



Programação e Sistemas de Informação

CURSO PROFISSIONAL TÉCNICO DE
GESTÃO E PROGRAMAÇÃO DE SISTEMAS INFORMÁTICOS

Introdução aos Sistemas de Informação

MÓDULO 12

Professor: João Martiniano

Introdução às bases de dados

- Base de dados: uma base de dados consiste num conjunto de dados, com uma determinada estrutura
- Dados: a informação que se pretende armazenar, manipular ou consultar
- As bases de dados (não informáticas) já existem há muito
- Exemplos:
 - lista telefónica
 - registos em papel de clientes de uma empresa
 - etc.
- As sociedades dos países desenvolvidas dependem cada vez mais de bases de dados informáticas
- As bases de dados tornaram-se essenciais para o normal funcionamento das sociedades
- Todos os dias, são gerados e utilizados pelos cidadãos, organizações públicas e privadas, uma crescente quantidade de dados

Introdução às bases de dados

- Exemplos de bases de dados:
 - Escolas
 - classificações dos alunos
 - sumários
 - marcação de refeições
 - etc.
 - Bancos:
 - levantamento/depósito de dinheiro (multibanco)
 - pagamentos
 - dados dos clientes/dados das contas
 - etc.
 - Bibliotecas
 - dados dos livros
 - requisições de livros
 - dados pessoais dos leitores
 - etc.

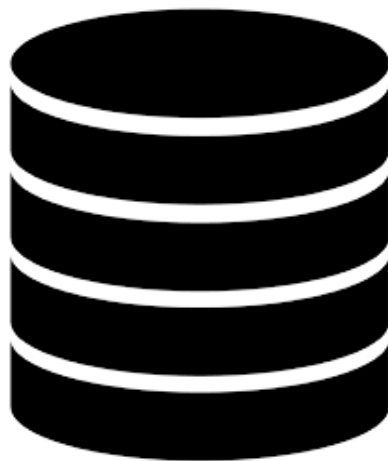
Introdução às bases de dados

Exercício 1

- Identifique três situações concretas em que, na nossa sociedade, são utilizadas bases de dados informáticas

Introdução às bases de dados

- Ícone normalmente utilizado para representar bases de dados informáticas:



Sistema de Gestão de Bases de Dados (SGBD)

- Um SGBD é um software especializado que trabalha com bases de dados
- Objetivos:

- criar e manipular bases de dados
- gerir bases de dados
- obter dados
- interagir com utilizadores e outras aplicações
- etc.

- Exemplos de SGBD:

- Microsoft Access
- SQL Server
- MySQL / MariaDB
- Oracle
- etc.



- Normalmente as bases de dados criadas com um tipo de SGBD não podem ser utilizadas por outro tipo de SGBD

A arquitetura cliente-servidor nas bases de dados

- Os SGBD funcionam em arquitetura cliente-servidor (*client-server*):

