



Banco de Dados

# Mapeamento ER e ERR para Relacional



FACOM – UFMS

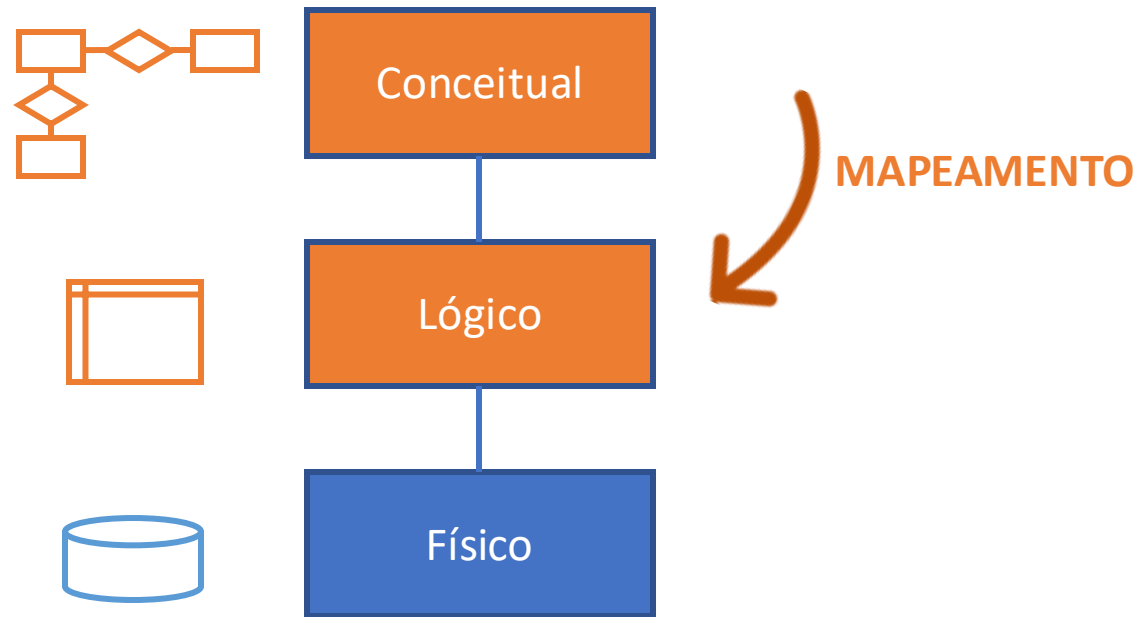
Vanessa Borges

[vanessa@facom.ufms.br](mailto:vanessa@facom.ufms.br)

- Ramez Elmasri e Shamkant B. Navathe 6° Ed (2010)
- Capítulo 9: projeto de banco de dados relacional por mapeamento ER e EER para relacional
- Calor Alberto Heuser, 6° Ed (2009)
- Capítulo 5: transformação entre modelos

# Mapeamento esquema conceitual para lógico

- Descreveremos agora como é realizada a **transformação** de um **diagrama ER** em uma **coleção de tabelas com restrições associadas**, ou seja, um **esquema de um banco de dados relacional**.



# Sumário

- **Algoritmo de mapeamento do ER para o Relacional**

- Passo 1: Mapear Conjuntos de Entidades Regulares
- Passo 2: Mapear Conjuntos de Entidades Fracas
- Passo 3: Mapear Conjuntos de Relacionamento Binário 1:1
- Passo 4: Mapear Conjuntos de Relacionamento Binário Regular 1:N
- Passo 5: Mapear Conjuntos de Relacionamento Binário M:N
- Passo 6: Mapear Atributos Multivalorados ou Compostos
- Passo 7: Mapear Conjuntos de Relacionamentos n-ário,  $n > 2$
- Passo 8: Relacionamento Recursivo



# Sumário

- **Algoritmo de mapeamento do ER para o Relacional**

- Passo 1: Mapear Conjuntos de Entidades Regulares
- Passo 2: Mapear Conjuntos de Entidades Fracas
- Passo 3: Mapear Conjuntos de Relacionamento Binário 1:1
- Passo 4: Mapear Conjuntos de Relacionamento Binário Regular 1:N
- Passo 5: Mapear Conjuntos de Relacionamento Binário M:N
- Passo 6: Mapear Atributos Multivalorados ou Compostos
- Passo 7: Mapear Conjuntos de Relacionamentos n-ário,  $n > 2$
- Passo 8: Relacionamento Recursivo

- **Mapeamento EER para Relações**

- Passo 9: Opções para mapeamento de especialização ou generalização



# Sumário

- **Algoritmo de mapeamento do ER para o Relacional**

- Passo 1: Mapear Conjuntos de Entidades Regulares
- Passo 2: Mapear Conjuntos de Entidades Fracas
- Passo 3: Mapear Conjuntos de Relacionamento Binário 1:1
- Passo 4: Mapear Conjuntos de Relacionamento Binário Regular 1:N
- Passo 5: Mapear Conjuntos de Relacionamento Binário M:N
- Passo 6: Mapear Atributos Multivalorados ou Compostos
- Passo 7: Mapear Conjuntos de Relacionamentos n-ário,  $n > 2$
- Passo 8: Relacionamento Recursivo

