

CBD5951 - Estudo sobre os Modelos Conceituais e Normas Catalográficas no Universo de Metadados Descritivos

Projeto Conceitual de Bancos de Dados: Modelo Entidade-Relacionamento

Kelly Rosa Braghetto
kellyrb@ime.usp.br

Departamento de Ciência da Computação
Instituto de Matemática e Estatística – USP

Agosto de 2023

Bancos de dados

Exemplos

- ▶ Uma agenda de telefones e endereços de seus contatos
- ▶ O catálogo com as informações do acervo de uma biblioteca
- ▶ Os dados de imposto de renda da Receita Federal
- ▶ Os registros de matrículas e notas dos alunos de uma universidade
- ▶ As informações sobre o estoque e as vendas de uma loja
- ▶ Os prontuários dos pacientes de um hospital
- ▶ Os registros meteorológicos coletados na cidade de São Paulo
- ▶ ...

Big Data - Estatísticas

Fonte: <https://www.domo.com/learn/data-never-sleeps-9>



Propriedades implícitas de um banco de dados (BD)

1. Representar um aspecto do mundo real (= minimundo)
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 de usuários em potencial e algumas aplicações pré-concebidas, nas quais esses usuários estão interessados

Softwares para a manutenção de bancos de dados

Um dado BD informatizado pode ser criado e mantido por:

- ▶ um programa de aplicação desenvolvido especificamente para essa tarefa

ou

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

Projeto de bancos de dados

Envolve as seguintes etapas:

1. Levantamento e análise dos requisitos
2. Projeto conceitual
3. Projeto lógico
4. Projeto físico

Fase 1: Levantamento e análise dos requisitos

Nessa fase, o projetista:

- ▶ Registra de forma detalhada e completa os requisitos dos usuários com relação aos dados
- ▶ Identifica também requisitos funcionais conhecidos das aplicações (operações de consulta e modificação sobre o BD)

Sobre os requisitos:

- ▶ São levantados por meio de entrevistas com os produtores e os usuários dos dados
- ▶ Incluem os dados exigidos para processamento, os seus relacionamentos naturais e as informações relevantes para a escolha da plataforma de software para o BD

Fase 2: Projeto conceitual

Fase de criação de um esquema conceitual para o BD, utilizando um modelo de dados conceitual (= de alto nível).

Esquema conceitual (definição):

Descrição concisa de requisitos de dados dos usuários, contendo descrições detalhadas sobre os tipos de entidades, relacionamentos e restrições, expressos de acordo com os conceitos existentes no modelo de dados conceitual empregado

Fase 2: Projeto conceitual

Características de um esquema conceitual:

- ▶ Fácil compreensão (não envolve detalhes de implementação)
- ▶ Pode ser usado na comunicação com usuários não técnicos

Exemplos bastante usados:

- ▶ **Modelo Entidade-Relacionamento (ER)**
- ▶ Diagramas de Classe da UML

Fase 3: Projeto lógico

Essa fase corresponde ao mapeamento do esquema conceitual para um modelo de dados de implementação e (opcionalmente) o refinamento do esquema.

⇒ Passo necessário para a implementação do BD utilizando um Sistema Gerenciador de Bancos de Dados.

Exemplos de modelos de implementação bastante usados:

- ▶ Modelo Relacional
- ▶ Modelo Objeto-Relacional

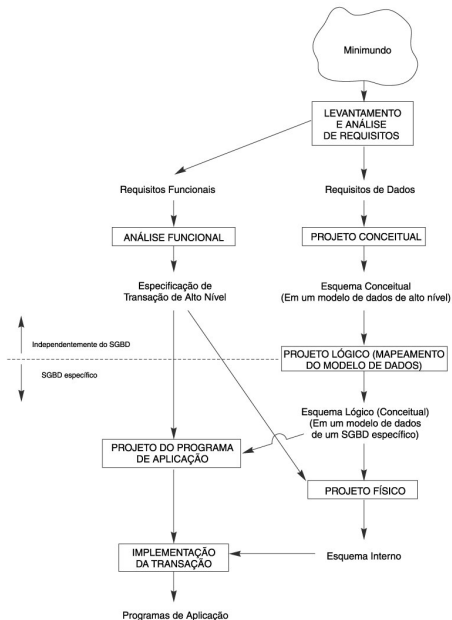
Fase 4: Projeto físico

Fase na qual são definidas as estruturas de armazenamento interno, índices, configurações de arquivos (para os arquivos do BD) e outros ajustes finos.

- ▶ Finalidade: otimizar o desempenho das operações de consulta e manipulação dos dados
- ▶ Pode até mesmo modificar o projeto de BD resultante das fases anteriores, a fim de satisfazer critérios de desempenho desejados

Principais fases do projeto de um banco de dados

Fonte da imagem: *Sistemas de Bancos de Dados* (6ª edição), Elmasri e Navathe. Pearson, 2010.



Abstrações de dados

A modelagem conceitual de dados está relacionada à representação dos dados por meio de um conjunto de abstrações.

Abstrações têm:

- ▶ **Estrutura:** do ponto de vista estrutural, abstrações são detalhes que são deliberadamente omitidos numa dada visão de implementação para facilitar a visão global dos dados.
- ▶ **Comportamento:** do ponto de vista comportamental, abstrações determinam restrições, regras e operações que devem ser tomadas (automaticamente) na manipulação do conjunto de dados.

