

Introdução

Banco de Dados
Prof. Paulo Perris

Introdução

- **O que é um Banco de Dados (BD)?**
 - É uma coleção de dados relacionados e armazenados em algum dispositivo.
- **Propriedades de um BD:**
 - Um BD é uma coleção de dados com algum significado inerente.
 - Um BD é projetado, construído e “povoado” com dados para um específico propósito.
 - Um BD representa um mini-mundo.

Introdução

- O que é um SGBD (Sistema de Gestão de Banco de Dados)?

Introdução

- O que é um SGBD (Sistema de Gestão de Banco de Dados)?
 - É um software que permite construir e manipular um BD.

Introdução

- O que é um SGBD (Sistema de Gestão de Banco de Dados)?
 - É um software que permite construir e manipular um BD.
 - Ex.:
 - Oracle
 - SQL Server
 - DB2
 - Postgresql
 - MySQL
 - Sybase

Introdução

- Por que usar um SGBD?

Introdução

- Por que usar um SGBD?
 - **controlar redundância**

Introdução

- Por que usar um SGBD?
 - controlar redundância
 - **compartilhamento de dados**

Introdução

- Por que usar um SGBD?
 - controlar redundância
 - compartilhamento de dados
 - **independência de dados**

Introdução

- Por que usar um SGBD?
 - controlar redundância
 - compartilhamento de dados
 - independência de dados
 - **segurança**

Introdução

- Por que usar um SGBD?
 - controlar redundância
 - compartilhamento de dados
 - independência de dados
 - segurança
 - backup e recuperação à falhas

Introdução

- Por que usar um SGBD?
 - controlar redundância
 - compartilhamento de dados
 - independência de dados
 - segurança
 - backup e recuperação à falhas
 - **forçar restrições de integridade**

Introdução

- Por que usar um SGBD?
 - controlar redundância
 - compartilhamento de dados
 - independência de dados
 - segurança
 - backup e recuperação à falhas
 - forçar restrições de integridade
 - **aumentar a produtividade e disponibilidade**

Introdução

- Por que usar um SGBD?
 - controlar redundância
 - compartilhamento de dados
 - independência de dados
 - segurança
 - backup e recuperação à falhas
 - forçar restrições de integridade
 - aumentar a produtividade e disponibilidade
 - **flexibilidade, padronização.**

