

**Prof. Luiz Antonio**

**Disciplina: Banco de Dados I**

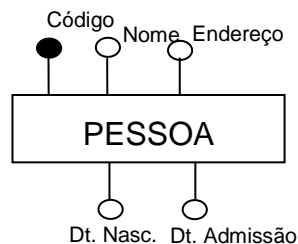
## **Aula 4: Transformação de ER para Relacional**

Após conhecer o modelo conceitual através do Modelo Entidade/Relacionamento e de conhecer o projeto lógico através do Modelo Relacional, iremos conhecer as regras de transformação do modelo ER para o Relacional, ou seja, de modelo conceitual para modelo físico. As regras para a transformação têm em vistas dois objetivos:

- Obter um banco de dados que permita boa performance de instruções de consulta e alterações de dados. Boa performance, a grosso modo, significa minimizar o acesso a disco, que consome o maior tempo em uma instrução.
- Obter um banco de dados que simplifique o desenvolvimento e manutenção de aplicações.

### **1. IMPLEMENTAÇÃO INICIAL DE ENTIDADES**

Esse passo é razoavelmente óbvio: cada entidade é traduzida para uma tabela. Neste processo, cada atributo da entidade define uma coluna desta tabela. Os atributos identificadores da entidade correspondem às colunas que compõem a chave primária da tabela.



***Pessoa(CodPessoa, Nome, Endereco, DtNasc, DtAdm)***

#### **1.1) Nomes de atributos e nomes de colunas:**

Quanto a transformação do nome dos atributos para nome de colunas, é recomendado usar nomes de colunas curtos. Seguindo o padrão dos SGBD's relacionais, os nomes das colunas não podem conter espaços em branco.

Também é usual sufixar o nome da chave primária com o nome da tabela, já que é bem provável que a mesma seja usada em uma chave estrangeira de outra tabela. Isso permite a fácil identificação da referência na chave estrangeira.

É importante salientar que este é um processo inicial e não definitivo. Nas próximas definições veremos que nem todas as entidades serão tabelas e que nem todas tabelas são uma entidade.

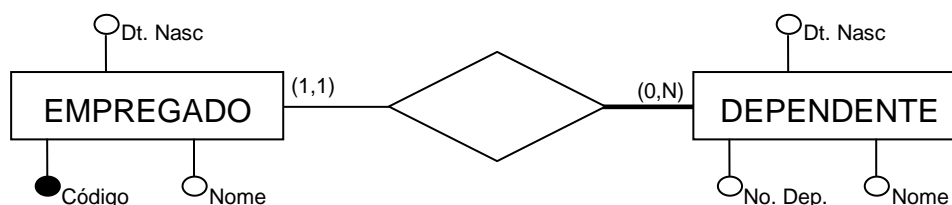
Prof. Luiz Antonio

Disciplina: Banco de Dados I

Aula 4: Transformação de ER para Relacional

### 1.2) Relacionamento Identificador

A situação do relacionamento identificador ocorre quando a ENTIDADE FRACA é identificada pela chave primária da entidade forte.



**Dependente**(CodEmp, NoDep, Nome, DtNasc)

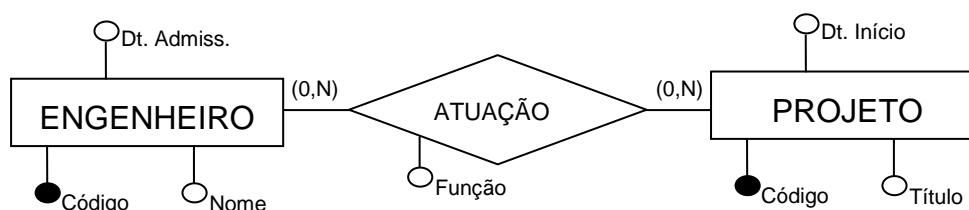
A regra é: para cada identificador externo deve ser criada uma coluna (ou várias, quando o identificador externo for composto) na tabela em questão, coluna esta que fará parte da chave primária da tabela.