Modelo Entidade-Relacionamento (ER)

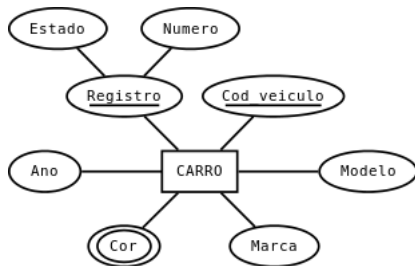
- ▶ Criado por Peter Chen em 1976
- ▶ É simples e de interpretação intuitiva (mesmo para usuários não-especialistas)
- ▶ É o modelo mais usado para a modelagem conceitual de BDs
- ▶ Descreve os dados com base em três conceitos principais:
 - ▶ **entidades**
 - ▶ **relacionamentos**
 - ▶ **atributos**

Entidades e Atributos

- ▶ **Entidade** – “algo” do mundo real, com uma existência independente. Pode ter existência física (ex.: uma pessoa, uma casa, um carro, etc.) ou conceitual (ex.: um projeto, uma disciplina).
- ▶ **Atributos** – propriedades particulares que descrevem uma entidade (ex.: nome da pessoa, endereço, função, etc.). Um dada entidade terá um valor para cada um de seus atributos. Esses valores constituem a maior parte dos dados armazenados em um BD.
- ▶ **Relacionamento** – uma associação entre entidades (de um mesmo tipo ou de tipos diferentes)

Notação para tipos de entidades e atributos no modelo ER

Exemplo de tipo de entidade e seus atributos



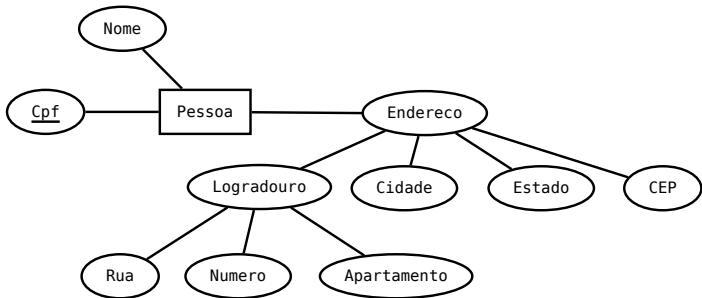
Tipos de atributos no modelo ER

Simple (atômicos) × Compostos

- ▶ **Atributos compostos** – podem ser divididos em partes menores. Ex.: o atributo endereço pode ser dividido em Rua, Cidade, Estado e CEP.
 - ▶ O seu valor é dado pela concatenação dos valores dos atributos simples que os compõem.
 - ▶ Devem ser usados quando o usuário algumas vezes se refere ao atributo como um grupo, mas em outras se refere a um de seus componentes.
(Caso contrário, o endereço, por exemplo, poderia ser definido como um atributo simples).
- ▶ **Atributos simples (ou atômicos)** – os atributos que não são divisíveis.

Tipos de atributos no modelo ER

Exemplo de hierarquia de atributos compostos



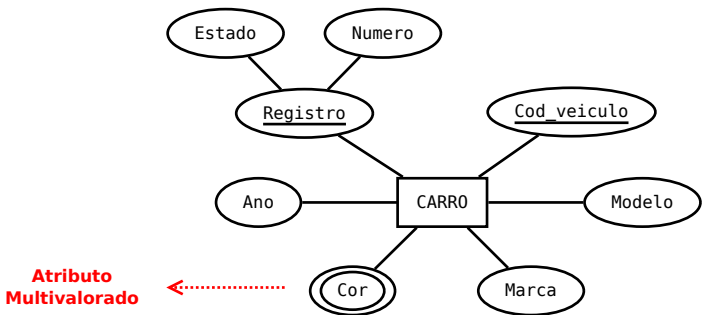
Tipos de atributos no modelo ER

Monovalorados × Multivalorados

- ▶ **Monovalorado** – atributo que tem um único valor para uma dada entidade.
 - ▶ **Ex.:** o atributo idade para uma pessoa.
- ▶ **Multivalorado** – atributo para o qual diferentes entidades podem ter diferentes quantidades de valores.
 - ▶ **Ex.:** atributo titulação para uma pessoa; algumas pessoas não possuem título algum, outras possuem 1, ou 2, ...
 - ▶ Um atributo multivalorado pode ter um **limite inferior e superior** pra restringir o número de valores permitidos a cada entidade individual.

Notação para tipos de entidades e atributos no modelo ER

Exemplo de atributo multivalorado

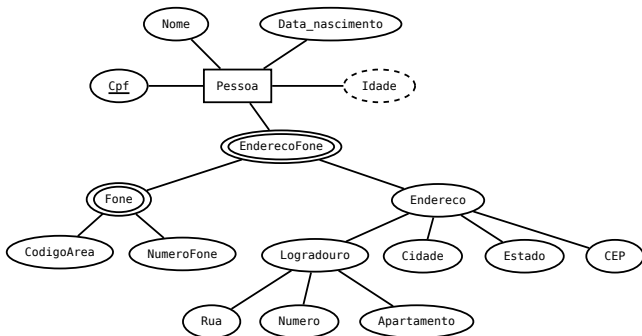


Tipos de atributos do modelo ER

Atributos Complexos

São atributos compostos e multivalorados, que podem ser aninhados de modo arbitrário.