Introdução – Comparação

SGBD

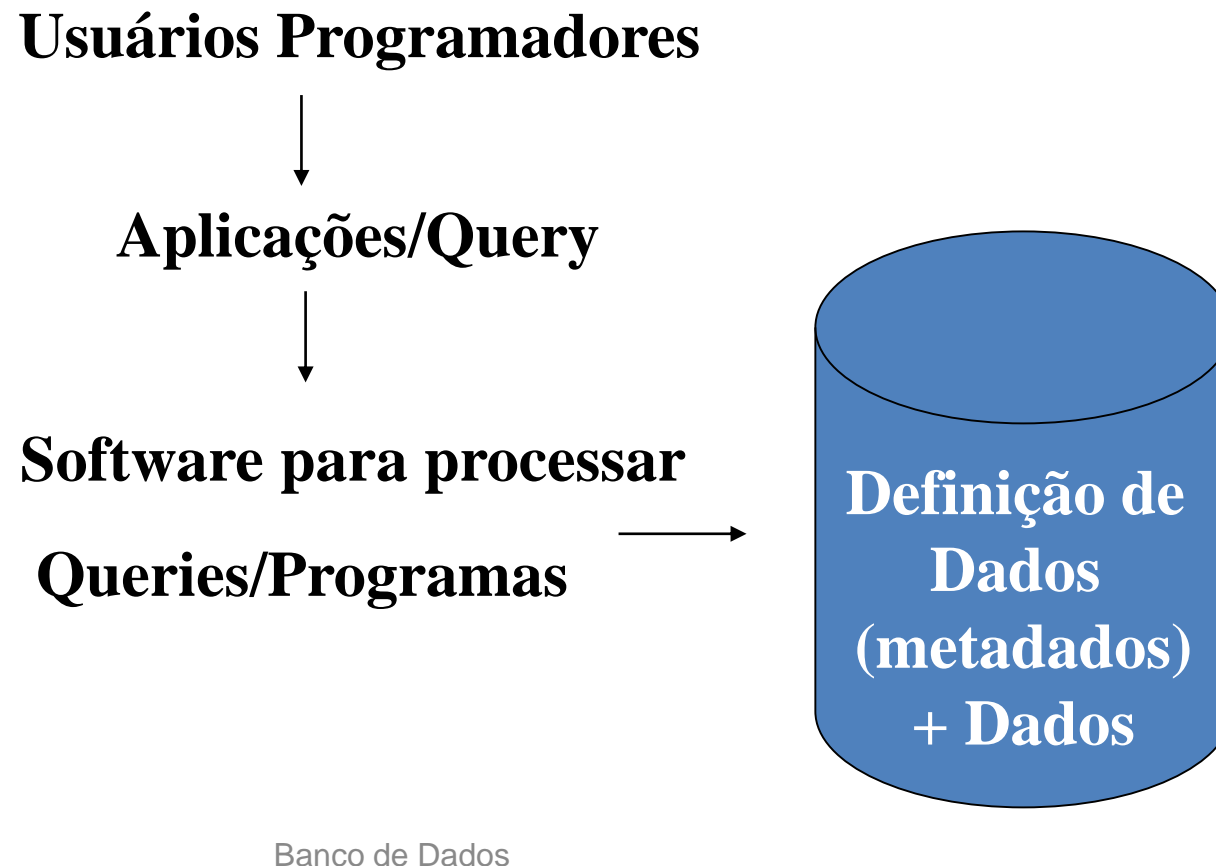
x

Sistema de Arquivos

- | | |
|---|--|
| <ul style="list-style-type: none">• Armazena dados e metadados• redundância de dados é controlada• uso de linguagem de alto nível, não procedural (declarative: o quê?)• independência dados-programa• múltiplas visões dos dados• concorrência, tolerância a falhas, integridade, segurança | <ul style="list-style-type: none">• Definição é parte integrante da aplicação• há bastante redundância de dados• linguagem procedural (imperativa: como?)• dependência dados-programa• visão única dos dados |
|---|--|

Introdução

- Ex. Uma arquitetura simplificada de um SGBD:



Introdução

- **Arquitetura de um SGBD:**
 - Interface de alto nível de abstração que provê: consultas, manipulação de dados, definição de dados, geradores de relatórios, Ling. 4 geração;
 - Tradutor/otimizador de consultas;
 - Visões do usuário de BD;
 - Controle de concorrência (sincronização de acessos simultâneos ao BD);

Introdução

- **Arquitetura de um SGBD (cont.):**
 - **Controle de integridade (validação de restrições de integridade);**
 - **Controle de segurança (autorização de acesso aos dados);**
 - **Controle de recuperação (torna o SGBD tolerante à falhas);**
 - **Eficiente sistema de arquivos com técnicas indexação eficientes que permitem armazenar e manipular os dados do BD.**

Introdução

- **Modelos e Esquemas de Dados**

- Um modelo de dados é um conjunto de conceitos para descrever um BD. Os conceitos são classificados como: estruturas de dados, regras de integridade de dados (RI), operações sobre os dados. Os modelos são eles próprios classificáveis, como veremos no item seguinte.
- Um esquema de dados é a descrição de um BD, segundo um modelo de dados.

- **Instância de um BD**

- estado do esquema de um BD, num dado instante.

Introdução

- **Modelos e Esquemas conceituais**
 - Um modelo conceitual é aquele que não é utilizado pelos SGBDs, mas é importante na chamada fase conceitual do projeto de um BD. Exemplo de modelo conceitual: **MER**, e suas variações.
 - Um esquema conceitual é a descrição conceitual de um BD específico, segundo um modelo conceitual. Exemplo: Esquema ER.

Introdução

- **Modelos Lógicos**

- são os modelos de dados utilizados pelos sistemas de gerência de banco de dados (SGBD). São lógicos porque sua implementação não precisa ser conhecida. Exemplos: **Modelo Relacional**, Modelos Orientados a Objeto Jasmine, GemStone, etc.

