

Banco de Dados

Profa. Leticia T. M. Zoby

leticia.zoby@udf.edu.br

Introdução

- O projeto lógico consta da transformação de um modelo ER em um modelo lógico, que implementa, a nível de SGBD relacional, os dados representados abstratamente no modelo ER.

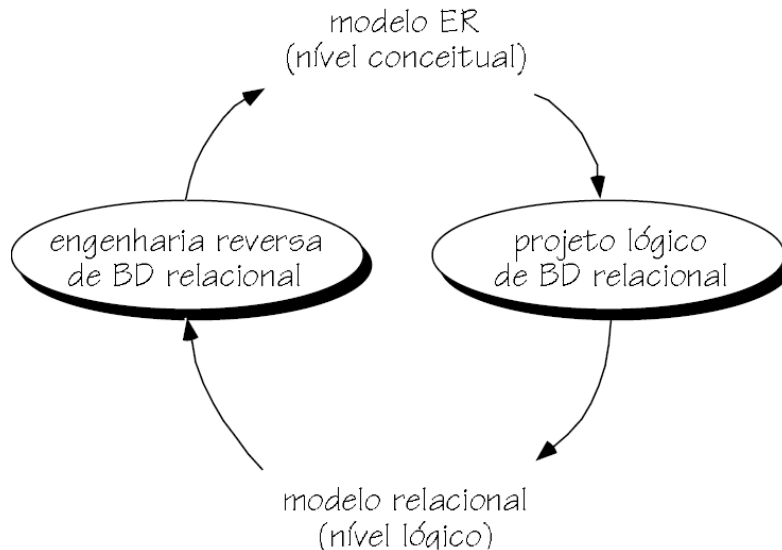


Figura: Transformações entre modelo ER e modelo relacional (Heuser, 1998).

Introdução

- Objetivos da Transformação:
 - Melhorar a performance do BD
 - Evitar Junções
 - Diminuir número de chaves
 - Exemplo: Considerando que se deseja armazenar dados sobre clientes em um banco de dados relacional. Deseja-se armazenar, para cada cliente, seu código, seu nome, o nome da pessoa de contato, o endereço e o telefone.

Cliente (CodCliente, Nome, NomeContato, Endereço, Telefone)

ou

Cliente (CodCliente, Nome, NomeContato)
ClienteEnder (CodCliente, Endereço, Telefone)
CodCliente referencia Cliente

- Evitar campos opcionais
- Simplificar o desenvolvimento do Software

Introdução

- A transformação de um modelo ER em um modelo relacional dá-se nos seguintes passos:
 - Passo1: Tradução das entidades e seus atributos
 - Passo2: Tradução dos relacionamentos e seus atributos
 - Passo3: Tradução da especialização/ generalização

Introdução

- A transformação de um modelo ER em um modelo relacional dá-se nos seguintes passos:
 - Passo1: Tradução das entidades e seus atributos
 - Passo2: Tradução dos relacionamentos e seus atributos
 - Passo3: Tradução da especialização/ generalização

Introdução

Passo1: Tradução das entidades e seus atributos

- Cada entidade irá se transformar em uma tabela
- Cada atributo da entidade se transformará em uma coluna desta tabela
- Atributos identificadores irão corresponder à chave primária da tabela

Introdução

Passo1: Tradução das entidades e seus atributos

Ex:



Pessoa (CodigoPess,Nome,Endereço,DataNasc,DataAdm)

Introdução

Nomes de atributos e nomes de colunas

- Não é aconselhável simplesmente transcrever os nomes de atributos para nomes de colunas. Manter nomes **curtos** nas colunas
- Nome de coluna não pode conter espaços em branco
- Nomes de colunas que podem se repetir em outras tabelas colocar o sufixo com o nome da tabela.
- Ex: data de nascimento -> DataNasc

Introdução

Implementação de relacionamentos

- A decisão de qual estratégia utilizar vai depender da cardinalidade(máxima e mínima)
- Existem diferentes estratégias para a transformação de relacionamentos em modelo lógico
- As estratégias são:
 - Tabela própria
 - Adição de coluna
 - Fusão de tabelas