## 2. IMPLEMENTAÇÃO DE RELACIONAMENTOS

O fator determinante para a tradução dos relacionamentos é cardinalidade das entidades que participam dele. As **três** formas básicas de tradução de relacionamentos para o modelo relacional são:

### 2.1) Tabela Própria:

Este tipo de implementação é vastamente usada em relacionamentos N:N. Nesta tradução, o relacionamento é implementado através de uma tabela própria, contendo os atributos identificadores das entidades relacionadas e os possíveis **atributos do relacionamento**:



**Engenheiro**(CodEng, Nome, DtAdm)

**Projeto**(CodProj, Título, DtIni)

**Atuação**(CodEnd, CodProj, Funcao)

CodEng referencia Engenheiro

CodProj referencia Projeto

Prof. Luiz Antonio

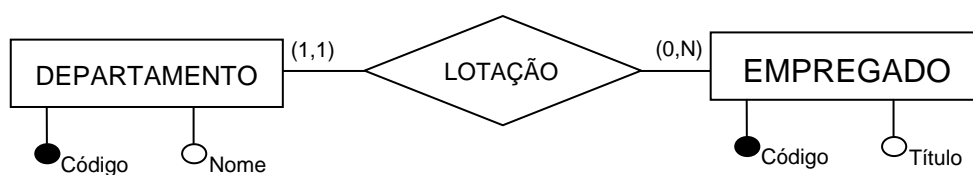
Disciplina: Banco de Dados I

## Aula 4: Transformação de ER para Relacional

### 2.2) Colunas adicionais dentro de uma tabela de entidade:

Outra forma de implementar um relacionamento é a inserindo colunas em uma tabela correspondente a uma das entidades. Esse tipo de implementação só é possível quando uma das entidades que participa do relacionamento tem cardinalidade máxima UM (no exemplo abaixo, a entidade EMPREGADO). Neste caso a tabela da entidade com cardinalidade máxima UM recebe as seguintes colunas:

- Coluna correspondente ao *atributo identificador* da entidade relacionada
- Colunas correspondentes a *atributos do relacionamento* (se houverem)



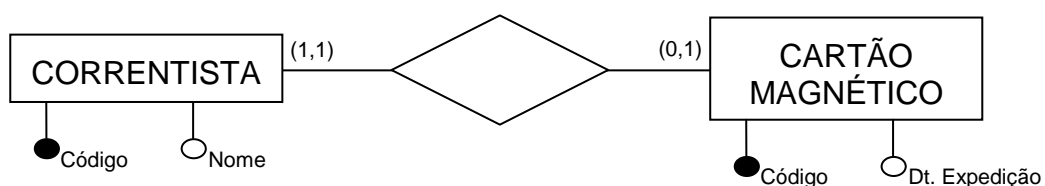
**Departamento**(CodDept, Nome)

**Empregado**(CodEmp, Nome, CodDept)

*CodDept referencia Departamento*

### 2.3) Fusão de entidades em uma tabela:

Esta forma de tradução do modelo ER para o modelo Relacional (lógico) só é possível em relacionamentos 1:1. A fusão entre entidades consiste em implementar todos os atributos de ambas entidades e do relacionamento, se houver, em uma única tabela.



**Correntista**(CodCorr, Nome, CodCartao, DtExpCartao)

Prof. Luiz Antonio

Disciplina: Banco de Dados I

Aula 4: Transformação de ER para Relacional

#### 2.4) Regras para Implementação de Relacionamentos

Como no modelo entidade/relacionamento, no modelo relacional as situações podem não ter uma solução exata. Ao definir um modelo físico, é importante se atentar se o mesmo se adequa às regras de boa performance e facilidade na implementação e, principalmente se ele é adequado a situação do cliente para quem o sistema está sendo modelado.

Com base nisso, a tabela abaixo mostra as possíveis situações de transformações que podemos nos deparar, baseado nas 3 formas vistas acima:

Tipo de relacionamento	Regra de implementação		
	Tabela própria	Adição coluna	Fusão tabelas
<b>Relacionamentos 1:1</b>			
$(0,1) \diamond (0,1)$	±	✓	×
$(0,1) \diamond (1,1)$	×	±	✓
$(1,1) \diamond (1,1)$	×	±	✓
<b>Relacionamentos 1:n</b>			
$(0,1) \diamond (0,n)$	±	✓	×
$(0,1) \diamond (1,n)$	±	✓	×
$(1,1) \diamond (0,n)$	×	✓	×
$(1,1) \diamond (1,n)$	×	✓	×
<b>Relacionamentos n:n</b>			
$(0,n) \diamond (0,n)$	✓	×	×
$(0,n) \diamond (1,n)$	✓	×	×
$(1,n) \diamond (1,n)$	✓	×	×