- ▶ Ex.: uma pessoa pode ter mais de uma residência e cada uma delas pode ter múltiplos telefones



Tipos de atributos no modelo ER

Armazenados × Derivados

- ▶ **Atributo derivado** – é derivado a partir de outro(s) atributo(s) ou entidade(s) relacionado(s). Exemplos:
 - ▶ o atributo **idade** é derivado a partir da data de nascimento e da data corrente
 - ▶ o atributo **número de funcionários** de um departamento pode ser derivado da contagem do número de funcionários associados ao departamento
- ▶ **Atributo armazenado** – é um atributo que não é derivado.

Notação para tipos de entidades e atributos no modelo ER

Exemplo de atributo derivado



Domínio de valores dos atributos

- ▶ Cada atributo simples de um tipo de entidade possui um **domínio de valores**, que determina o conjunto de valores válidos que ele pode assumir em cada entidade.
- ▶ Os domínios de valores não são mostrados no diagrama ER.

Valor NULL (nulo)

- ▶ É um valor especial, usado quando uma entidade **não possuiu um valor** para um atributo.
- ▶ O NULL serve tanto para indicar que um atributo **não se aplica** a uma dada entidade, quanto para indicar que o valor para um atributo de uma dada entidade é **desconhecido**.
- ▶ “Desconhecido” se aplica a dois casos distintos:
 - ▶ quando é sabido que existe um valor para o atributo, mas ele está faltando (**ex.:** Altura – todo mundo tem!)
 - ▶ quando não é sabido se o valor existe ou não (**ex.:** FoneResidencial – uma pessoa pode ou não ter)

Tipo de entidade

Um **tipo de entidade** define um conjunto de entidades que possuem os mesmos atributos.

Conceitos importantes:

- ▶ **Atributo-chave (restrição de exclusividade)** – é um atributo para o qual o valor é distinto para toda entidade do conjunto de entidades do tipo a que pertence.
 - ▶ Os valores do atributo-chave **identificam cada entidade univocamente**.

Atributo-chave de um tipo de entidade

Conceitos importantes:

- ▶ **Chave composta** – é formada por diversos atributos, cuja combinação dos valores é distinta para cada entidade.
Uma chave composta deve ser minimal – todos os atributos componentes devem ser necessários para a garantia da unicidade.
- ▶ Alguns tipos de entidade têm mais de um atributo chave (ex.: os atributos NUSP e CPF para ALUNO).
Outros, podem nem ter uma chave – são os chamados **tipos de entidade fraca**.

Tipos de Entidade

Exemplo de dois tipos de entidade (com chaves simples) e algumas entidades membro de cada um deles

Chave de Empregado: nome

Chave de Empresa: nome

NOME DO TIPO ENTIDADE:

EMPREGADO

Nome, Idade, Salario

EMPRESA

Nome, Sede Administrativa, Presidente

**CONJUNTO DE ENTIDADE:
(EXTENSÃO)**

e_1 ●

(John Smith, 55, 80k)

e_2 ●

(Fred Brown, 40, 30K)

e_3 ●

(Judy Clark, 25, 20K)

⋮

c_1 ●

(Sunco Oil, Houston, John Smith)

c_2 ●

(Fast Computer, Dallas, Bob King)

⋮

Tipos de Entidade

Conjunto de entidades do tipo de entidade CARRO (com uma chave composta e outra simples)

Chaves de Carro: Registro(NumeroRegistro, Estado) ou IdVeiculo

CARRO
Registro(NumeroRegistro, Estado), IDVeiculo, Marca, Modelo, Ano, {Cor}

car₁ •
((ABC 123, TEXAS), TK629, Ford Mustang, conversível, 1998, {vermelho, preto})

car₂ •
((ABC 123, NOVA YORK), WP9872, Nissan Maxima, 4 portas, 1999, {azul})

car₃ •
((VSY 720, TEXAS), TD729, Chrysler LeBaron, 4 portas, 1995, {branco, azul})

⋮

Exemplo: BD EMPRESA

Requisitos de dados - Parte 2

- ▶ Armazenamos o nome (primeiro nome, inicial do meio e último nome), número do Cadastro de Pessoa Física (CPF), endereço, salário, sexo (gênero) e data de nascimento de cada funcionário. Um funcionário é designado para um departamento, mas pode trabalhar em vários projetos, que não necessariamente são controlados pelo mesmo departamento. Registramos o número atual de horas por semana que um funcionário trabalha em cada projeto. Também registramos o supervisor direto de cada funcionário (que é outro funcionário).
- ▶ Queremos registrar os dependentes de cada funcionário para fins de seguro. Para cada dependente, mantemos o nome, sexo, data de nascimento e parentesco com o funcionário.

Diagrama ER inicial para o esquema EMPRESA



Tipo de relacionamento

Um **tipo de relacionamento** R entre n tipos de entidades E_1, E_2, \dots, E_n define um conjunto de associações (= relacionamentos) entre as entidades desses tipos.