Introdução

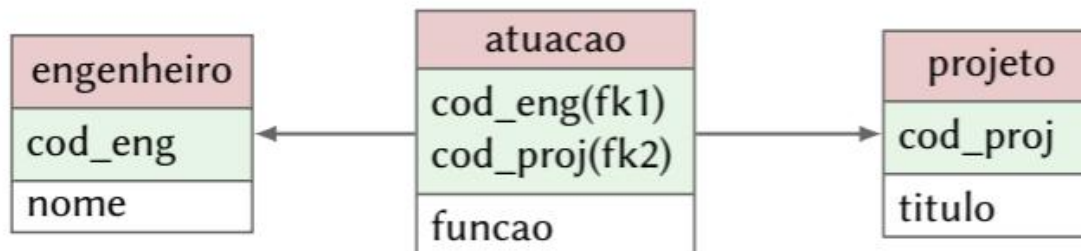
Implementação de relacionamentos – Estratégias:

- Tabela própria

Diagrama ER:



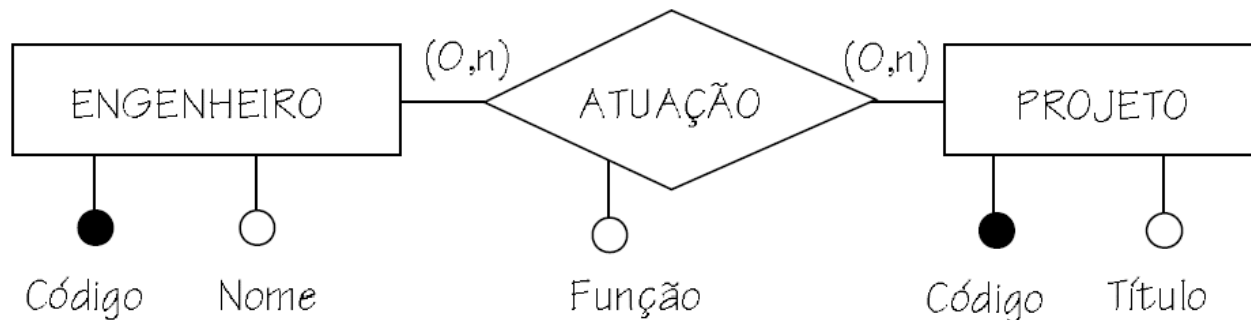
Esquema relacional correspondente:



Introdução

Implementação de relacionamentos – Estratégias:

- Tabela própria



Esquema relacional correspondente:

Engenheiro (CodEng, Nome)

Projeto (CodProj, Título)

Atuação (CodEng, CopProj, Função)

CodEng referencia Engenheiro

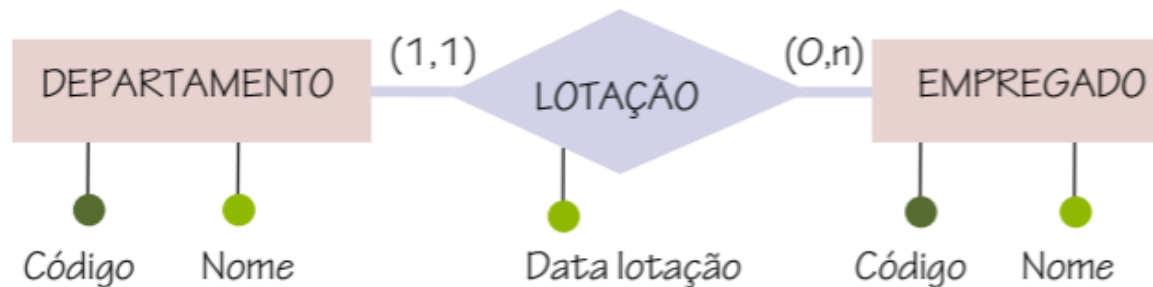
CodProj referencia Projeto

Introdução

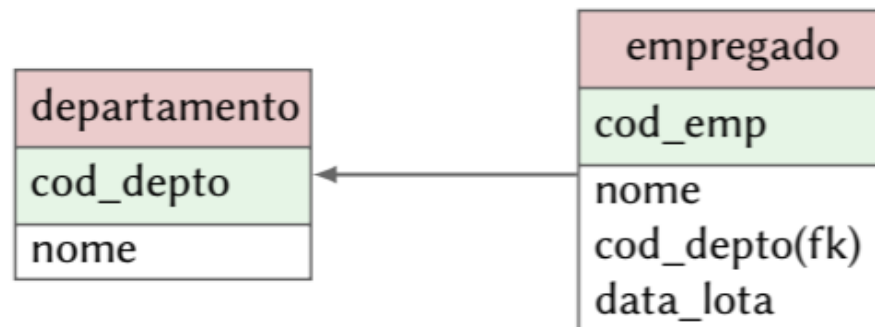
Implementação de relacionamentos – Estratégias:

- Adição de coluna

Diagrama ER:



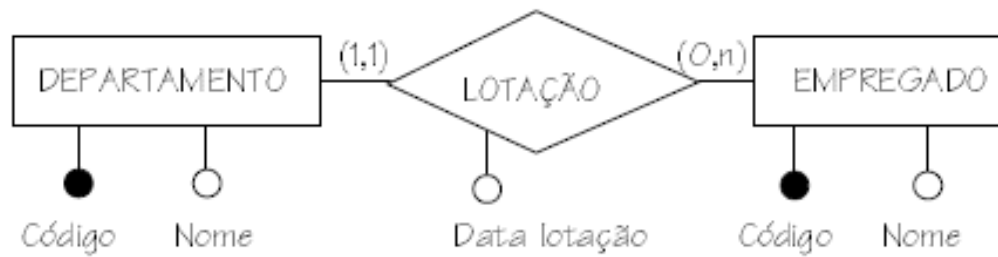
Esquema relacional correspondente:



Introdução

Implementação de relacionamentos – Estratégias:

- Adição de coluna



Esquema relacional correspondente:

Departamento (CodDept, Nome)

Empregado (CodEmp, Nome, CodDept, DataLota)

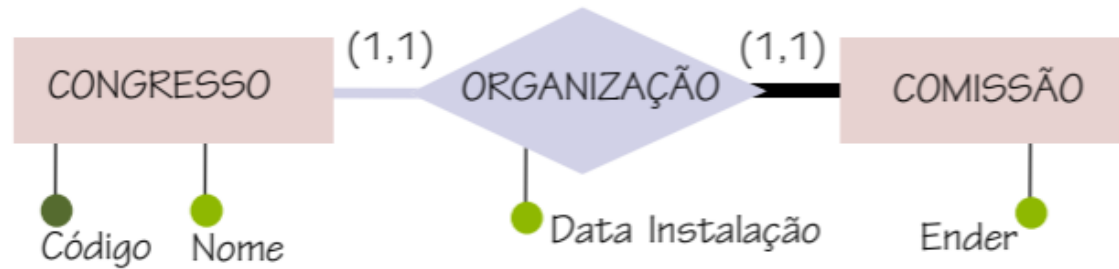
CodDept referencia Departamento

Introdução

Implementação de relacionamentos – Estratégias:

- Fusão de tabelas

Diagrama ER:



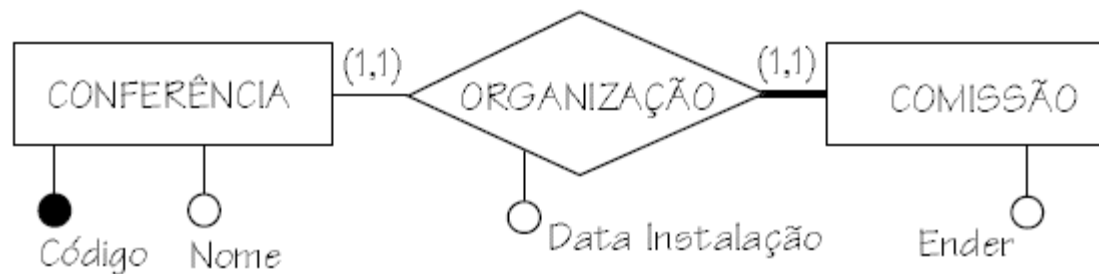
Esquema relacional correspondente:

congresso
cod_congresso
nome
data_instala_comissao_org
endereco_comissao_org

Introdução

Implementação de relacionamentos – Estratégias:

- Fusão de tabelas



Esquema relacional correspondente:

Conferência (CodConf, Nome, DataInstComOrg, EnderComOrg)

