Segurança
  - 1.6 Dependências funcionais e normalização
2. Componente Prática
  - 2.1 O Sistema de Gestão de Bases de Dados (SGBD) MySQL:
    - O modelo Cliente/Servidor.
    - ferramentas de administração e de consulta não gráficas.
    - ferramentas de administração e de consulta gráficas.
  - 2.2 O Modelo (L|W|M)AMP.
    - A linguagem HTML.
    - A Linguagem PHP.
    - Implementação de uma base de dados e dos seus interfaces de gestão e de utilização através de uma Página da Rede.

2014/09/29 (v97)  
1 / 311

2014/09/29 (v97)  
2 / 311

## Bibliografia

### Bases de Dados

- Database System Concepts (6th edition), Silberschatz, Korth and Sudarshan, McGraw-Hill, 2010.
- Tecnologia de bases de dados (3<sup>a</sup> Edição), Pereira, José Luis Mota, FCA-Ed.Informática, 1998. (68P/PER/3aed).

### MySQL

- Manual de referência do MySQL, <http://dev.mysql.com/doc/>.

### PHP/MySQL/Web

- PHP and MySQL Web Development (Developer's Library) (5th Edition) Luke Welling, Laura Thomson. Addison-Wesley Professional, 2014.
- PHP e MySQL Desenvolvimento Web, Luke Welling e Laura Thomson, Campus, 3<sup>a</sup> Edição, 2005, ISBN-13: 9788535217148.
- PHP 5/MySQL Programming, Andy Harris, Premier Press, 2004.
- Programação com PHP 5.3, Carlos Serrão e Joaquim Marques, FCA, 2009.
- Manual de referência do PHP,  
[http://www.php.net/manual/pt\\_BR/](http://www.php.net/manual/pt_BR/).

Apontamentos da disciplina <http://www.mat.uc.pt/~pedro/lectivos/BasesDados/>

## Introdução

- Sistemas de Gestão de Bases de Dados
- Visão dos dados
- Modelos de dados
- Linguagem de Definição de Dados
- Linguagem de Manipulação de Dados
- Gestão de Transacções
- Gestão de Armazenamento
- Administrador da Base de Dados
- Utilizadores da Base de Dados
- Estrutura Global do Sistema

2014/09/29 (v97)  
3 / 311

2014/09/29 (v97)  
4 / 311

# Bases de Dados!?

Objectivos na utilização de sistemas informáticos:

- ▶ Cálculo - linguagens de manipulação de dados numéricos, linguagens de programação, Fortran, Lisp, C, ....
- ▶ Processamento de informação - linguagens de processamento de informação não numérica, Cobol, ... SQL.

No primeiro caso o objectivos de cálculo mantém-se, as formas de programação têm evoluído.

No segundo caso o objectivo também se mantém, houve no entanto uma clara mudança na forma de o encarar.

## Processamento de Dados I — Ficheiros

Uma primeira aproximação ao problema do processamento de informação (dados) é dada por:

- ▶ Linguagem de programação especializada para o processamento de sequências de caracteres (mais do que no cálculo numérico):  
Cobol, ...
- ▶ Sistema de ficheiros:  
Sistema de ficheiro geridos pelo Sistema Operativo da máquina aonde está a base de dados.

## Sistemas de Ficheiros - Inconvenientes

- ▶ Redundância e inconsistência de dados:
  - ▶ Múltiplos formatos, duplicação de informação em ficheiros diferentes.
- ▶ Dificuldades no acesso aos dados:
  - ▶ Necessidade de escrever um novo programa para efectuar uma nova tarefa.
- ▶ Isolamento de dados — múltiplos ficheiros e formatos.
- ▶ Problemas de integridade:
  - ▶ Restrições de integridade (por exemplo: saldo da conta  $\geq 0$ ) estão incluídas no código dos programas.
  - ▶ Difícil alterar ou adicionar novas restrições.
- ▶ Atomicidade das alterações:
  - ▶ Falhas podem colocar a base de dados num estado inconsistente com alterações parciais já efectuadas. Por exemplo; a transferência de dinheiro de uma conta para outra ou deve ser totalmente realizada ou nenhuma alteração deve ser efectuada.
- ▶ Acessos concorrentes por diversos utilizadores:
  - ▶ Acessos concorrentes necessários por motivos de eficiência
  - ▶ Os acessos concorrentes não controlados podem originar inconsistências.
- ▶ Problemas de segurança.