- ▶ Matematicamente, um tipo de relacionamentos R é um conjunto de instâncias de relacionamento r_i , onde cada r_i associa n entidades individuais (e_1, e_2, \dots, e_n) , e cada entidade e_j em r_i é um membro do tipo de entidade E_j , $1 \leq j \leq n$.
- ▶ Cada tipo de entidade E_j é dito **participante** do tipo de relacionamento R , da mesma forma que cada entidade e_j é dita participante da instância de relacionamento r_i .
- ▶ Informalmente, cada r_i em R representa uma associação de entidades existente no minimundo, que inclui exatamente uma entidade de cada tipo de entidade participante.

Notação para tipos de relacionamento

Exemplo: relacionamento TRABALHA_PARA entre os tipos de entidade FUNCIONÁRIO e DEPARTAMENTO



Propriedades de um tipo de relacionamento

Nome do papel

- ▶ Indica o papel que cada tipo de entidade desempenha em um tipo de relacionamento.
 - ▶ **Ex.:** no tipo de relacionamento TRABALHA_PARA, o FUNCIONÁRIO desempenha o papel de *trabalhador*, enquanto o DEPARTAMENTO desempenha o papel de *empregador*.

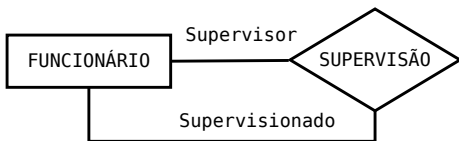


Propriedades de um tipo de relacionamento

Nome dos papéis em relacionamentos recursivos

- ▶ Um mesmo tipo de entidade pode participar mais de uma vez em um mesmo tipo de relacionamento (= **relacionamento recursivo** ou **auto-relacionamento**).
- ▶ Em relacionamentos recursivos, os nomes dos papéis são fundamentais para definir o sentido de cada participação.

Ex.: tipo de relacionamento SUPERVISÃO, em que o tipo de entidade FUNCIONÁRIO participa duas vezes – uma no papel de *supervisor*, outra no papel de *supervisionado*.



Restrições sobre tipos de relacionamento binários

Existem restrições (determinadas por situações do minimundo) que limitam as combinações de entidades que podem participar de um relacionamento binário.

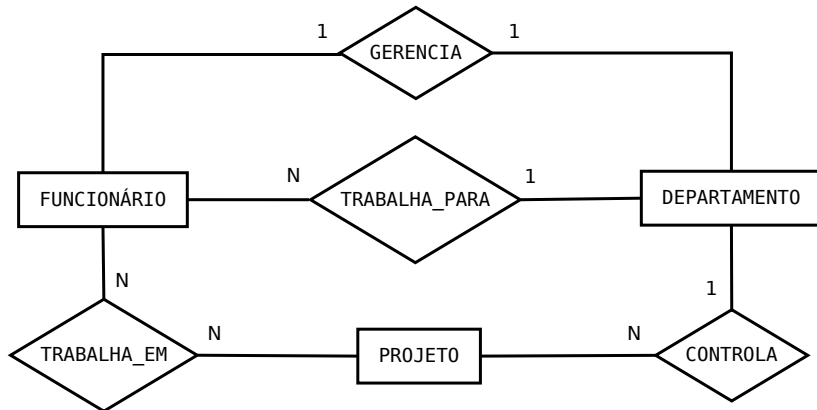
Ex.: um funcionário pode trabalhar apenas para um único departamento.

Restrições possíveis para relacionamentos binários:

- ▶ Razão de cardinalidade
- ▶ Restrição de participação

Notação da razão da cardinalidade

Exemplos de tipos de relacionamento com diferentes cardinalidades



Restrições sobre tipos de relacionamento binários

Razão de cardinalidade

- ▶ Especifica o **número máximo** de instâncias do relacionamento em que uma entidade pode participar.
- ▶ As razões de cardinalidade possíveis são
 - ▶ 1:1 (um para um)
 - ▶ 1:N (um para muitos)
 - ▶ N:1 (muitos para um)
 - ▶ M:N (muitos para muitos)

Restrições sobre tipos de relacionamento binários

Razão de cardinalidade – Exemplo:

- ▶ No relacionamento TRABALHA_PARA, DEPARTAMENTO:FUNCIONÁRIO tem razão 1:N
 - ▶ Um departamento pode empregar qualquer número de funcionários (= 0 ou mais = “muitos”)
 - ▶ Um funcionários pode trabalhar para no máximo um departamento (= 0 ou 1)

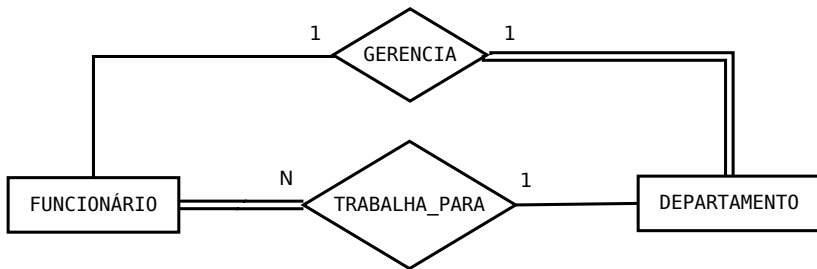
Restrições sobre tipos de relacionamento binários

Restrição de participação

- ▶ Determina o **número mínimo** de instâncias de relacionamento em que uma entidade deve participar.
 - ▶ Também é chamada de *restrição de cardinalidade mínima*
- ▶ A participação pode ser:
 - ▶ **total** – ex.: todo funcionário deve trabalhar para um departamento, ou seja, um funcionário só pode existir se estiver associado a um departamento. Assim, FUNCIONÁRIO tem participação total no relacionamento. Esse tipo de participação também é conhecido como **dependência de existência**.
 - ▶ **parcial** – ex.: nem todo funcionário gerencia um departamento. Assim, FUNCIONÁRIO tem uma participação parcial no tipo de relacionamento GERENCIA.

