

Transformações entre modelos – Parte 3

CAPÍTULO 5

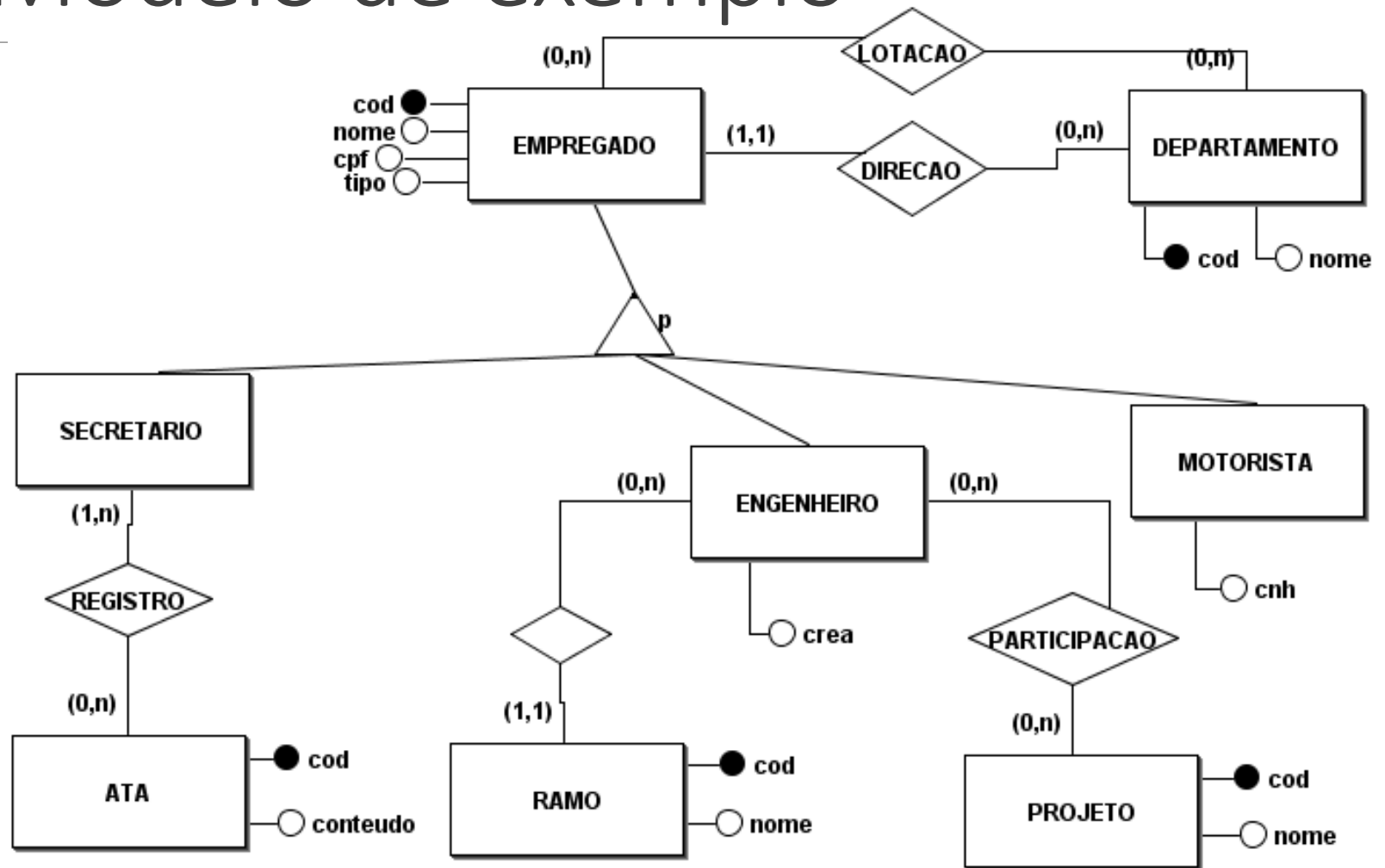
©Carlos A. Heuser - Transparências para uso com o livro Projeto de Banco de Dados,
Ed. Sagra&Luzzatto, Porto Alegre, 1999

Implementação de generalização/especialização

Implementação de generalização/especialização

- Duas alternativas básicas
 - uso de uma tabela para cada entidade
 - uso de uma única tabela para toda hierarquia
- Outra alternativa (exótica)
 - Subdivisão de entidade genérica