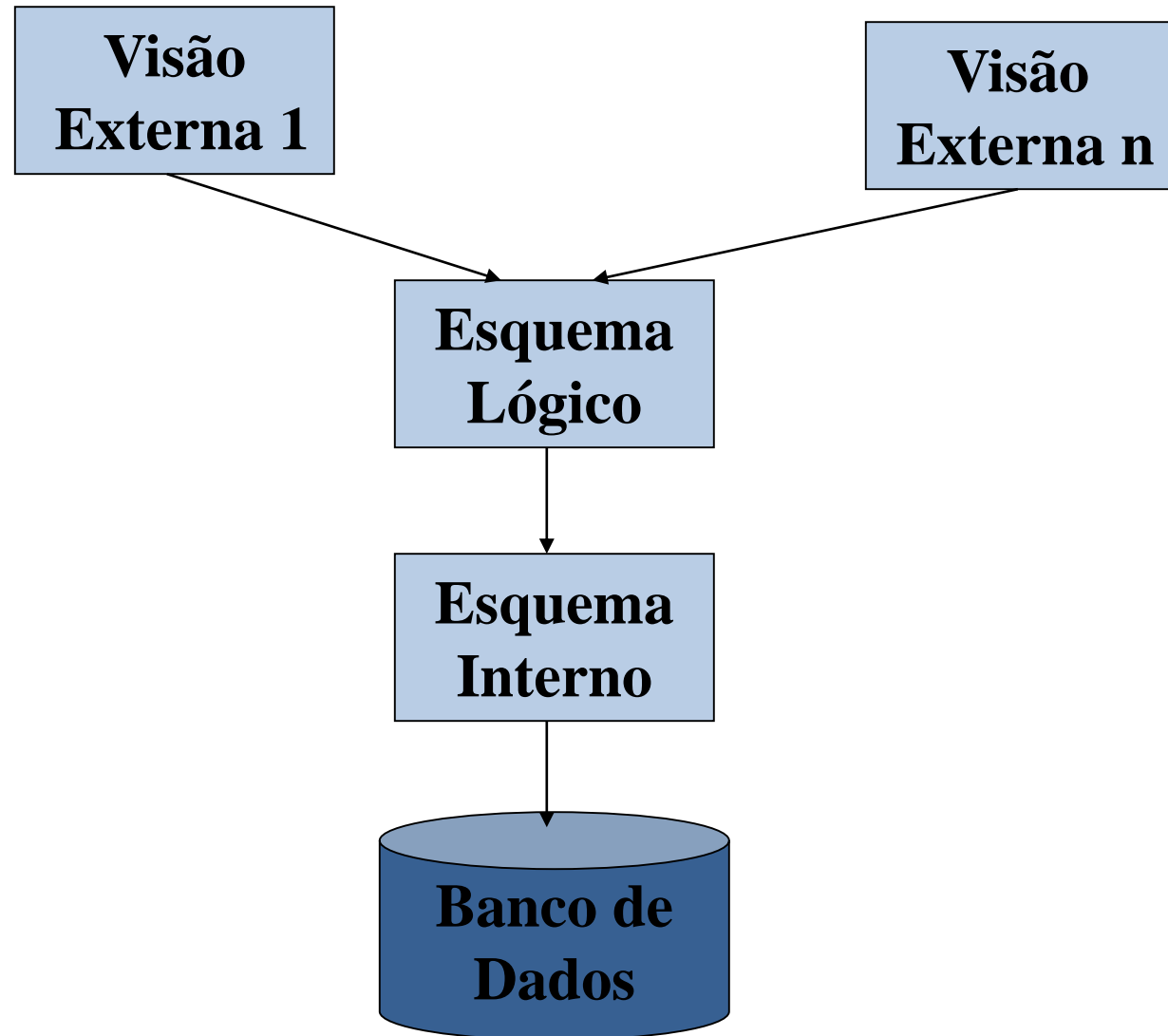
- **Esquema Lógico**

- É a descrição das estruturas e das operações de um BD específico, utilizando um modelo de dados.
- Exemplos: Esquema Relacional, Esquema OO Jasmine.

Introdução

- **Modelos e Esquemas Internos**
 - Um modelo interno (também chamado de modelo físico) é aquele que é usado para implementar um modelo lógico. Cada SGBD tem o seu modelo interno.
 - Um esquema interno ou físico é a descrição interna de um BD, segundo o modelo interno ou físico.

Arquitetura ANSI/SPARC



Banco de Dados

Introdução

- Usuários de um SGBD:
 - Administrador de BD (ABD)
 - Analista de Dados
 - Analista de Sistemas
 - Programador de Aplicações
 - Usuários finais

Introdução

- Linguagens
 - Linguagem de Definição de Dados
 - Linguagem de Manipulação de Dados
 - **Linguagem de Armazenamento**
 - Linguagem de Definição de Visão
- ✓ A linguagem de banco de dados relacional SQL representa uma combinação de DDL, VDL e DML, bem como as instruções para especificação de restrição, evolução de esquema e outros recursos.

Certificação em SGBDs

- Certificação profissional está em evidência recentemente em vários trabalhos de TI
- Em BD existem vários programas de certificação:
 - Oracle: www.oracle.com/education/certification/
 - Microsoft SQL Server:
www.microsoft.com/trainingandservices/default.asp?PageID=training
 - IBM DB2: www.ibm.com/certify
 - Sybase: www.sybase.com/education/profcert/
 - Informix:
www.informix.com/informix/training/courses/certific/welcome.htm