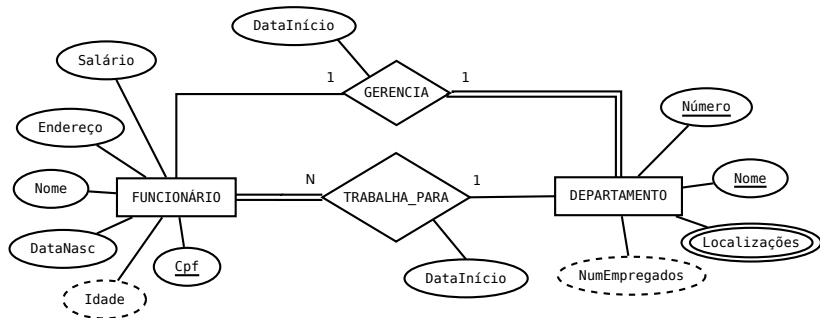
Notação da restrição de participação

Exemplos de tipos de relacionamento envolvendo diferentes restrições de participação



Atributos de tipos de relacionamento

Tipos de relacionamento podem ter atributos, de forma similar aos tipos de entidade.

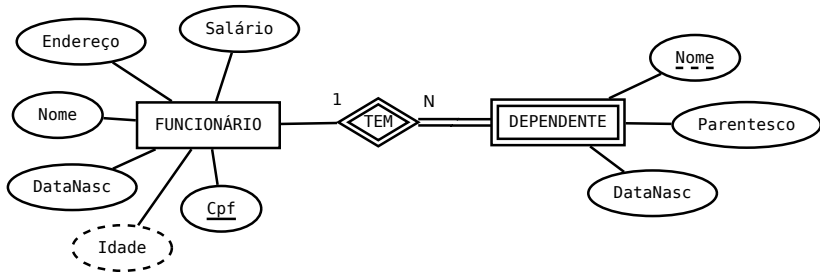


Tipo de entidade fraca

- ▶ **Tipo de entidade forte** – tipo de entidade que possui um atributo-chave.
- ▶ **Tipo de entidade fraca** – tipo de entidade que não possui um atributo-chave.
 - ▶ Entidades de tipos de entidade fraca são identificadas por estarem relacionadas (associadas) a entidades de um outro tipo de entidade (chamado de **tipo de entidade identificador** ou **tipo de entidade proprietária**).
 - ▶ Esse tipo de relacionamento é chamado de **relacionamento identificador** do tipo de entidade fraca.

Notação dos tipos de entidade fraca

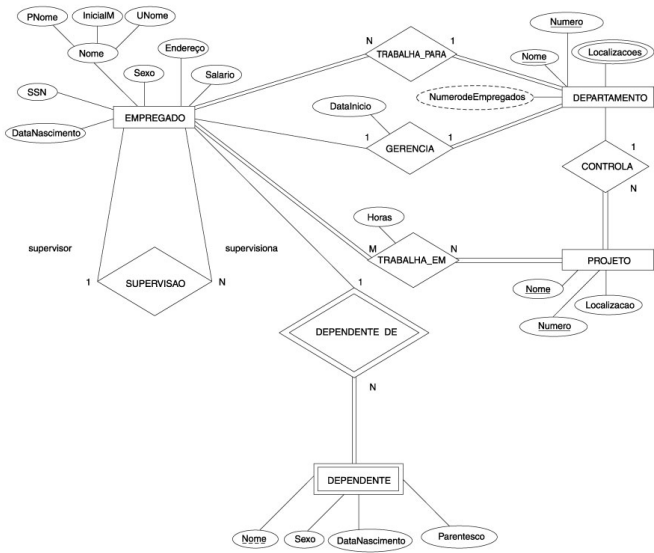
Exemplo: Tipo de entidade fraca DEPENDENTE em um relacionamento identificador com FUNCIONÁRIO



Tipo de entidade fraca

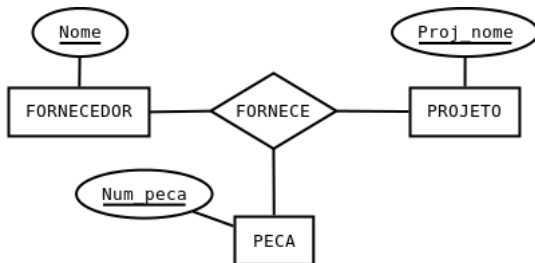
- ▶ Um tipo de entidade fraca sempre tem uma restrição de participação total em relação ao seu relacionamento identificador.
- ▶ Um tipo de entidade fraca normalmente tem uma **chave parcial**, que é um conjunto de atributos que identifica univocamente as entidades fracas que estão relacionadas a uma mesma entidade proprietária.
 - ▶ No pior caso, a chave parcial será a composição de todos atributos do tipo de entidade fraca.
- ▶ Quando um tipo de entidade fraca não é participante em tipos de relacionamento, então ele pode ser definido como um atributo complexo (composto, multivalorado) em seu tipo de entidade proprietária.