Introdução

Implementação de relacionamentos

Tipo de relacionamento	Cardinalidades		
	Tabela própria	Adição coluna	Fusão tabelas
Relacionamentos 1:1			
$(0,1) - \diamond - (0,1)$	±	✓	✗
$(0,1) - \diamond - (1,1)$	∓	±	✓
$(1,1) - \diamond - (1,1)$	∓	∓	✓
Relacionamentos 1:n			
$(0,1) - \diamond - (0,n)$	±	✓	✗
$(0,1) - \diamond - (1,n)$	±	✓	✗
$(1,1) - \diamond - (0,n)$	∓	✓	✗
$(1,1) - \diamond - (1,n)$	∓	✓	✗
Relacionamentos n:n			
$(0,n) - \diamond - (0,n)$	✓	✗	✗
$(0,n) - \diamond - (1,n)$	✓	✗	✗
$(1,n) - \diamond - (1,n)$	✓	✗	✗

✓: Alternativa preferida

±: Pode ser usada, primeira opção

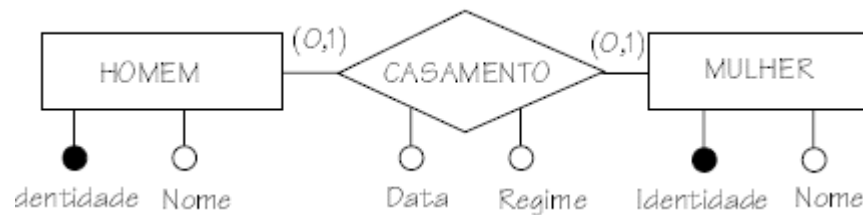
∓: Pode ser usada, segunda opção

✗: Não cabe como solução

Introdução

Implementação de relacionamentos

- Relacionamento 1:1



- Adição de coluna

Mulher (IdentM, Nome, IdentH, Data, Regime)
IdentH referencia Homem
Homem (IdentH, Nome)

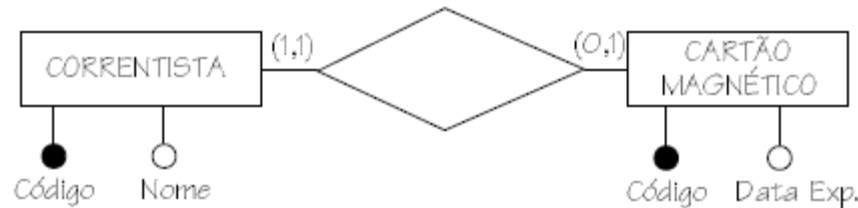
- Tabela Própria

Mulher (IdentM, Nome)
Homem (IdentH, Nome)
Casamento (IdentM, IdentH, Data, Regime)
IdentM referencia Mulher
IdentH referencia Homem

Introdução

Implementação de relacionamentos

- Relacionamento 1:1



- Fusão de Tabela

Esquema relacional correspondente:

Correntista (CodCorrent, Nome, CodCartão, DataExp)

- Adição de coluna

Correntista (CodCorrent, Nome)

Cartão(CodCartão, DataExp, CodCorrent)

CodCorrent referencia Correntista

Introdução

Implementação de relacionamentos

- Relacionamento 1:1



- Fusão de Tabela

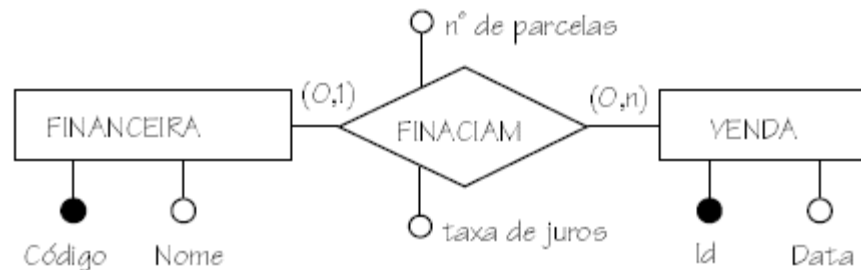
Esquema relacional correspondente:

Conferência (CodConf, Nome, DataInstComOrg, EnderComOrg)

Introdução

Implementação de relacionamentos

- Relacionamento 1:n



- Adição de coluna

Financeira (CodFin, Nome)
Venda (IdVend, Data, CodFin, NoParc, TxJuros)
CodFin referencia Financeira