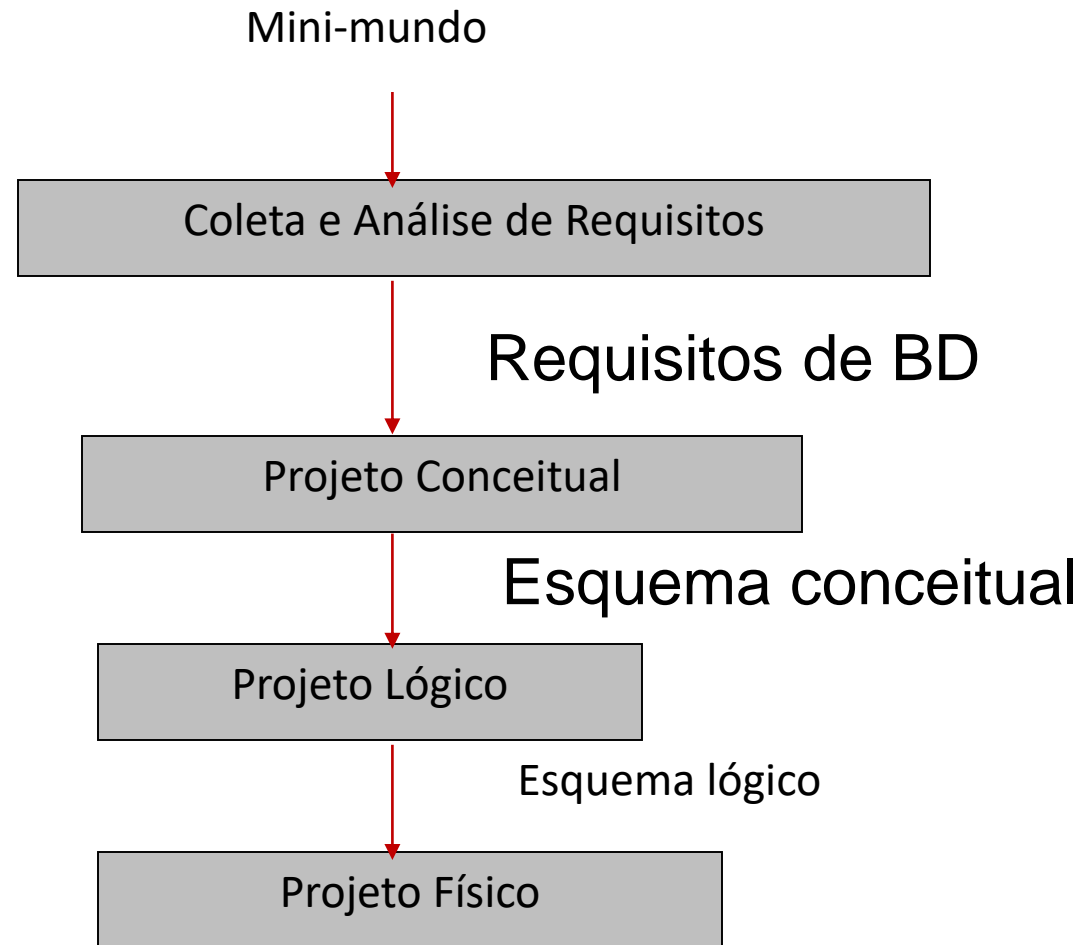
Modelo Entidade Relacionamento - MER

Banco de Dados
Prof. Paulo Perris

Modelo Entidade Relacionamento

- **Modelagem semântica permite aproximar o modelo obtido do mundo real.**
- **O cliente sabe que quer o banco de dados, mas ele não sabe o que o banco de dados deve conter.**
- **Um modelo é um componente chave em uma forma eficiente de desenvolver um banco de dados que realmente funciona.**

Fases de um projeto de BD



Entidades

Uma entidade é tudo aquilo sobre o qual se deseja manter dados.

- Podendo representar:
 - objetos concretos: pessoas, livros, carros, ...
 - conceitos abstratos: empresas, eventos, embarques, ...

Entidades

- É um subconjunto de objetos (instâncias) que:
 - desempenha o mesmo papel semântico
 - possui os mesmos tipos de propriedades (atributos)

Ex.:

- Conjunto de todas as contas correntes de um banco
- Conjunto de todos os empregados de uma empresa
- Conjunto de todos os filmes de um produtor

Empregado

Aluno

Empréstimo

Entidades

- Entidades devem ser descritas num Dicionário de Dados

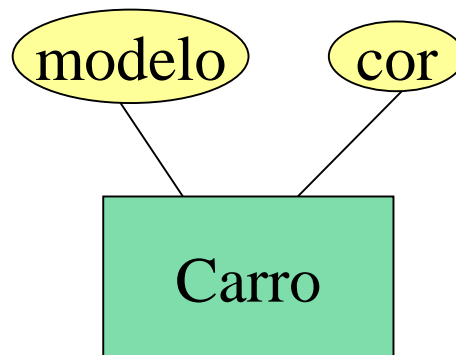
Entidade: EMPREGADO

Descrição: Pessoa que mantém vínculo empregatício com a Empresa através de um contrato de trabalho de acordo com a legislação trabalhista

Atributos

São as propriedades que caracterizam ou descrevem uma entidade ou um relacionamento.

- Ex.: A entidade CARRO poderia ter os seguintes atributos:
 - ❖ Placa, fabricante, modelo, ano de fabricação, cor, preço



Atributos

Atributos devem também ser descritos no Dicionário de Dados:

Entidade: EMPREGADO

Atributo: Data de Admissão

Descrição: data na qual foi assinado o contrato de trabalho entre a empresa e o empregado

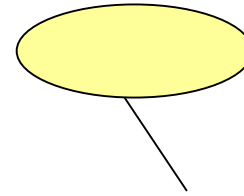
Domínio: data posterior a 03/01/78 (data de criação da empresa) e a data de nascimento do empregado

Atributos

- ***Simples***: é atômico.

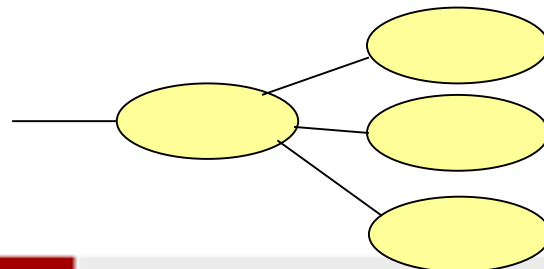
Ex. Idade: numérico

Nome: cadeia de caracteres



- ***Composto***: contém sub-atributos que compõem o atributo.

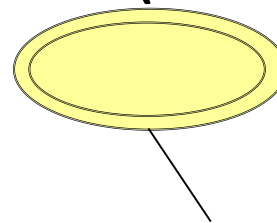
Ex. Endereço(rua, número, bairro, CEP, cidade)



Atributos

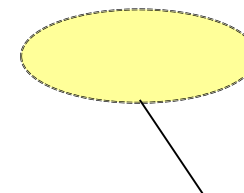
- **Multivalorados:** possuem vários valores numa instância de uma entidade.

Ex.: PESSOA:TitulaçãoSuperior(nenhum, Bel. MSc., PhD)



- **Atributos derivados:** podem ser determinados a partir de outros atributos/entidades.

Ex.: Idade e dataAniversário



Instância de Entidade

Instância:

Objeto de uma entidade com suas respectivas propriedades que é distinguível dos outros objetos.