Diagrama ER para o esquema EMPRESA



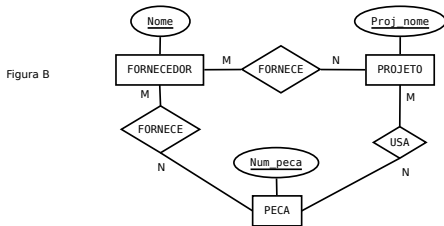
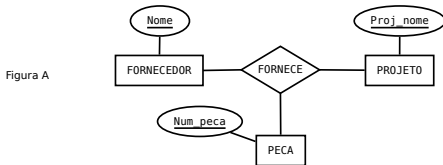
Tipos de relacionamento de grau maior que dois

Exemplo: tipo de relacionamento FORNECE



Tipos de relacionamento de grau maior que dois

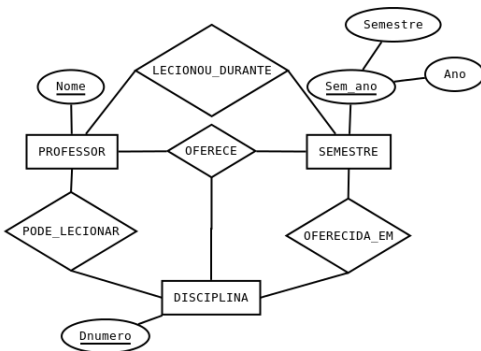
Exemplo: tipo de relacionamento FORNECE



Os 3 relacionamentos binários (Fig. B) **NÃO** são equivalentes ao ternário da Fig. A.

Tipos de relacionamento de grau maior que dois

Exemplo: tipo de relacionamento OFERECE



Existe alguma redundância no modelo acima?

Modelo Entidade-Relacionamento Estendido

Contexto (década de 1980)

- ▶ Desejo: projeto de bancos de dados que refletisse mais precisamente as restrições de dados
- ▶ Necessidade de conceitos adicionais (+ abstrações!) para a *modelagem semântica de dados*
- ▶ Novos modelos criados em áreas como as de *representação do conhecimento* (IA) e *modelagem de objetos* (ES)
- ▶ Resultado: **modelo ER estendido** (ou **EER**)

Conceitos do modelo EER

- ▶ Todos os conceitos do modelo ER
- ▶ **Subclasse / Superclasse**
 - ▶ **Especialização / Generalização**
 - ▶ Herança de atributo e relacionamento
- ▶ **Categoria (ou Tipo de União)**

EER – Subclasses e superclasses

- ▶ **Classe** – conjunto de entidades; isso inclui qualquer construção do EER que agrupe as entidades, como os *tipos de entidade*, *subclasses*, *superclasses* e *categorias*
- ▶ Uma **subclasse** S é um subconjunto das entidades de outra classe, chamada de **superclasse**. Portanto, no relacionamento classe/subclasse indicado como C/S , $S \subseteq C$.
- ▶ Uma entidade pode ser membro de várias subclasses
- ▶ Uma entidade membro de uma subclasse herda todos os atributos e relacionamentos da sua superclasse
- ▶ Uma subclasse também é considerada um tipo de entidade

Especialização – exemplo

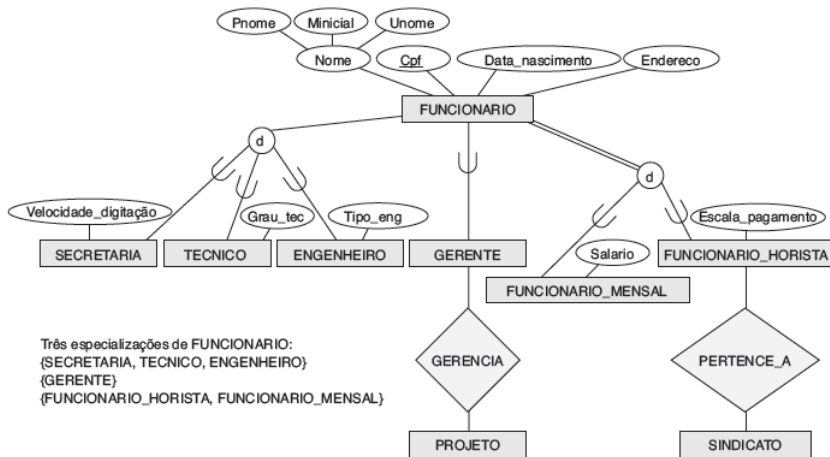
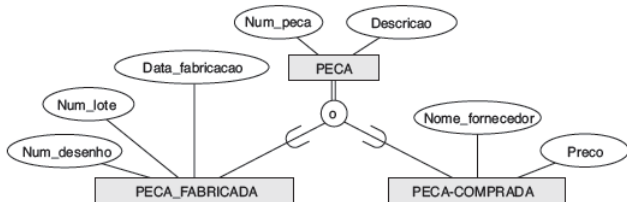