Servidor

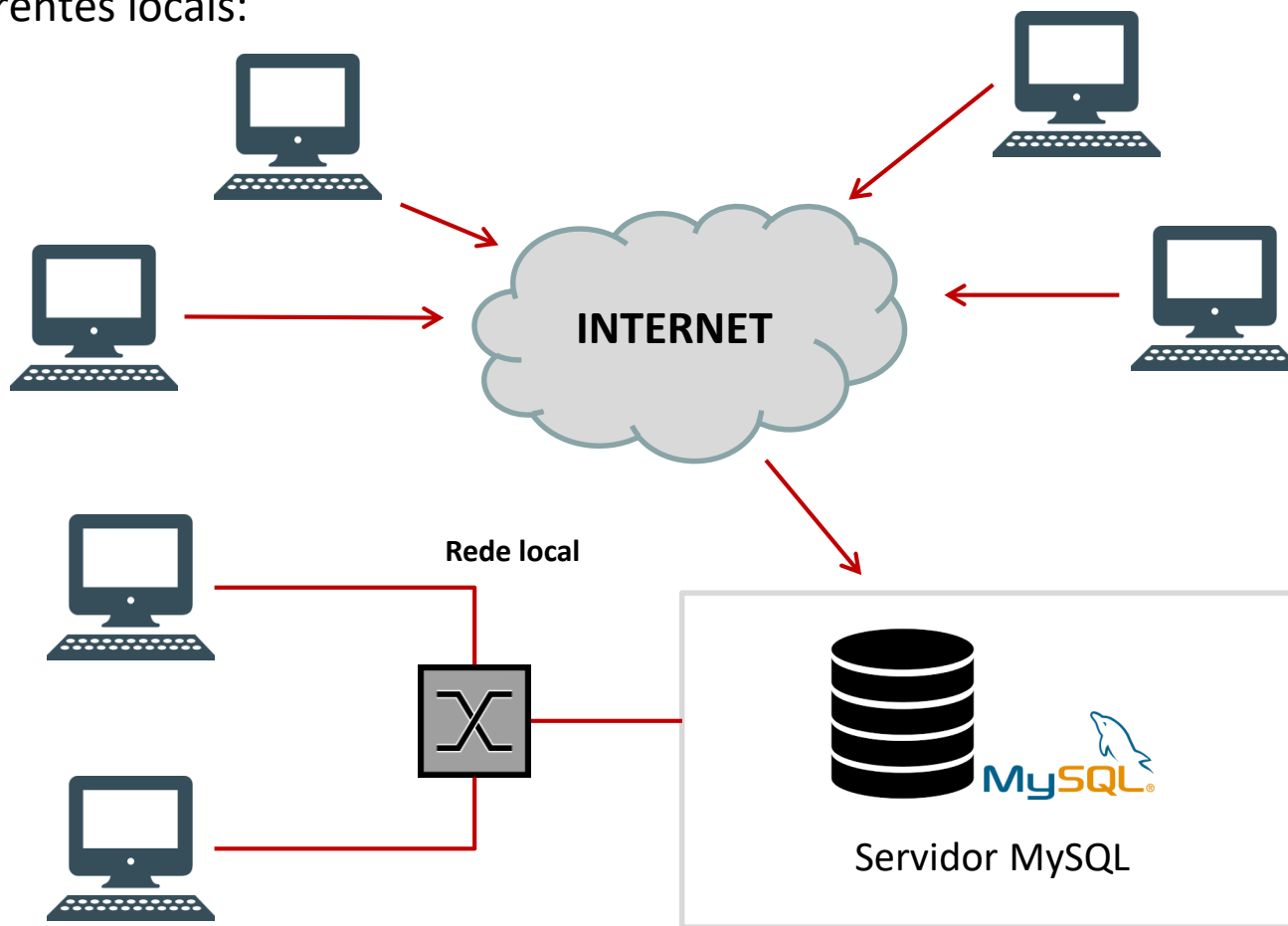
- Em funcionamento contínuo
- Armazena os dados
- Executa operações solicitadas pelo cliente
- Responde a pedidos por parte do cliente, enviando-lhe os dados solicitados
- Um servidor pode servir múltiplos clientes
- Pode ou não estar no mesmo sistema (computador) do(s) cliente(s)

Cliente

- Pode-se entender como cliente:
 - um computador (ou dispositivo equivalente, como tablet)
 - um software para trabalhar com servidores de bases de dados
- O software cliente, permite interagir com o servidor:
 - pedir dados, ver dados, solicitar a execução de operações, etc.

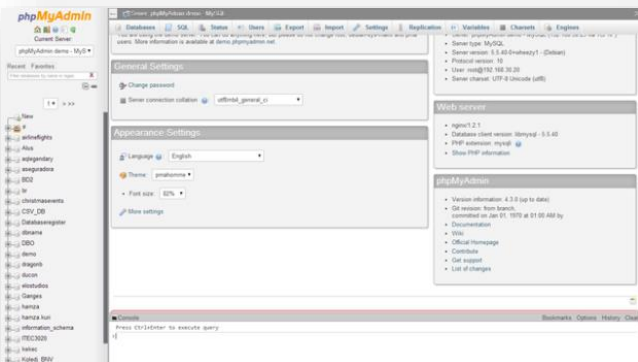
A arquitetura cliente-servidor nas bases de dados

- Por exemplo, um servidor MySQL pode ser acedida por vários clientes, a partir de diferentes locais:

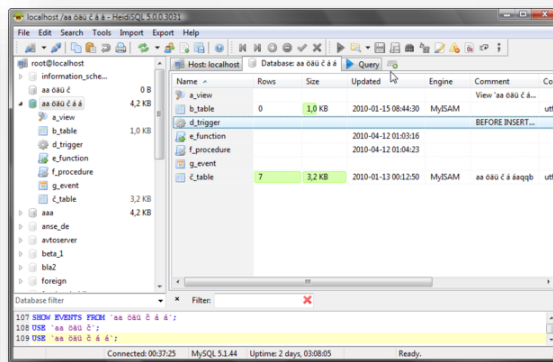


A arquitetura cliente-servidor nas bases de dados

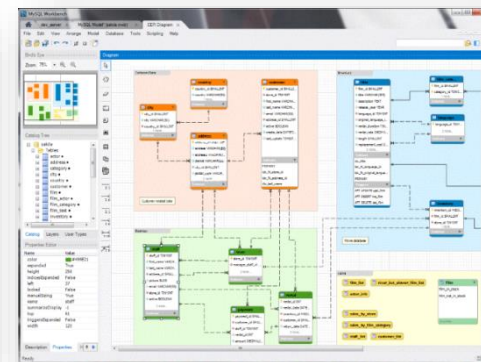
- Para além disso, o mesmo servidor MySQL pode ser acedida por diferentes tipos de clientes:



phpMyAdmin



HeidiSQL



MySQL Workbench



Modelos de bases de dados

- Existem vários modelos de bases de dados
- Têm evoluído ao longo do tempo
- Alguns exemplos:
 - *flat file* (modelo de base de dados plana)
 - modelo hierárquico
 - modelo rede
 - modelo relacional
 - modelo orientado a objetos
- Os modelos de bases de dados são uma forma de:
 - capturar os modelos naturais (uma situação do mundo real, com dados)
 - e transpô-los para os sistemas informáticos

Flat file (modelo de base de dados plana)

- Neste modelo, a base de dados reside num ficheiro
- O ficheiro contém registos de dados
- Os registos estão organizados de forma uniforme
- É um tipo de organização de dados simples e "plana" (daí o termo *flat file*)
- Não há forma de estabelecer, explicitamente, uma relação entre a informação
- Normalmente os registos estão organizados por linhas (1 registo por linha)
- Existem diferentes formas de colocar os dados
- Os dados dos campos podem estar separados:
 - por vírgulas, espaços ou outro carater
 - ou ter um comprimento fixo
- Os dados podem também estar delimitados por um determinado carater

***Flat file* (modelo de base de dados plana)**

- Exemplo de um *flat file* contendo uma base de dados de automóveis:

Base de dados 'Automóveis'

Marca,Modelo,Matrícula,Ano

Opel,Corsa,10-AA-09,1997

Tesla,Model X,31-01-SW,2018

Opel,Ampera,15-BA-11,2012

Renault,Twingo,89-11-HZ,2000

Renault,Clio,01-AA-01,2009

Flat file (modelo de base de dados plana)

- Exemplo de um *flat file* contendo uma base de dados de automóveis:

Base de dados 'Automóveis'

Designação
dos campos



Marca,Modelo,Matrícula,Ano

Opel,Corsa,10-AA-09,1997

Tesla,Model X,31-01-SW,2018

Opel,Ampera,15-BA-11,2012

Renault,Twingo,89-11-HZ,2000

Renault,Clio,01-AA-01,2009

Flat file (modelo de base de dados plana)

- Exemplo de um *flat file* contendo uma base de dados de automóveis:

Base de dados 'Automóveis'

| | Marca,Modelo,Matrícula,Ano |
|-------------|------------------------------|
| Registo 1 → | Opel,Corsa,10-AA-09,1997 |
| Registo 2 → | Tesla,Model X,31-01-SW,2018 |
| Registo 3 → | Opel,Ampera,15-BA-11,2012 |
| Registo 4 → | Renault,Twingo,89-11-HZ,2000 |
| Registo 5 → | Renault,Clio,01-AA-01,2009 |

Flat file (modelo de base de dados plana)

- Exemplo de um *flat file* contendo uma base de dados de automóveis:

Base de dados 'Automóveis'

| Marca | Modelo | Matrícula | Ano |
|---------|---------|-----------|------|
| Opel | Corsa | 10-AA-09 | 1997 |
| Tesla | Model X | 31-01-SW | 2018 |
| Opel | Ampera | 15-BA-11 | 2012 |
| Renault | Twingo | 89-11-HZ | 2000 |
| Renault | Clio | 01-AA-01 | 2009 |