2014/09/29 (v97)  
5 / 311

## Processamento de Dados II — SGBDs

- ▶ Colecção de dados inter-relacionados (Base de Dados).
- ▶ Conjunto de programas para construir a base de dados (DDL).
- ▶ Conjunto de programas para aceder aos dados (DML).
- ▶ Deve fornecer um ambiente (gráficos e/ou não gráficos) de utilização conveniente e eficiente.
- ▶ Exemplos de aplicações de SGBDs:
  - ▶ **Banca**: todas as transacções e movimentos; **Companhias aéreas**: reservas, horários; **Universidades**: inscrições, notas; **Vendas**: clientes, produtos, compras; **Indústria**: produção, inventário, pedidos, cadeia de fornecimento; **Recursos humanos**: registos dos empregados, salários, impostos; **Sistema de gestão de artigos** numa conferência/revista; **lojas “on-line”**; **Bases de Dados Geográficas**; ....
- ▶ Exemplos de SGBDs:
  - ▶ Modelo Cliente/Servidor: Oracle; MySQL; PostgreSQL; Informix; ...
  - ▶ Ficheiro Único: SQLite; Firebird; ...
  - ▶ Modelo “Consulta por Exemplos”: MS-Access

2014/09/29 (v97)  
7 / 311

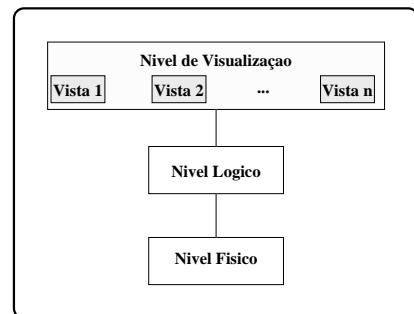
2014/09/29 (v97)  
6 / 311

## SGBDs — Níveis de Abstracção

Nível Físico: descreve como um registo (e.g. cliente) é armazenado

Nível Lógico: descreve os dados armazenados na base de dados, assim como as associações entre os dados

Nível de Visualização: as aplicações ocultam os detalhes dos tipos de dados. Por motivos de segurança alguma da informação pode ser omitida (Por exemplo: o salário de um funcionário).



2014/09/29 (v97)  
9 / 311

## Instâncias e Esquemas

- ▶ Semelhante a variáveis e constantes das linguagens de programação.
- ▶ **Esquema** — a estrutura lógica da base de dados
  - ▶ e.g., a base de dados é constituída por informação sobre clientes, contas e as associações entre si.
  - ▶ Análogo à declaração de uma variável (de um dado tipo)
  - ▶ **Esquema físico**: desenho da base de dados ao nível físico
  - ▶ **Esquema lógico**: desenho da base de dados ao nível lógico
- ▶ **Instância** — o conteúdo de uma base de dados num instante de tempo
  - ▶ Análogo ao valor de uma variável num dado instante
- ▶ **Independência física dos dados** — a capacidade de modificar o esquema físico sem alterar o esquema lógico
  - ▶ As aplicações dependem do esquema lógico
  - ▶ Em geral, as interfaces entre os vários níveis e componentes devem estar bem definidas de modo a que alterações numa parte não influenciem grandemente outras partes.

2014/09/29 (v97)  
10 / 311

## Modelos de Dados

- ▶ Um conjunto de ferramentas para descrever
  - ▶ dados
  - ▶ associações entre dados
  - ▶ semântica dos dados
  - ▶ restrições sobre os dados
- ▶ Modelo Entidade-Associação (Relação) ([ver](#))
- ▶ Modelo Relacional ([ver](#))
- ▶ Outros modelos (passado):
  - ▶ modelo hierárquico ([ver](#))
  - ▶ modelo em rede ([ver](#))
- ▶ Outros modelos (presente/futuro!?):
  - ▶ modelo orientado para objectos ([ver](#))
  - ▶ modelos de dados semi-estruturados ([ver](#))
  - ▶ Modelos baseados em grafos ([ver](#))

([Exemplo usando o modelo E-A](#))

2014/09/29 (v97)  
11 / 311

## Modelo Entidade-Associação

- ▶ O modelo “Entity-Relationship” é baseado na percepção de que o mundo real consiste numa colecção de objectos (“entities”) e de associações entre eles (“relationships”).
  - ▶ Entidades (objectos):
    - ▶ edifícios;
    - ▶ salas;
    - ▶ equipamento.
  - ▶ Associações entre entidades:
    - ▶ pertence, associa equipamento e salas; o equipamento 2920 pertence ao gabinete 6.1.
- ▶ O modelo Entidade-Associação é muito usado na concepção de bases de dados.

[Os diferentes Modelos de Dados](#)

2014/09/29 (v97)  
12 / 311

## Modelo Relacional

- ▶ O modelo relacional usa uma coleção de tabelas para representar tanto os dados como as associações entre eles. Cada tabela tem múltiplas colunas, sendo que cada coluna tem um nome diferente.
- ▶ O modelo relacional é um exemplo de modelo baseado em registos.
- ▶ O modelo relacional é o mais usado na actualidade sendo que a vasta maioria dos actuais sistemas implementa o modelo relacional.