Modelo de exemplo



Uma tabela por hierarquia

- Todas tabelas referentes às especializações são fundidas em uma única tabela
- Tabela contém:
 - Chave primária correspondente ao identificador da entidade mais genérica
 - Caso não exista, adicionar uma coluna Tipo
 - Uma coluna para cada atributo da entidade genérica
 - Colunas referentes aos relacionamentos dos quais participa a entidade genérica e que sejam implementados através da alternativa de adicionar colunas à tabela da entidade genérica

segue

Uma tabela por hierarquia

- Tabela contém (continuação):
 - Uma coluna para cada atributo de cada entidade especializada (opcional)
 - Colunas referentes aos relacionamentos dos quais participa cada entidade especializada e que sejam implementados através da alternativa de adicionar colunas à tabela da entidade (campo opcional)

Uma tabela por hierarquia

Emp (CodEmp,tipo,nome,cpf,codDept, **cartHabil,crea,codRamo**)

codDept referencia Depto

codRamo referencia Ramo

Depto (codDept, nome, codDiretor)

codiretor referencia Emp

Ramo (codRamo, nome)

Ata (codAta,conteudo)

Registro (codEmp,codAta)

codEmp referencia Emp

codAta referencia Ata

Projeto (codProj,Nome)

Participação (codEmp,codProj)

codEmp referencia Emp

codProj referencia Projeto

Uma tabela por entidade

- Criar uma tabela para cada entidade que compõe a hierarquia
- Incluir a chave primária da tabela correspondente à entidade genérica, em cada tabela correspondente a uma entidade especializada

Uma tabela por entidade

Emp (codEmp, tipo, nome, cpf, codDept)

codDept referencia Depto

Motorista(códigoEmp, cartHabil)

codEmp referencia Emp

Engenheiro(codEmp, crea, codRamo)

codEmp referencia Emp

codRamo referencia Ramo

Secretario(codEmp)

codEmp referencia Emp

Depto (codDept, nome, codDiretor)

codDiretor referencia Emp

Ramo (codRamo, nome)

Ata (codAta, conteudo)

Registro(codEmp, codAta)

codEmp referencia Secretario

codAta referencia Ata

Projeto (codProj, nome)

Participação (codEmp, codProj)

codEmp referencia Engenheiro

codProj referencia Projeto

Vantagens da implementação com tabela única

- Dados referentes à entidade genérica + dados referentes às especializações
 - em uma única linha
- Minimiza junções
- Menor número de chaves

Vantagens da implementação com uma tabela por entidade

- Colunas opcionais
 - apenas aquelas referentes a atributos que podem ser vazios do ponto de vista da aplicação.
- Restringe melhor os relacionamentos dos quais só podem participar instâncias das entidades especializadas

Subdivisão da entidade genérica

- Uma tabela para cada entidade especializada que não possua outra especialização (entidade folha da árvore)
- Tabela contém
 - dados da entidade especializada +
 - dados da entidade genérica

Subdivisão da entidade genérica

EmpOutros (codEmp, tipo, nome, cpf, codDept)

codDept referencia Depto

Engenheiro(codEmp, nome, cpf, codDept, crea, codRamo)

codDept referencia Depto

codRamo referencia Ramo

Motorista(codEmp, nome, cpf, codDept, cartHabil)

codDept referencia Depto

Secretario(codEmp, nome, cpf, codDept)

codDept referencia Depto

Depto (codDept, nome, codDiretor)

codiretor referencia ????

Ramo (codRamo, nome)

Ata (codAta, conteudo)

Registro (codEmp, codAta)

codEmp referencia Secretario

codAta referencia Ata

Projeto (codProj, nome)

Participação (codEmp, codProj)

codEmp referencia Engenheiro

codProj referencia Projeto

Subdivisão da entidade genérica

○ Vantagem

- Mais eficiente para acessos a alguma entidade especializada
 - Menos colunas do que usar a tabela única
 - Menos junções do que usar uma tabela por entidade

○ Desvantagens:

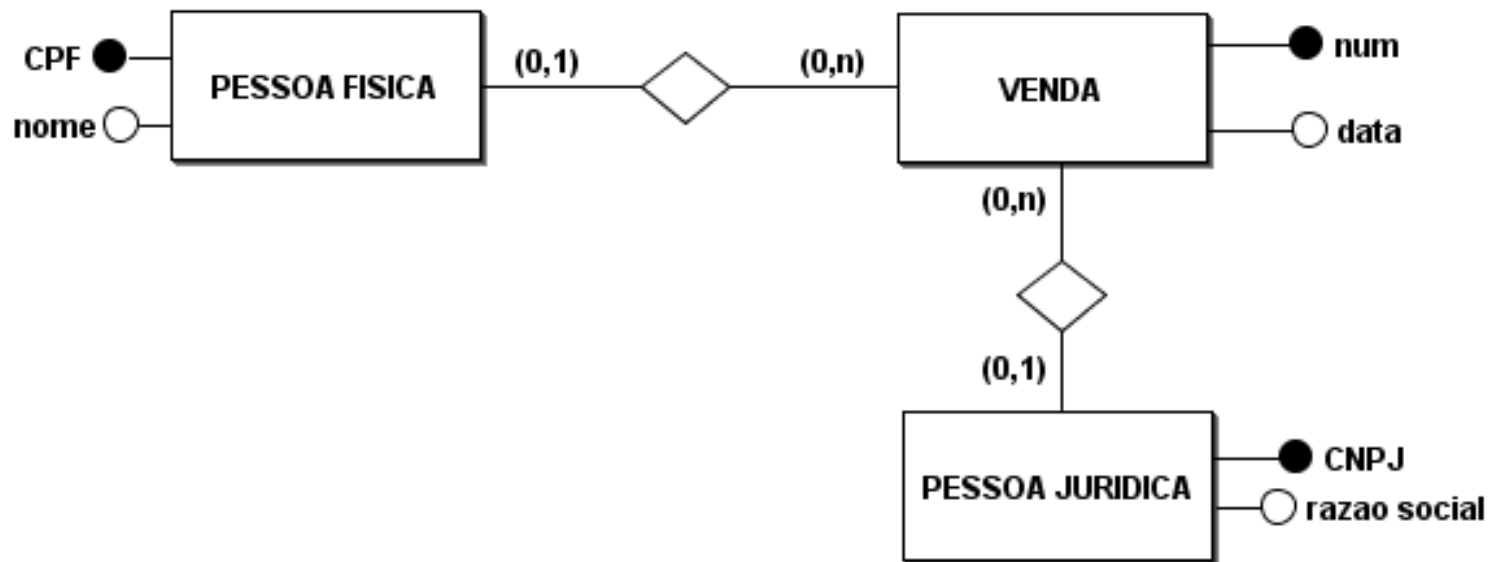
- Unicidade da chave primária
 - não é garantida pelo SGBD
 - deve ser garantida pela aplicação
- Não há como especificar ao SGBD restrições de integridade referenciais, que façam referência ao conjunto de empregados como um todo
- Mais esforço para manutenção do esquema em caso de modificação de coluna

Refinamento do Modelo Relacional

Relacionamentos mutuamente exclusivos

- Algumas vezes
 - esquema de BD criado através do uso das regras de transformação não atende plenamente os requisitos impostos ao sistema
- Exemplos
 - relacionamentos mutuamente exclusivos
 - Simulação de atributos multivalorados
- Necessário buscar alternativa ou alterar o modelo
 - Alternativas somente devem ser tentadas em último caso
 - Do ponto de vista da programação são sempre piores

Relacionamentos mutuamente exclusivos



Relacionamentos mutuamente exclusivos

- Implementação pelas regras

PessFis(CPF,Nome)

PessJur(CNPJ,RazSoc)

Venda(No,data,CPF,CNPJ)

CPF referencia PessFis

CNPJ referencia PessJur

- colunas CPF e CNPJ em Venda são especificadas como opcionais

Relacionamentos mutuamente exclusivos

- Implementação alternativa

- criar uma única coluna na qual aparece o CPF ou o CNPJ do comprador

PessFis(CPF, Nome)

PessJur(CNPJ, RazSoc)

Venda(No, data, CPF_CNPJ, TipoCompr)