↑
Campo 'Modelo'

***Flat file* (modelo de base de dados plana)**

- Exemplo de um *flat file* contendo uma base de dados de automóveis:

Base de dados 'Automóveis'

```
Marca,Modelo,Matrícula,Ano  
Opel,Corsa,10-AA-09,1997  
Tesla,Model X,31-01-SW,2018  
Opel,Ampera,15-BA-11,2012  
Renault,Twingo,89-11-HZ,2000  
Renault,Clio,01-AA-01,2009
```

Separador

Flat file (modelo de base de dados plana)

- Um exemplo de uma base de dados deste tipo é o ficheiro `/etc/passwd` nos sistemas Unix
- Contém os dados dos utilizadores com conta no sistema
- Cada linha corresponde aos dados de um utilizador
- Formato dos registos:

```
username:password:UID:GID:full name:home directory:login shell
```

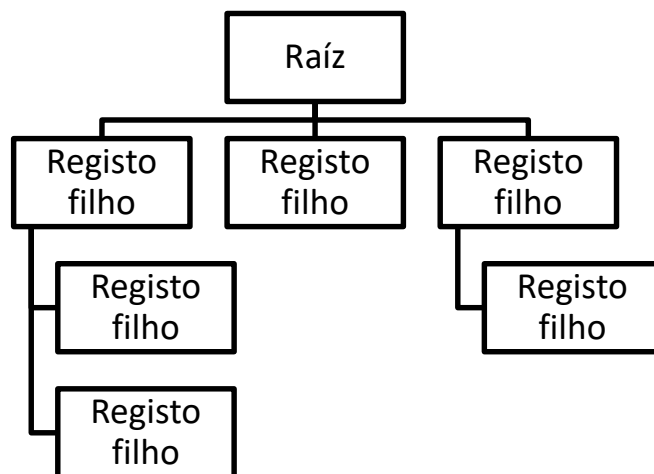
- Exemplo:

```
kirk:XvsQk556tle1:102:100:James T. Kirk:/home/kirk:/bin/bash
```

- **Pergunta: qual é o separador utilizado nesta base de dados?**

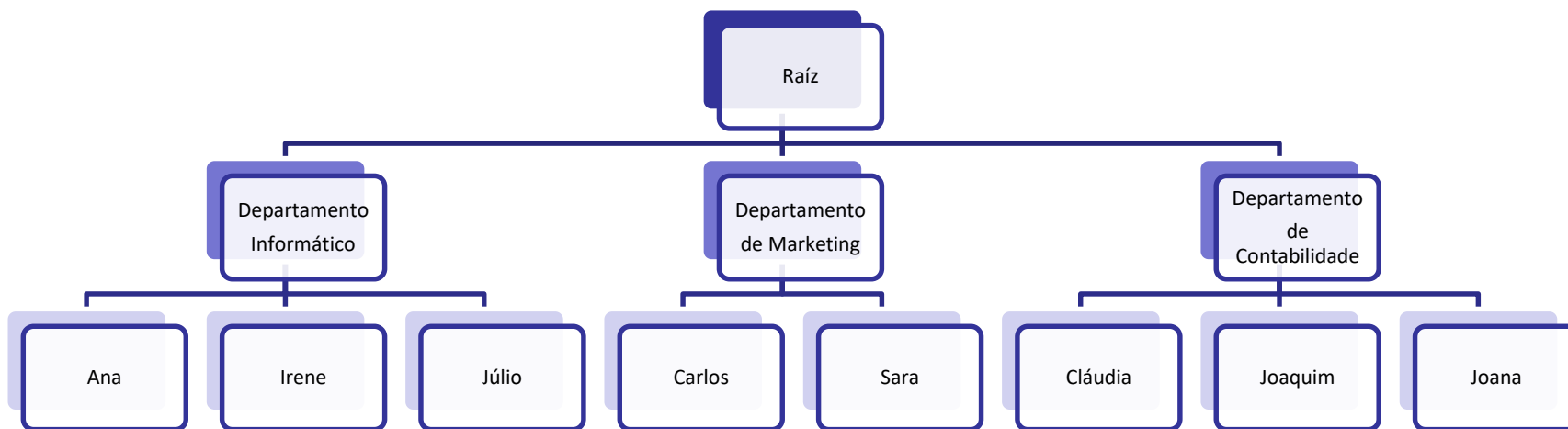
Modelo hierárquico (*hierarchical model*)