Diagram EER para representar subclasses e especialização

Especialização – exemplo



Especialização sobreposta

EER – especialização

- ▶ Uma **especialização** $Z = \{S_1, S_2, \dots, S_n\}$ é um conjunto de subclasses que têm a mesma superclasse Z , isto é, Z/S_i , para todo i em $\{1, 2, \dots, n\}$.
- ▶ Z é chamada da **superclasse da especialização**, ou de **generalização das subclasses** $\{S_1, S_2, \dots, S_n\}$.
- ▶ Uma especialização Z é **total** sempre que: $\bigcup_{i=1}^n S_i = Z$.
Do contrário, Z é **parcial**.
- ▶ Z é **disjunta** (símbolo **d** no círculo) sempre que tivermos $S_i \cap S_j = \emptyset$ para $i \neq j$.
Do contrário, Z é **sobreposta** (símbolo **o** no círculo).

EER – especialização / generalização

- ▶ **Especialização** – é o processo de definir um **conjunto de subclasses** de um tipo de entidade, chamado de **superclasse** da especialização.

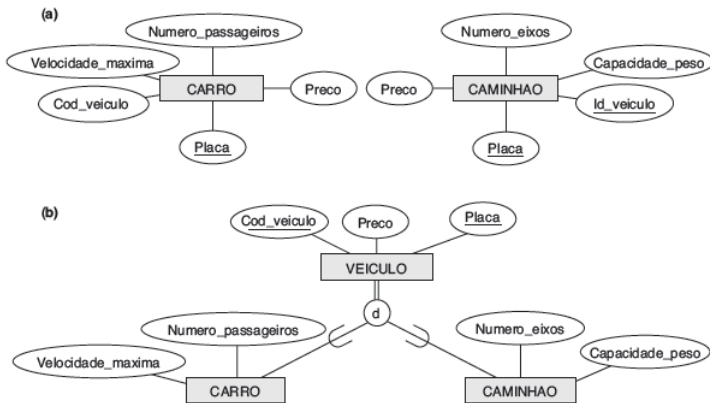
Especialização = refinamento conceitual

- ▶ **Generalização** – é o processo inverso, de suprimir as diferenças entre vários tipos de entidade (subclasses) e identificar suas características comuns, generalizando-as em uma única superclasse.

Generalização = síntese conceitual

O relacionamento entre uma subclasse e sua superclasse é chamado de “**É-UM**”.

Generalização – exemplo



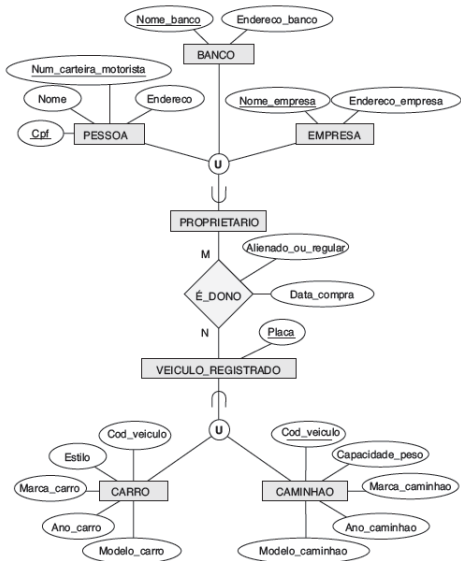
- (a) Tipos de entidade CARRO e CAMINHAO.
- (b) Generalizando esses tipos em VEICULO.

Abstração de especialização/generalização

Razões para se usar especializações/generalizações no modelo EER

1. quando certos atributos podem ser usados somente em algumas das entidades da superclasse. Uma subclasse é definida de modo a agrupar as entidades para as quais esses atributos se aplicam;
2. quando apenas as entidades que são membros de uma subclasse podem participar de algum tipo de relacionamento.

EER – Categorização



Duas categorias:
PROPRIETARIO e
VEICULO_REGISTRADO

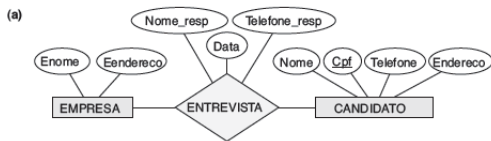
EER – categoria ou tipo de união

- ▶ Uma **categoria** T é formalmente definida como uma classe que é um subconjunto da união de n superclasses definidas D_1, D_2, \dots, D_n , $n > 1$, e é especificada como:

$$T \subseteq (D_1 \cup D_2 \cup \dots \cup D_n)$$
- ▶ Um predicado p_i nos atributos de D_i pode ser usado para especificar os membros de cada D_i que são membros de T . Se um predicado for especificado em todo D_i , temos:

$$T = (D_1[p_1] \cup D_2[p_2] \cup \dots \cup D_n[p_n])$$
- ▶ Em uma **categoria total**, $T = (D_1 \cup D_2 \cup \dots \cup D_n)$, ou seja, T controla a **união de todas as entidades** em suas superclasses.
- ▶ Em uma **categoria parcial**, T pode controlar um **subconjunto da união**.