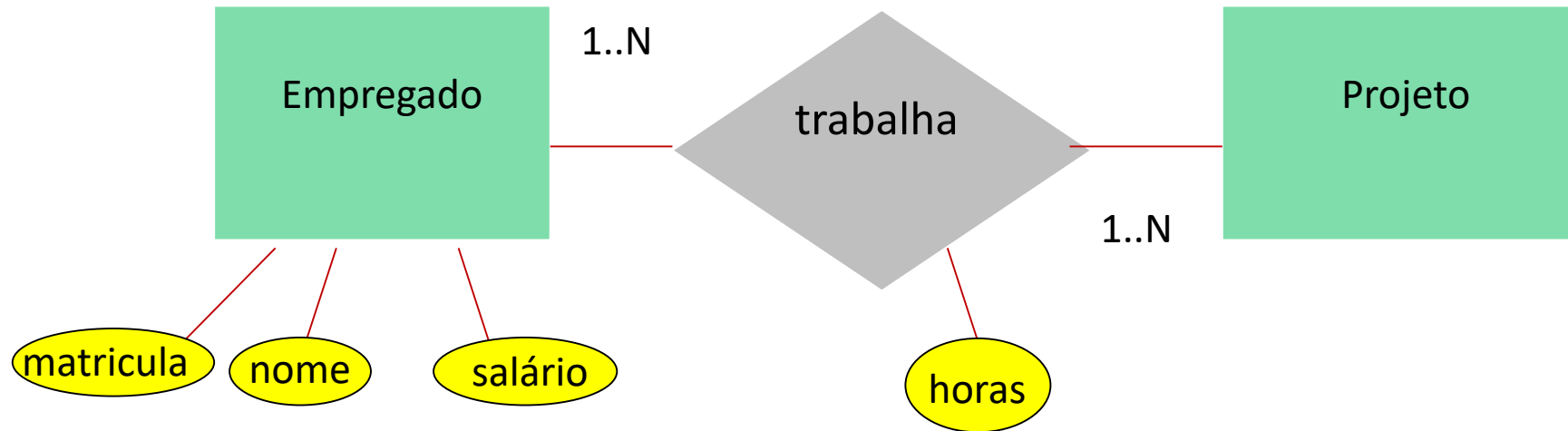
Ex.: A entidade Empregado poderia ter a seguinte instância: “Maria dos Anjos, 31 anos, Secretária, Solteira, R\$ 800,00”

Relacionamentos

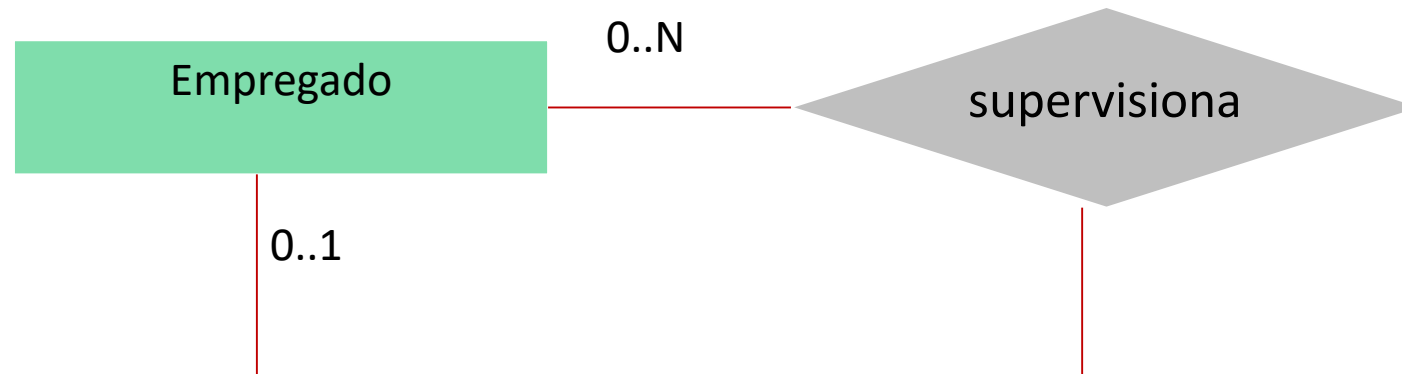
São funções que mapeiam um conjunto de instâncias de uma entidade em um outro conjunto de instâncias de outra entidade (ou da mesma entidade: “auto relacionamento”). Em outras palavras, são associações entre diversas entidades.

Ex.: “Um empregado trabalha num projeto”
“Um cliente possui conta bancária”
“Um filme possui vários atores”

Relacionamentos



Auto-Relacionamentos



Restrições de Integridade

Caracterizam as restrições nas quais os relacionamentos entre entidades estão submetidos (regras do negócio).

Ex.:

“Todo empregado deve estar lotado num departamento”

“Existe Cliente que não foi recomendado por Cliente”

“Toda Nota Fiscal deve ter pelo menos um item discriminado”

“Toda multa deve estar associada a um carro”

“Existe carro sem multa associada”

Restrições de Integridade

- Podemos caracterizar um relacionamento em termos de:
 - 1. Cardinalidade:** quantidade de instâncias que podem participar do relacionamento
 - 2. Totalidade:** obrigatoriedade da ocorrência do relacionamento entre as entidades envolvidas.

Restrições de Integridade

- Tipos de Cardinalidade
 - **Um_para_Um (1:1)**: uma instância de uma entidade A está associada a no máximo a uma instância de uma entidade B, e vice-versa.
 - **Um_para_Muitos (1:N)**: uma instância de uma entidade A está associada a qualquer número de instâncias da entidade B. Porém, uma instância da entidade B pode estar associada, no máximo, a uma instância da entidade A.