- Os dados encontram-se organizados hierarquicamente, em registos, interligados por meio de ligações
- Cada registo contém campos
- Cada campo contém apenas um valor
- Um registo apenas pode ter um registo antecessor (registo "pai") mas pode ter vários registos sucessores ("filhos")
- Ao primeiro registo da hierarquia dá-se o nome de raiz (ou *root*)
- Desvantagem: os dados estão organizados



Modelo hierárquico (*hierarchical model*)

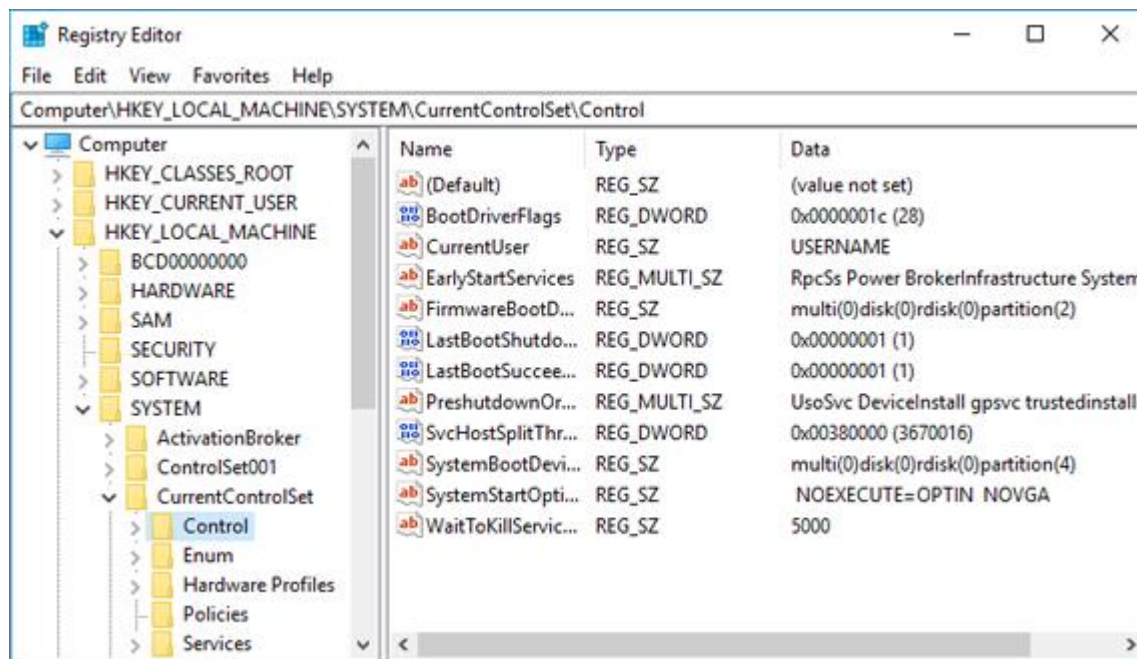
- Exemplo: departamentos de uma empresa e respetivos colaboradores



- Neste modelo, cada colaborador apenas pode trabalhar num único departamento

Modelo hierárquico (*hierarchical model*)

- O **Windows Registry** é considerada uma base de dados hierárquica (contém várias definições do sistema e aplicativos instalados):



Modelo hierárquico (*hierarchical model*)

- O **Windows Registry** é considerada uma base de dados hierárquica (contém várias definições do sistema e aplicativos instalados):

Registos →

| Name | Type | Data |
|---------------------|--------------|---|
| (Default) | REG_SZ | (value not set) |
| BootDriverFlags | REG_DWORD | 0x0000001c (28) |
| CurrentUser | REG_SZ | USERNAME |
| EarlyStartServices | REG_MULTI_SZ | RpcSs Power BrokerInfrastructure System |
| FirmwareBootD... | REG_SZ | multi(0)disk(0)rdisk(0)partition(2) |
| LastBootShutdo... | REG_DWORD | 0x00000001 (1) |
| LastBootSuccee... | REG_DWORD | 0x00000001 (1) |
| PreshutdownOr... | REG_MULTI_SZ | UsoSvc DeviceInstall gpSvc trustedinstall |
| SvcHostSplitThr... | REG_DWORD | 0x00380000 (3670016) |
| SystemBootDevi... | REG_SZ | multi(0)disk(0)rdisk(0)partition(4) |
| SystemStartOpti... | REG_SZ | NOEXECUTE=OPTIN NOVGA |
| WaitToKillServic... | REG_SZ | 5000 |