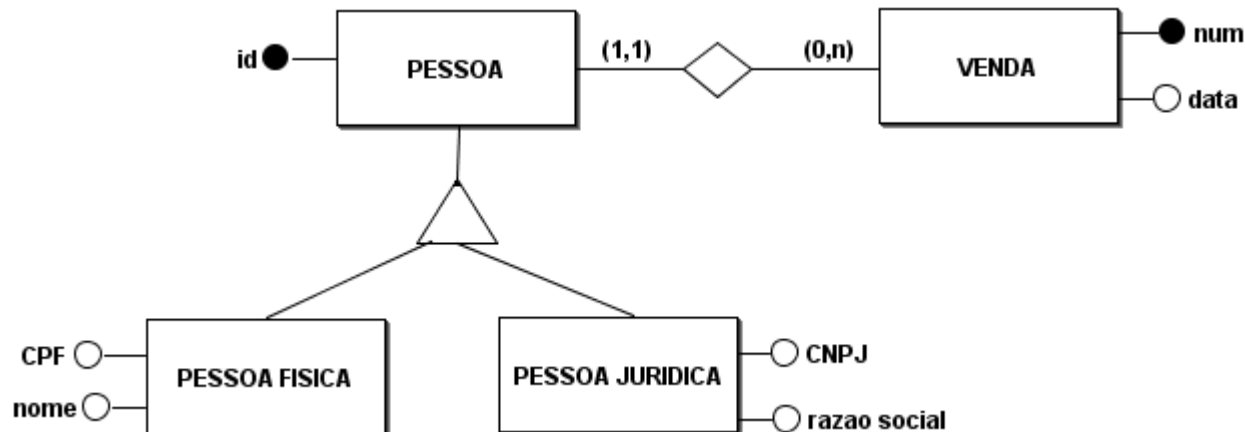
CPF_CNPJ referencia ???

- Desvantagem

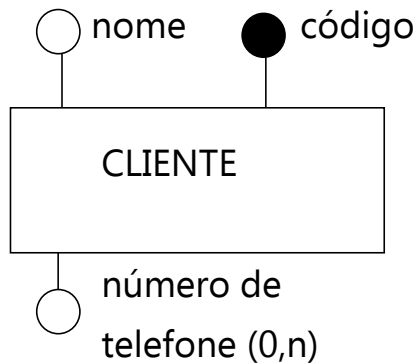
- Não é possível especificar ao SGBD que o campo CPF/CNPJ é chave estrangeira
 - O campo não referencia uma única tabela

Relacionamentos mutuamente exclusivos

- Em alguns casos, o melhor é partir de um modelo conceitual diferente
- No exemplo em questão, um modelo diferente recorreria à generalização/especialização
 - Problema: deve-se mudar a forma de identificar instâncias de pessoa



Tratamento de atributos multivalorados



Cliente (CodCli, Nome, telefone???)

Atributos multivalorados alternativa

- Condições de contorno:
 - Raros clientes possuem mais que dois telefones.
 - Quando isso ocorrer
 - é suficiente armazenarmos apenas dois números.
- Não há consultas ao banco de dados usando o número de telefone como critério de seleção
 - Números de telefone são apenas exibidos ou impressos juntos às demais informações de cliente

Simulação de atributos multivalorados

○ Alternativa 1

- Simular uma coluna multivalorada através da criação de diversas colunas **NumTel** sufixadas por um número

Cliente (CodCli,Nome,NumTel1,NumTel2)

○ Alternativa 2

- Simular uma coluna multivalorada através da criação de uma única coluna que use um delimitador para separar os valores de telefone

Cliente (CodCli,Nome,telefones)

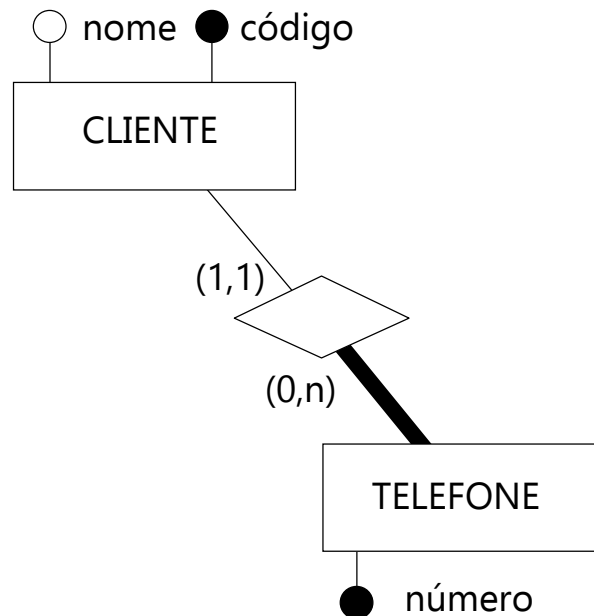
Simulação de atributos multivalorados

- Inconvenientes

- Não se consegue armazenar um número ilimitado de telefones
- Consulta usando o número de telefone como critério de busca torna-se mais complicada

Tratamento de atributos multivalorados

- Em alguns casos, o melhor é partir de um modelo conceitual diferente
- No exemplo em questão, um modelo diferente recorreria a uma entidade separada para telefone
 - Problema: necessário junção para combinar o cliente com os seus telefones



Atividade Individual

Gere o modelo relacional correspondente ao modelo conceitual do próximo slide