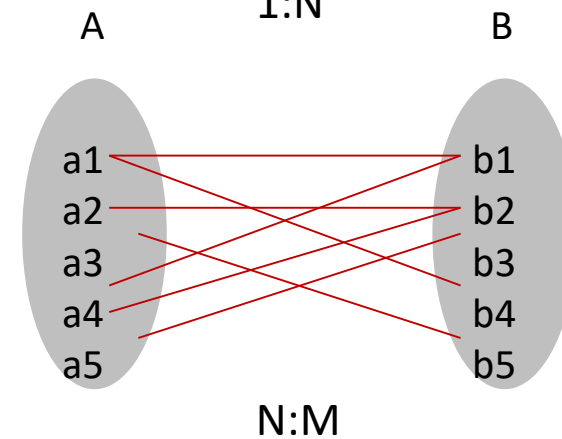
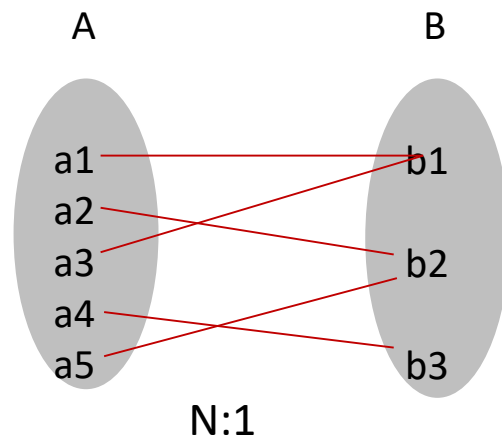
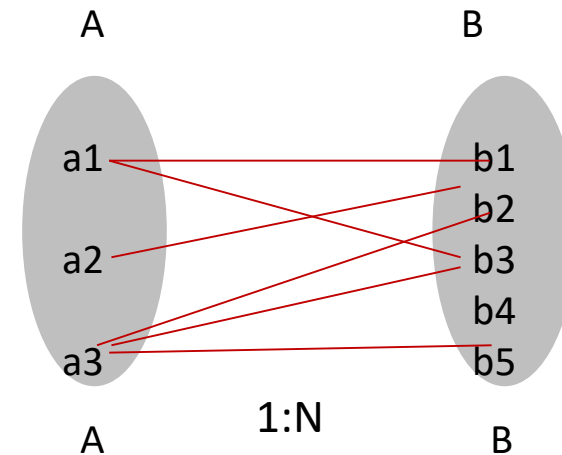
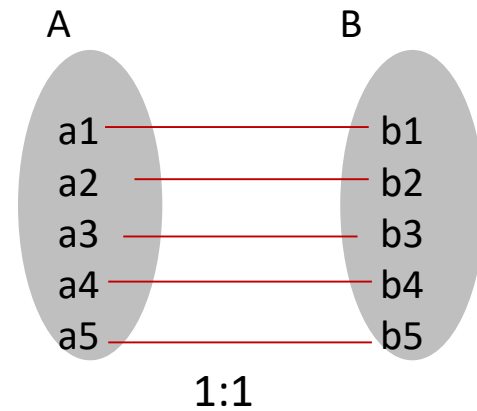
Restrições de Integridade

- Tipos de Cardinalidade

- **Muitos_para_Um (N:1)**: uma instância da entidade A está associada a uma instância de B. Porém, uma instância de B pode estar associada a qualquer número de instâncias de A.
- **Muitos_para_Muitos(M:N)**: uma instância da entidade A está associada a qualquer número de instâncias da entidade B, e vice-versa.

OBS.: o uso de “zero” (0..N) ou “um” (1..N) indica a totalidade do relacionamento.

Restrições de Integridade



Representação clássica Chen

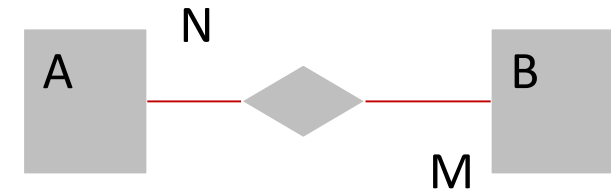
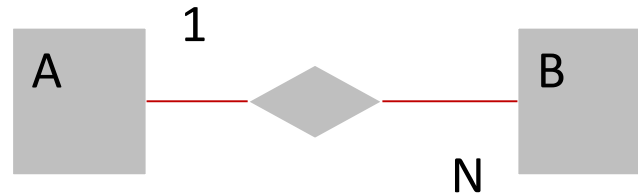
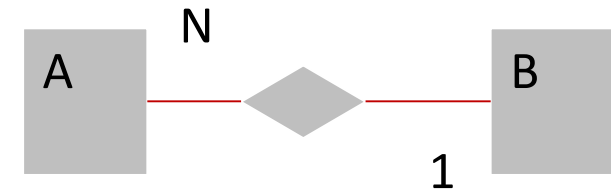
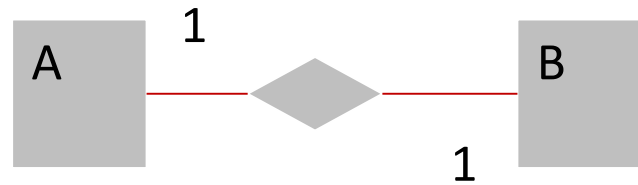
Na **representação clássica de Chen**, usada para modelar banco de dados de forma conceitual:

Entidades são representadas por retângulos.