✓ Alternativa preferida    ± Pode ser usada    × Não usar

Prof. Luiz Antonio

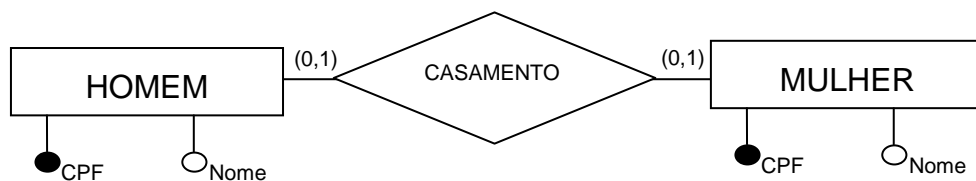
Disciplina: Banco de Dados I

Aula 4: Transformação de ER para Relacional

### 2.4)1. Exemplos

- Relacionamentos 1:1

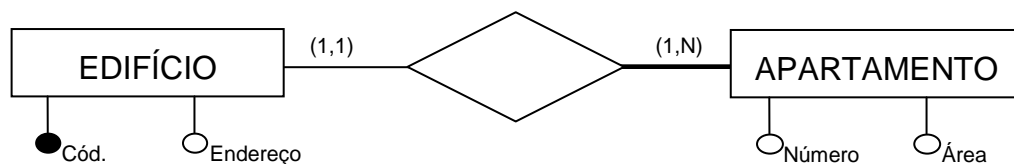
Neste exemplo de 1:1, onde ambas entidades tem participação opcional (cardinalidade mínima 0), segundo a tabela vista anteriormente, a transformação é feita através da adição de colunas. Como a cardinalidade de ambas tabelas são iguais e não há entidade fraca, a adição das colunas pode ser em qualquer uma das tabelas:



**Mulher**(CpfMul, Nome, CpfHom)  
 CpfHom referencia Homem  
**Homem**(CpfHom, Nome)

- Relacionamentos 1:N

Um típico exemplo de 1:N ocorre quando temos uma entidade fraca, com cardinalidade mínima 1 (associação obrigatória). Neste caso usamos a adição de coluna (atributo identificador da entidade forte) na tabela da entidade fraca, com o conceito de relacionamento identificador:



**Edifício**(CodEd, Endereco)  
**Apartamento**(CodEd, NoAp, Area)  
 CodEd referencia Edifício

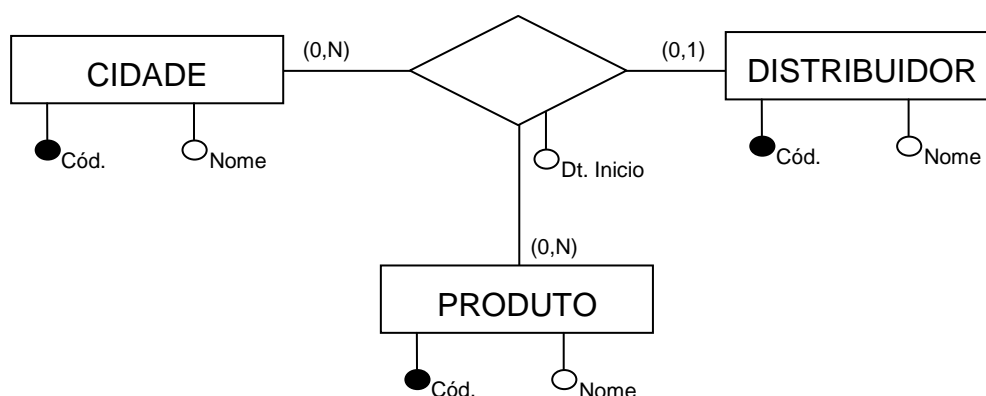
Prof. Luiz Antonio

Disciplina: Banco de Dados I

Aula 4: Transformação de ER para Relacional

### 2.5) Relacionamentos com Grau Maior que 2

Para um relacionamento com grau maior que 2 (relacionamento ternário) não há regras específicas para cada caso. O que ocorre é a simples transformação do relacionamento em uma **entidade fraca**. Esta nova entidade é ligada através de um relacionamento binário a cada entidade participante do relacionamento original:



*Cidade(CodCidade, Nome)*

*Produto(CodProd, Nome)*

*Distribuidor(CodDistr, Nome)*

*Distribuição(CodCidade, CodProd, CodDistr, DtIni)*

*CodCidade referencia Cidade*

*CodProd referencia Produto*

*CodDistr referencia Distribuidor*

### 2.6) Implementação de Generalização/Especialização

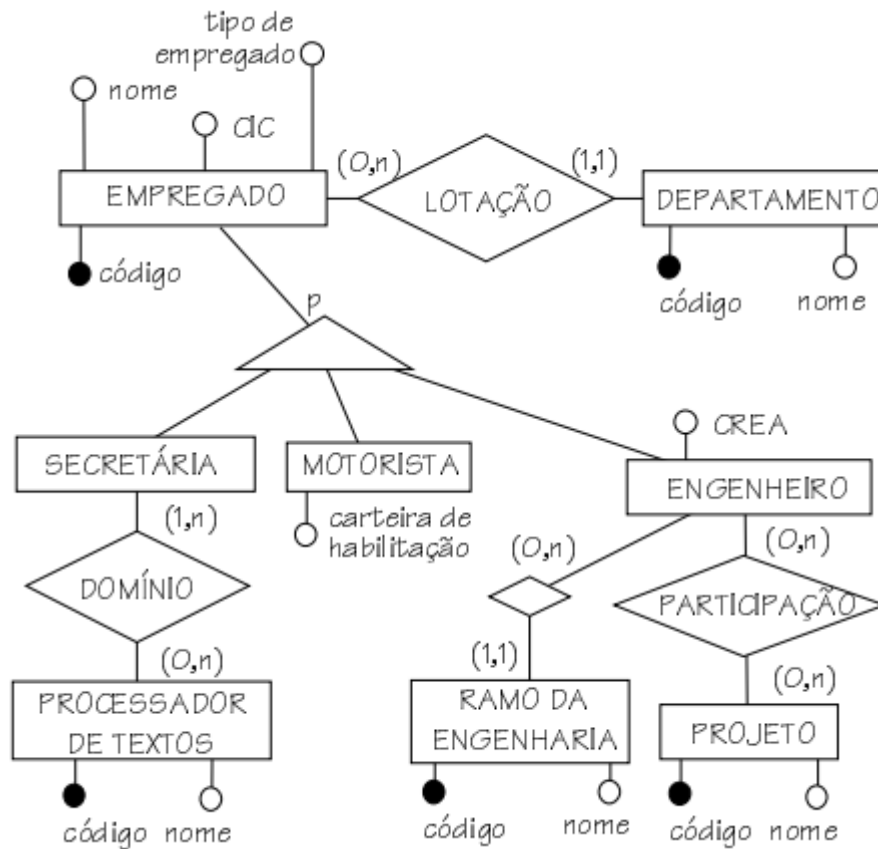
Há duas alternativas para implementação de generalização/especialização:

- **Uso de uma tabela para toda hierarquia**
  - A chave primária será o atributo identificador da entidade genérica
  - Adição de uma coluna TIPO (caso não exista), que representará as entidades especializadas
  - Uma coluna para cada atributo da entidade genérica
  - Uma coluna para cada atributo da entidade especializada (colunas opcionais)
  - Colunas referentes aos relacionamentos dos quais participa cada entidade especializada

Prof. Luiz Antonio

Disciplina: Banco de Dados I

Aula 4: Transformação de ER para Relacional



**Empregado**(CodEmp, Tipo, Nome, CIC, CodDept, CarHabil, CREA, CodRamo)

CodDept referencia Departamento

CodRamo referencia Ramo

**Departamento**(CodDept, Nome)

**Ramo**(CodRamo, Nome)

**ProcessTexto**(CodProc, Nome)

**Domínio**(CodEmp, CodProc)

CodEmp referencia Empregado

CodProc referencia ProcessTexto

**Projeto**(CodProj, Nome)

**Participação**(CodEmp, CodProj)

CodEmp referencia Empregado

CodProj referencia Projeto

**Prof. Luiz Antonio**

**Disciplina: Banco de Dados I**

**Aula 4: Transformação de ER para Relacional**

- **Uso de uma tabela para cada entidade**

Nesta alternativa criamos uma tabela para cada entidade, inclusive as especializadas, aplicando as regras correspondentes à implementação de entidades e relacionamentos já apresentadas nas seções anteriores. A chave primária das entidades especializadas será a chave primária da entidade genérica. Vejamos o exemplo acima usando esta forma:

***Empregado(CodEmp, Tipo, Nome, CIC, CodDept)***

*CodDept referencia Departamento*

*CodRamo referencia Ramo*

***Motorista(CodEmp, CarHabil)***

*CodEmp referencia Empregado*

***Engenheiro (CodEmp, CREA, CodRamo)***

*CodEmp referencia Emp*

*CodRamo referencia Ramo*

***Departamento(CodDept, Nome)***

***Ramo(CodRamo, Nome)***

***ProcessTexto(CodProc, Nome)***

***Domínio(CodEmp, CodProc)***

*CodEmp referencia Empregado*

*CodProc referencia ProcessTexto*

***Projeto(CodProj, Nome)***

***Participação(CodEmp, CodProj)***

*CodEmp referencia Empregado*

*CodProj referencia Projeto*

Note que não foi criada uma entidade específica para secretária. Como ela não possui atributos específicos, não houve esta necessidade. Quanto a secretária foi criada somente a chave estrangeira na tabela domínio, referenciando a tabela empregado.