Exemplo: Dados de Entrevistas



O tipo de relacionamento ENTREVISTA



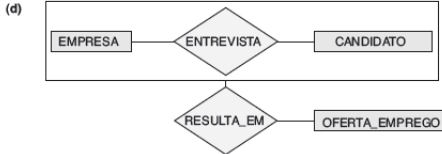
Inclusão de OFERTA_EMPREGO em relacionamento ternário (**incorreto!**)



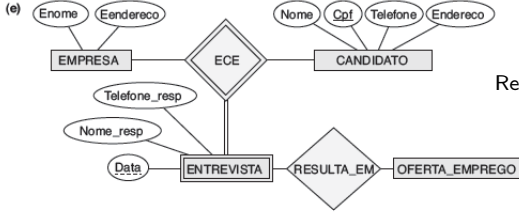
Relacionamento participando de outro relacionamento (**incorreto!**)

Abstração de agregação

A agregação consiste em combinar objetos relacionados por meio de uma instância de relacionamento em particular em um **objeto agregado de alto nível**. Esse caso não pode ser representado de forma explícita no modelo EER.



Usando objeto agregado



Representação alternativa, válida em ER

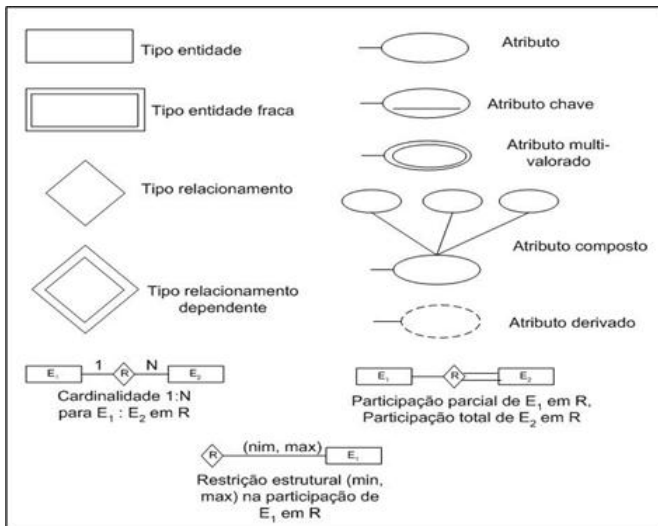
Abstração de agregação

Um objeto agregado de alto nível faz-se especialmente necessário quando:

- ▶ quando o relacionamento que o deu origem deve ter o seu próprio identificador
- ▶ quando pode haver mais de uma instância de seu tipo de relacionamento gerador envolvendo as mesmas entidades
- ▶ ele mesmo está associado a um outro objeto por meio de um relacionamento
- ▶ ele possui atributos que não são comuns a todas as instâncias do seu relacionamento gerador

O relacionamento entre os objetos primitivos e seu objeto agregado é chamado de **É-UMA-PARTE-DE**.

Resumo da notação ER



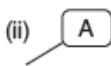
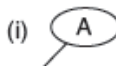
Notações alternativas para modelos ER

Símbolos para tipo de entidade, atributo e relacionamento

Símbolos de tipo/
classe de entidade



Símbolos de atributo

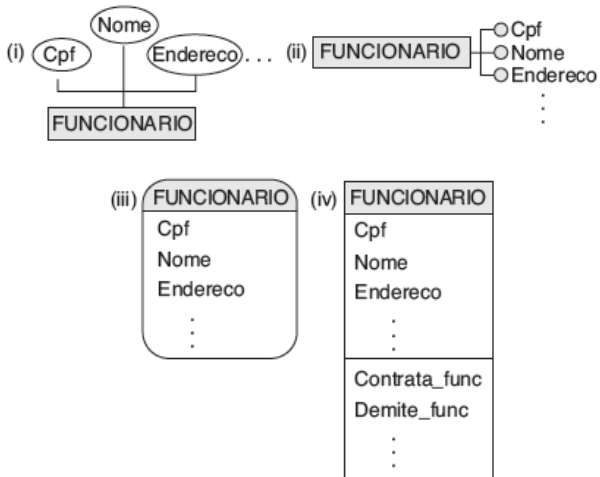


Símbolos de relacionamento



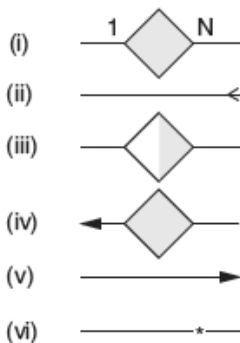
Notações alternativas para modelos ER

Símbolos para tipo de entidade, atributo

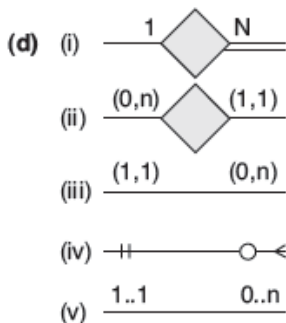


Notações alternativas para modelos ER

Exibindo cardinalidades



Notações (min,max)



Referências Bibliográficas

Principal

- ▶ *Sistemas de Bancos de Dados* (7ª ed.). Elmasri e Navathe. Pearson, 2016.
Capítulos 3 e 4

Complementares

- ▶ *Projeto de Banco de Dados* (6ª ed.). Heuser. Bookman, 2008.
- ▶ *Bancos de Dados: Aprenda o que são, melhore seu conhecimento, construa os seus*. Setzer e Silva. Edgar Blucher, 2005.
- ▶ *Projeto e Modelagem de Bancos de Dados*, Teorey, Lightstone e Nadeau, 2007.