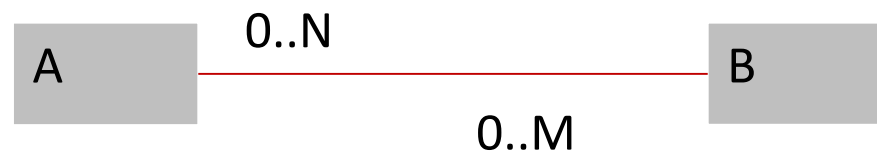
Atributos por elipses.

Relacionamentos por losangos.

Ligações entre eles por linhas.



Representação UML

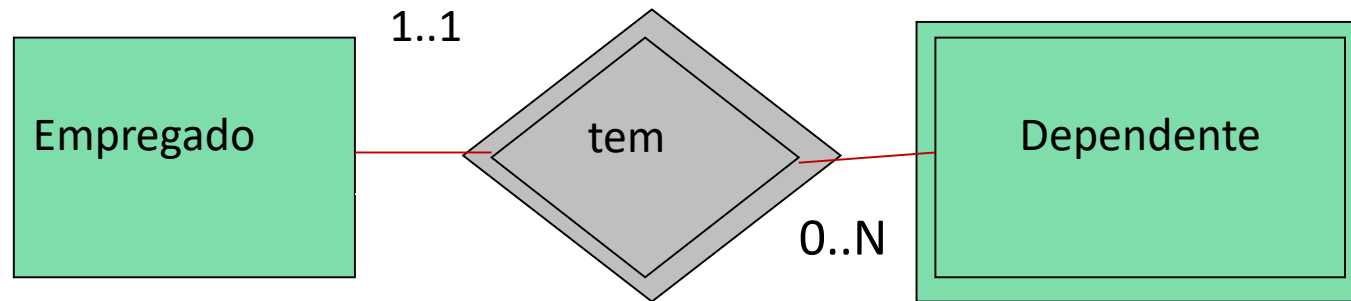


Entidades Fracas

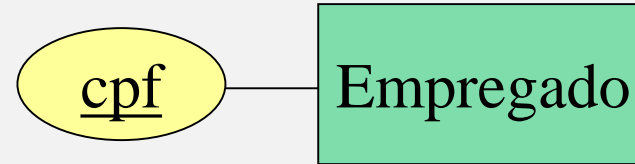
São entidades que são dependentes de existência ou de identificação de um outra entidade. É outra classe de restrição. Se a existência de uma instância x depende da existência de uma outra instância y , então x (instância subordinada) é *dependente de existência* de y (instância dominante), e, portanto, a entidade que contém x é fraca com relação à entidade que contém y . Então, se y for removido, x também o será.

Entidades Fracas

Ex.: Relacionamento entre Empregados e Dependentes



Atributo-Chave



- Como distinguir as instâncias de uma entidade? Num Banco de Dados, isto é feito através dos atributos das entidades que formam as chamadas ***chaves de identificação***.
- Toda instância de uma entidade deve ter uma chave de identificação, que deve ter um valor único e não nulo.

Atributo-Chave Composto

- Vários atributos juntos formam uma chave
- A combinação dos valores dos atributos deve ser única para cada instância da entidade
- Precisa ser mínima, não deve ter atributos supérfluos



Atributo-Chave

- O que fazer quando um atributo-chave está sujeito a mudanças?
 - um departamento mudar de nome?
 - for modificada a estrutura de codificação de Centros de Custos?
 - um departamento mudar de diretoria?

Solução: chave de identificação própria: “surrogate” ou object identification (object id)

Atributo-Chave Substituto

- Surrogates:
 - criados para cada entidade (chave primária)
 - identifica univocamente cada instância da entidade
 - **não** precisa ser percebido pelos usuários
 - **não** é controlado pelos usuários (gerado automaticamente pelo SGBD, através de sequence)

Atributo-Chave Substituto

- **Surrogates:**

Entidade = Identidade + Propriedades

- **Identidade:** chave de identificação própria

- **Propriedades:**

- chaves de identificação definidas pelo usuário
- outras propriedades

Atributo-Chave

- Regras de Integridade:
 - “Toda instância de uma entidade possui um valor para chave de identificação própria da entidade”
 - “O valor da chave de identificação própria para uma instância é único e não nulo dentro da entidade”
 - “O valor da chave de identificação própria de uma instância não pode ser modificado”

Obs.: Surrogates não devem ser reutilizados!!!

MER Estendido

- Superclasses e Subclasses

- ✓ Vimos que uma entidade é usada para representar um conjunto de instâncias do mesmo tipo (Ex. Empregado). Porém, muitas vezes uma entidade tem subentidades que necessitam ser representadas explicitamente.

Ex. Empregado pode ser agrupado em: Secretária, Engenheiro e Técnico

MER Estendido

- Subclasses (Subentidades)
 - ✓ Estas subentidades são subconjuntos da entidade Empregado, ou seja, cada instância de uma subentidade é também uma instância da entidade Empregado. Então dizemos que Secretária **é_uma** Empregada, Engenheiro **é_um** Empregado e Técnico **é_um** Empregado.
 - ✓ Este relacionamento **é_um** caracteriza a **herança**. Ou seja, a subentidade (subclasse) herda todos os atributos e relacionamentos da superentidade (superclasse).