◀ Os diferentes Modelos de Dados

2014/09/29 (v97)  
13/311

## Modelo baseado em Objectos

- ▶ O modelo baseado em objectos pretende extender o modelo E-A com a noção de encapsulamento de dados, métodos (funções) e identidade de objectos.
- ▶ O modelo relacional baseado em objectos combina as características do modelo de dados orientado aos objectos com o modelo de dados relacional.

◀ Os diferentes Modelos de Dados

2014/09/29 (v97)  
14/311

## Modelo de dados Semi-Estruturados

- ▶ O modelo de dados semi-estruturados permite a especificação de dados onde itens individuais de informação do mesmo tipo podem ter diferentes conjuntos de atributos (“variant records”).
- ▶ A linguagem XML (eXtensible Markup Language) é usualmente usada quando se pretende representar dados semi-estruturados.

◀ Os diferentes Modelos de Dados

2014/09/29 (v97)  
15/311

## Modelo Hierárquico

- ▶ No modelo hierárquico, a exemplo do modelo em rede, os dados são organizados em registos e as associações representadas por ponteiros.
- ▶ Ao contrário do modelo em rede os registos organizam-se segundo uma estrutura em árvore, em vez de um grafo generalizado.

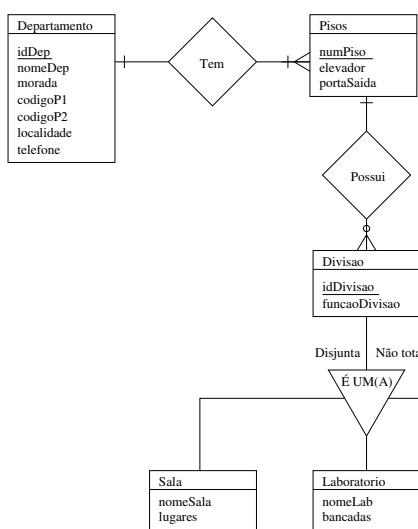
◀ Os diferentes Modelos de Dados

2014/09/29 (v97)  
16/311

## Modelo em Rede (grafo generalizado)

- ▶ O modelo em rede difere do modelo relacional no facto de que os dados são representados como uma colecção de registo e de ponteiros entre eles.
- ▶ Um registo, neste modelo, é similar a uma entidade do modelo E-A, um ponteiro é uma associação entre dois registo. Isto é um ponteiro pode ser visto como uma forma (binária) restrita de associação no sentido do modelo E-A.

[◀ Os diferentes Modelos de Dados](#)



2014/09/29 (v97)  
17/311

## Modelo Baseado na Teoria dos Grafos

- ▶ As bases de dados orientadas a grafos (BDOG) representam a informação como nós de um grafo e as suas relações como arcos, de forma que se possa usar a teoria dos grafos para precorrer a base de dados.
- ▶ A informação é guardada como um grafo: os nós representam as entidades e os arcos representam as relações.

(e.g Neo4j)

[◀ Os diferentes Modelos de Dados](#)

2014/09/29 (v97)  
18/311

## Modelo E-A

Exemplo de um Diagrama Entidade-Associação (DEA)

## Linguagem de Definição de Dados (DDL)

Especificação da notação para definição do esquema da base de dados. Por exemplo:

```

CREATE TABLE Equipamento (
    IdEquip  integer,
    NomeEquip  char(10),
    Descricao  text)
    
```

O compilador da DDL gera um conjunto de tabelas armazenadas num dicionário de dados.

O dicionário de dados contém meta-dados (dados sobre os dados):

- ▶ Esquema de bases de dados;
- ▶ Linguagem de definição de dados e armazenamento:
  - ▶ Linguagem onde se especificam as estruturas de armazenamento e métodos de acesso utilizados pela base de dados;
  - ▶ Normalmente uma extensão da linguagem da definição de dados.

2014/09/29 (v97)  
20/311

## Linguagem de Manipulação de Dados (DML)

Linguagem para aceder e manipular os dados organizados de acordo com o respectivo modelo de dados.

A DML também é conhecida por linguagem de consulta

Duas classes de linguagens:

Procedimental o utilizador especifica quais os dados que se pretendem assim como obter os dados;

Declarativa o utilizador especifica quais os dados pretendidos sem especificar a maneira como os obter

A SQL (declarativa, não procedural) é a linguagem de consulta mais utilizada.