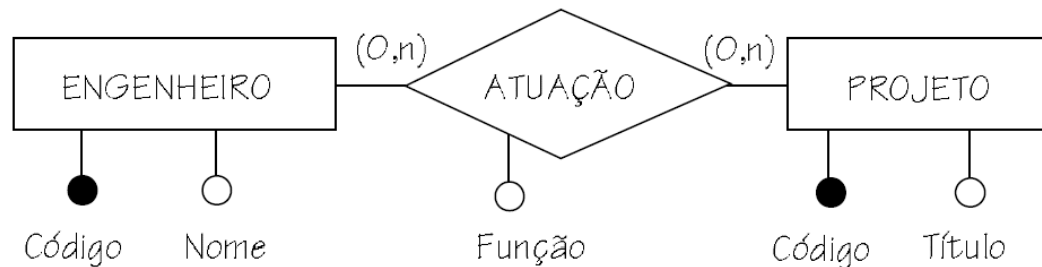
- Tabela própria

Financeira (CodFin, Nome)
Venda (IdVend, Data)
Fianciam (IdVend, CodFin, NoParc, TxJuros)
IdVend referencia Venda
CodFin referencia Financeira

Introdução

Implementação de relacionamentos

- Relacionamento n:n



Esquema relacional correspondente:

Engenheiro (CodEng, Nome)

Projeto (CodProj, Título)

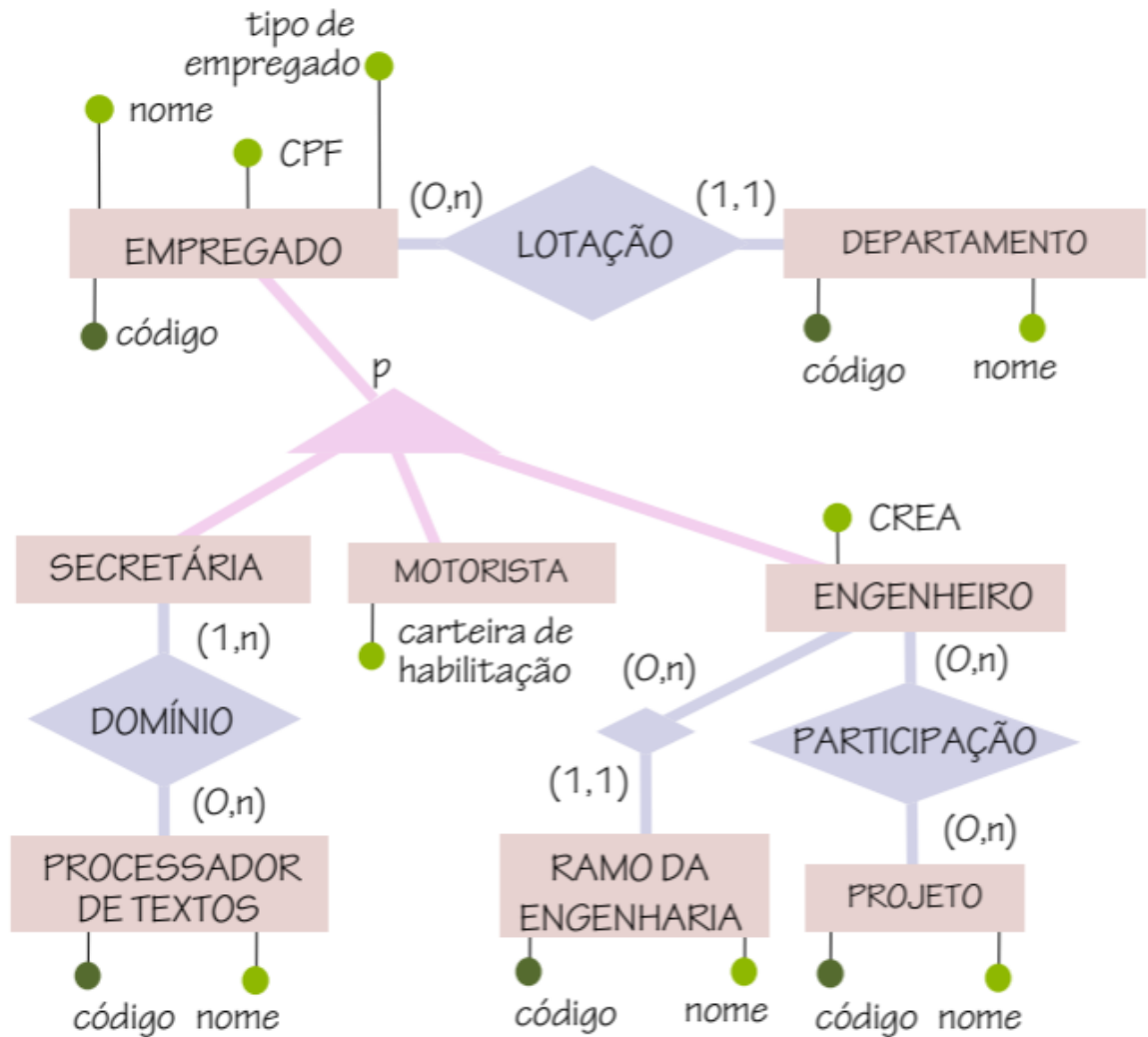
Atuação (CodEng, CopProj, Função)