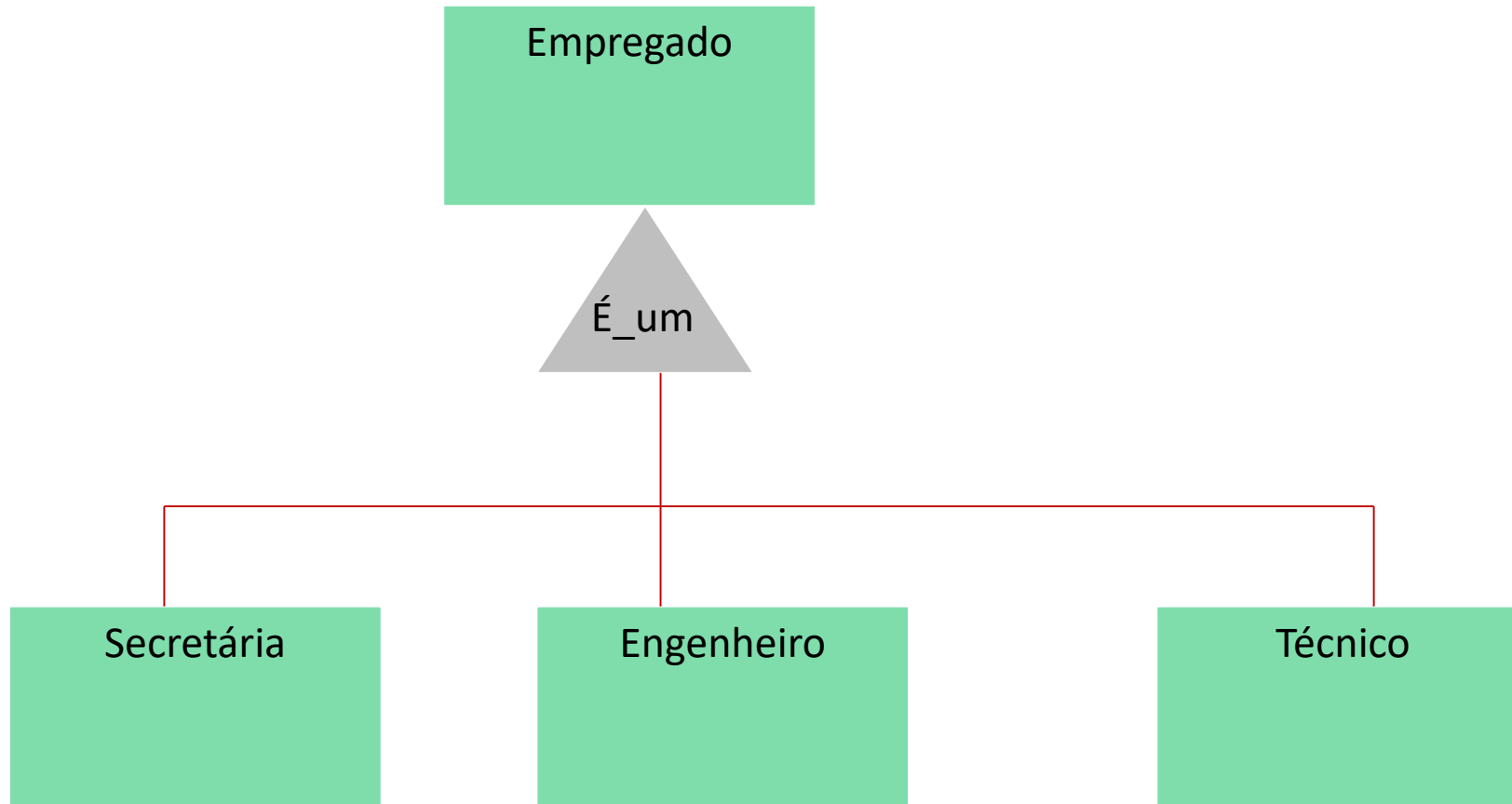
MER Estendido

- Superclasses e Subclasses

- ❑ É importante notar, que nem toda instância da superentidade é membro de uma subentidade.

Ex.: podemos ter empregados que não são nem secretária, nem engenheiro, nem técnico.

MER Estendido



Exercício

- Uma determinada empresa está lhe contratando para projetar o banco de dados que será utilizado para manter dados operacionais. O negócio da empresa é desenvolver projetos para terceiros.
- Projetos são controlados por departamentos e possuem um número, um nome e uma localização. Todo projeto deve ser controlado por um departamento.
- O departamento é caracterizado por um nome e um número e pode ter várias localizações. O número de funcionário de cada departamento deve ser calculado para controle da alta administração. Todo departamento deve ter um gerente que é um empregado da empresa. A data em que o empregado assumiu a gerência de um departamento deve ser armazenada. Empregados trabalham em projetos, porém pertencem a um único departamento. É necessário armazenar a quantidade de horas que um empregado trabalha em um determinado projeto. Todo departamento deve ter pelo menos um empregado e todo empregado deve trabalhar para um departamento.
- Os empregados possuem como atributos: cpf; nome composto por primeiro nome, nome do meio e sobrenome; data de nascimento; endereço, sexo e salário. Um empregado pode supervisionar outros empregados.
- Empregados possuem dependentes que são identificados pelo nome. O sexo, a data de nascimento e parentesco do dependente também são armazenados no banco de dados.