## SQL

Linguagem não-procedimental de uso generalizado. Por exemplo: encontrar o nome do equipamento com identificação (n. de inventário) "2920"

```
SELECT NomeEquip  
      FROM Equipamento  
 WHERE IdEquip = 2920
```

Outro exemplo: procurar as divisões do 3º piso.

```
SELECT IdDivisao  
      FROM Divisao , Piso  
 WHERE NumPiso = 3
```

As aplicações geralmente acedem a bases de dados por intermédio de:

- ▶ Extensões às linguagens permitindo SQL embutido;
- ▶ Interface de aplicações (e.g. ODBC/JDBC) permitindo o envio de consultas SQL para a base de dados

## Utilizadores da Base de Dados

Os utilizadores diferenciam-se pela forma esperada de interacção com o sistema:

Programadores de aplicações interagem com o sistema através de chamadas DML.

Utilizadores sofisticados constroem pedidos numa linguagem de consulta a bases de dados.

Utilizadores especializados escrevem aplicações de bases de dados especializadas que não se enquadram com o espírito do processamento de dados tradicional.

Utilizadores chamam uma das aplicações que foi construída previamente.

Por exemplo, pessoas a acederem a uma base de dados através da Rede, caixas, pessoal de secretariado.

2014/09/29 (v97)  
21 / 311

## Administrador da Base de Dados

Coordena todas as actividades do sistema de base de dados;  
As funções do administrador de bases de dados incluem:

- ▶ Definição do esquema;
- ▶ Definição dos métodos de acesso e estrutura de armazenamento;
- ▶ Modificação do esquema e da organização física;
- ▶ **Dar aos utilizadores autorizações de acesso à base de dados;**
- ▶ Especificar restrições de integridade;
- ▶ Servir de ligação entre os utilizadores;
- ▶ Monitorar a performance e responder a alterações nos requisitos.

2014/09/29 (v97)  
23 / 311

2014/09/29 (v97)  
22 / 311

## Gestão de Transacções

Uma transacção é um conjunto de operações que efectuam uma função lógica na aplicação de base de dados

A componente de gestão de transacções garante que a base de dados se mantém num estado consistente (correcto) apesar de falhas no sistema (por exemplo: falta de energia eléctrica e paragens abruptas do sistema operativo) e de transacções falhadas.

O gestor de controlo de concorrência coordena a interacção entre transacções concorrentes para garantir a consistência da base de dados.

## Gestão do Armazenamento

O gestor do armazenamento é um módulo de programa que fornece uma interface entre os dados de baixo nível armazenados na base de dados e as aplicações e consultas submetidas ao sistema.

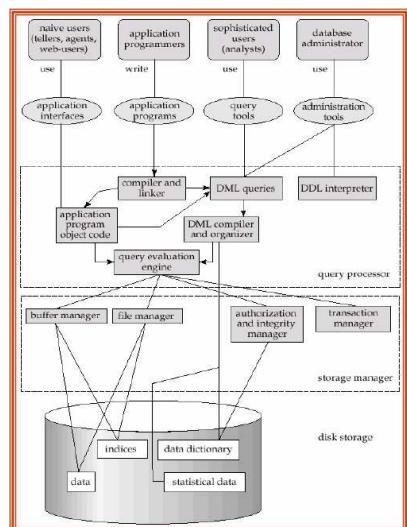
O gestor de armazenamento é responsável pelas seguintes tarefas:

- ▶ interacção com o gestor de ficheiros;
- ▶ armazenamento, recuperação e alteração eficientes dos dados.

2014/09/29 (v97)  
25 / 311

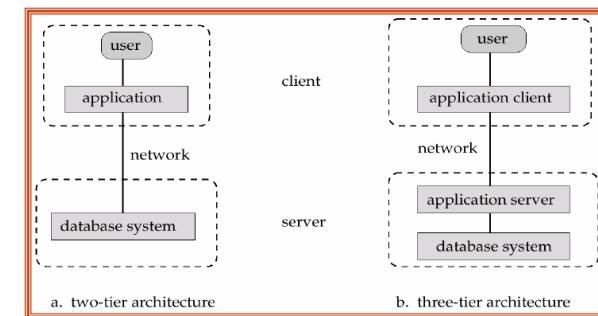
2014/09/29 (v97)  
26 / 311

## Estrutura Genérica do Sistema



2014/09/29 (v97)  
27 / 311

## Arquitecturas de Aplicação



Arquitectura de duas camadas: os programas clientes recorrem a, por exemplo, ODBC/JDBC para comunicar com a base de dados.

Arquitectura de três camadas: por exemplo, aplicações da Rede e aplicações construídas recorrendo a “software intermediário”.

2014/09/29 (v97)  
28 / 311