CodEng referencia Engenheiro

CodProj referencia Projeto

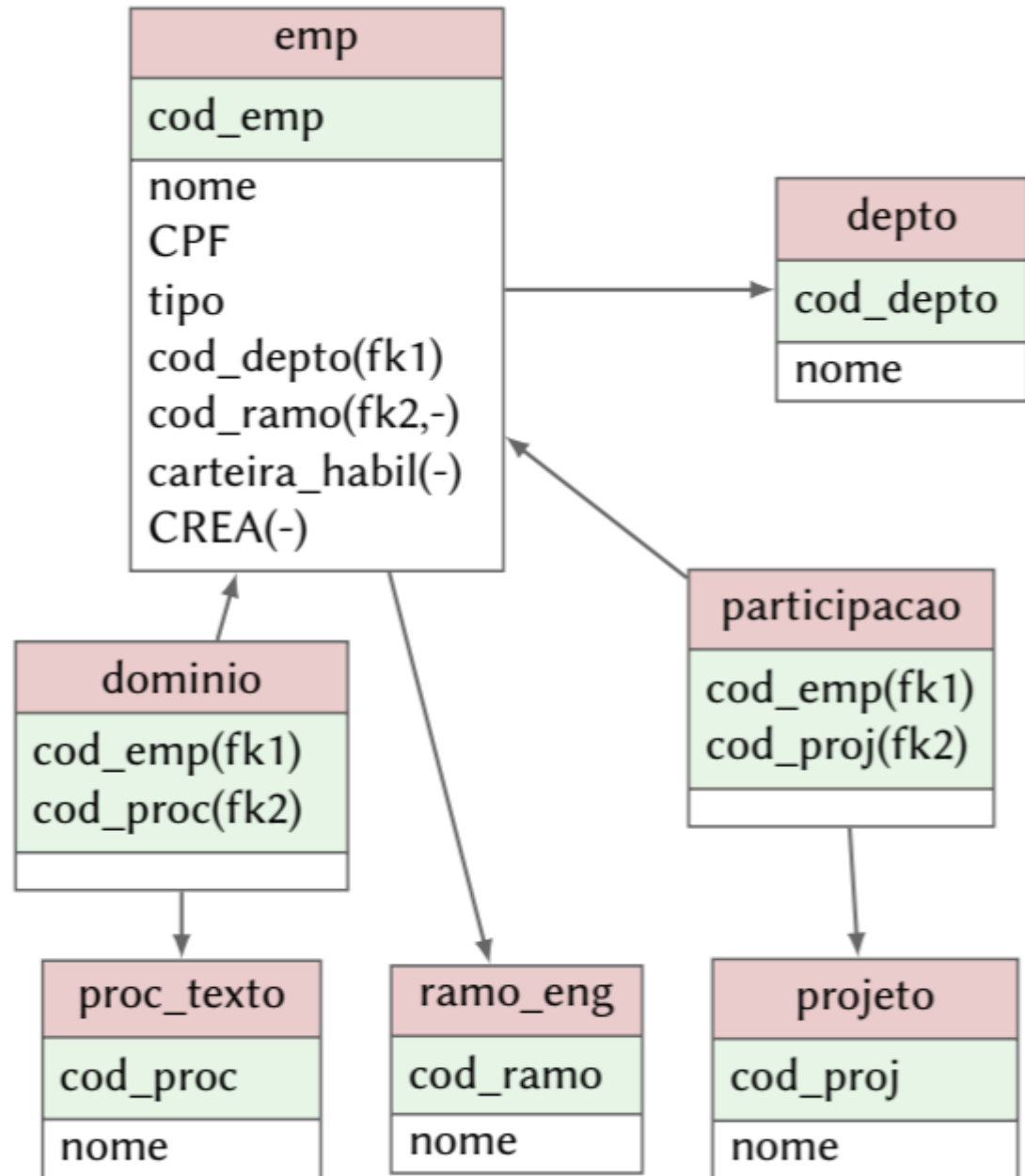
Introdução

Implementação de generalização/especialização:



Introduç

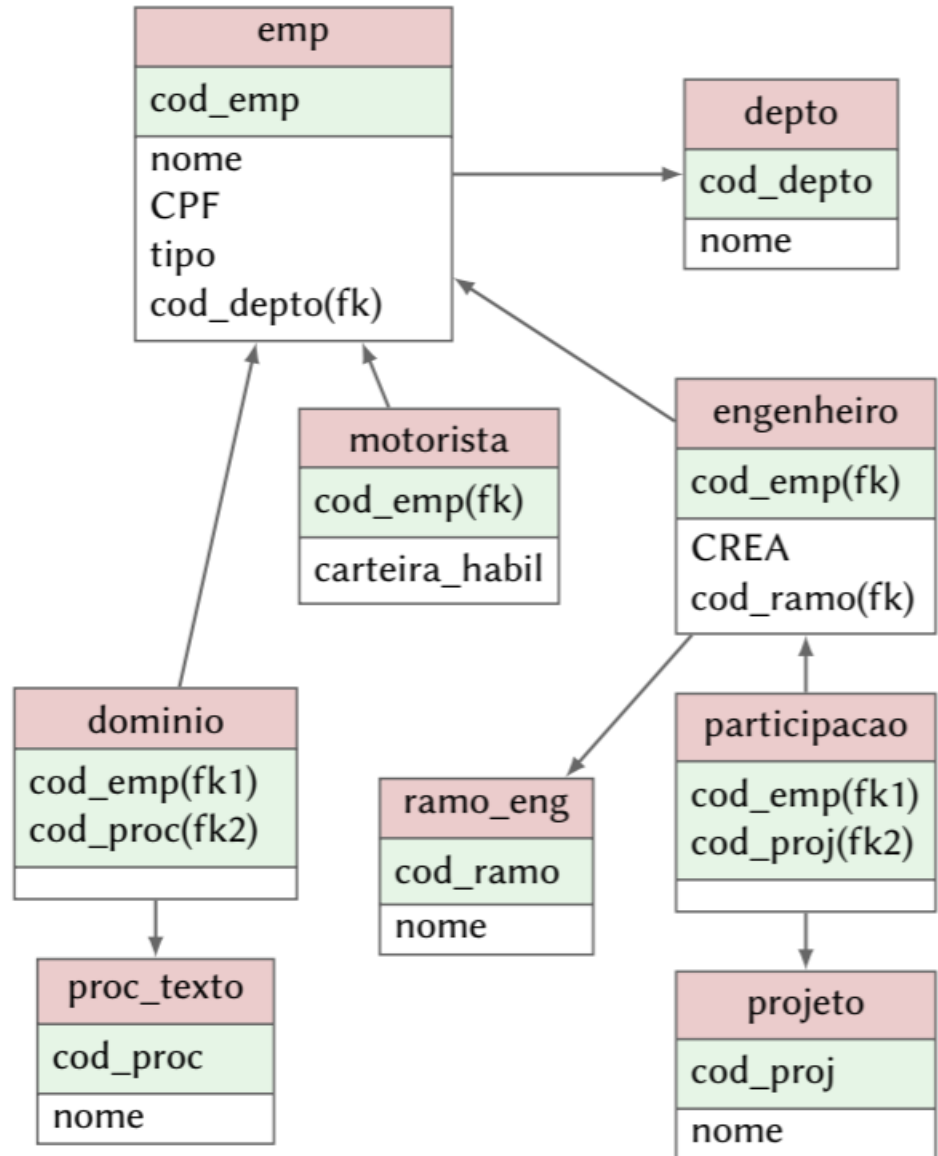
Implementação
de
generalização/
especialização –
Modelo Lógico
(alternativa
tabela única)



Introdução

Implementação
de
generalização/
especialização –

Modelo Lógico
(uma tabela por
entidade da
hierarquia)



Referência

- HEUSER, C. A. , Projeto de banco de Dados. 7 ed., 2024.