Campos do registo **Control**

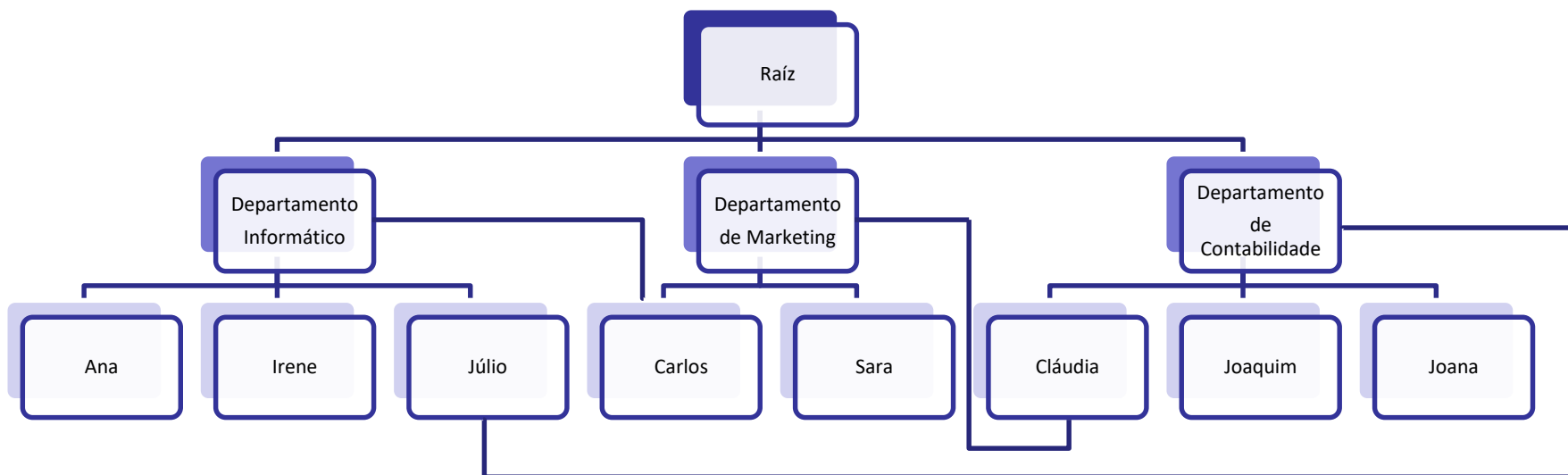
Valores dos campos

Modelo rede (*network model*)

- Este modelo surgiu como uma evolução do modelo hierárquico
- Neste modelo, os dados encontram-se organizados de forma mais flexível
- Cada registo pode ter vários registos "pai" e vários registos "filhos"
- Estão ligados numa estrutura de tipo rede
- É um modelo mais flexível pois adequa-se mais naturalmente à forma como as entidades se relacionam entre si num sistema
- Embora sendo mais flexível que o modelo hierárquico, também apresenta alguns problemas, tendo sido abandonado em favor do modelo relacional

Modelo rede (*network model*)

- Exemplo: os colaboradores podem trabalhar em vários departamentos



Modelo relacional (*relational model*)

- Os dados são armazenados em tabelas
- Podem existir relacionamentos entre tabelas
- Flexível e fácil de utilizar
- Atualmente é o modelo mais utilizado
- Exemplo:

Tabela Colaboradores

| <u>ID</u> | Nome | <u>Departamento_ID</u> |
|-----------|----------------|------------------------|
| 1 | Ana Carolina | 1 |
| 2 | Carlos Marques | 1 |
| 3 | Marta Santos | 2 |
| 4 | Tiago Alves | 3 |

Tabela Departamentos

| <u>ID</u> | Designacao |
|-----------|---------------|
| 1 | Contabilidade |
| 2 | Marketing |
| 3 | Informática |