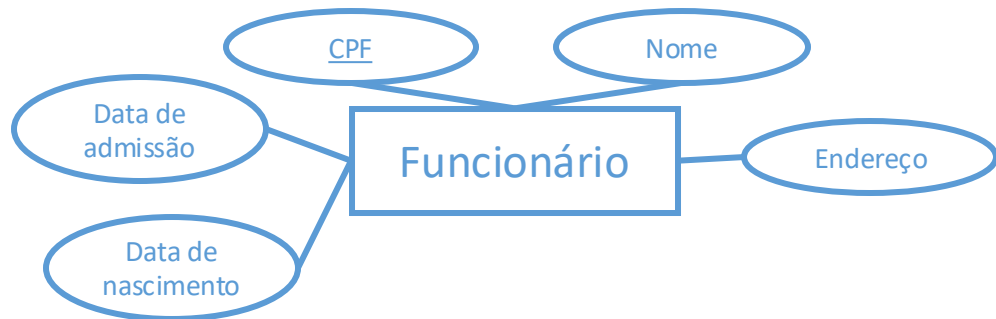
- **Mapeamento EER para Relações**

- Passo 9: Opções para mapeamento de especialização ou generalização



# Passo 1: Mapear Conjuntos de Entidades Regulares

- Para cada entidade forte E no esquema ER, **crie uma relação R que inclua todos os atributos simples de E**
  - Escolha um atributo chave de E como a chave primária de R
  - Se a chave escolhida de E for composta, o conjunto dos atributos simples que a formam deve ser a chave primária de R



Funcionario (CPF, Nome, Endereco, DtNasc, DtAdm)

**FUNCIONARIO**

| <u>CPF</u>  | Nome    | Endereco      | DtNasc     | DtAdm      |
|-------------|---------|---------------|------------|------------|
| 11111111111 | Antônio | Rua das velas | 12/12/1980 | 01/07/2017 |
| 22222222222 | Carlos  | Rua Itaipu    | 01/05/1993 | 08/05/2017 |



# Sumário

- **Algoritmo de mapeamento do ER para o Relacional**

- Passo 1: Mapear Conjuntos de Entidades Regulares
- **Passo 2: Mapear Conjuntos de Entidades Fracas**
- Passo 3: Mapear Conjuntos de Relacionamento Binário 1:1
- Passo 4: Mapear Conjuntos de Relacionamento Binário Regular 1:N
- Passo 5: Mapear Conjuntos de Relacionamento Binário M:N
- Passo 6: Mapear Atributos Multivalorados ou Compostos
- Passo 7: Mapear Conjuntos de Relacionamentos n-ário,  $n > 2$
- Passo 8: Relacionamento Recursivo

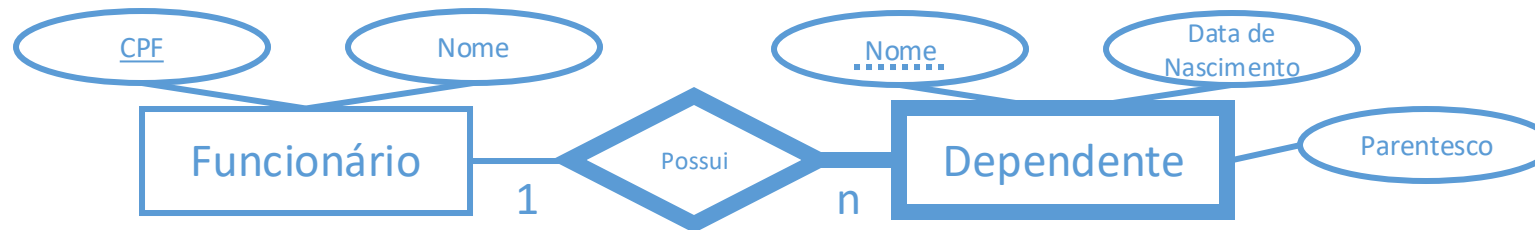
- **Mapeamento EER para Relações**

- Passo 9: Opções para mapeamento de especialização ou generalização



# Passo 2: Mapear Conjuntos de Entidades Fracas

- Entidade fraca traduzida em tabela
  - Atributos da entidade traduzidos em colunas da relação
- Chave estrangeira na tabela/entidade fraca = chave primária da entidade proprietária
- Chave primária da tabela/entidade fraca são os atributos identificadores da entidade fraca + chave primária da entidade proprietária



Funcionario(CPF, Nome)

Dependente(CPF (FK), NomeDep, DtNasc, Parentesco)

| FUNCIONARIO  |         | DEPENDENTE   |                |            |            |
|--------------|---------|--------------|----------------|------------|------------|
| <u>CPF</u>   | Nome    | <u>CPF</u>   | <u>NomeDep</u> | DtNasc     | Parentesco |
| 111111111111 | Antônio | 111111111111 | Ana            | 11/01/1990 | CONJUGE    |
| 222222222222 | Carlos  | 222222222222 | Ana            | 13/07/1998 | CONJUGE    |
| 333333333333 | Renata  | 222222222222 | Rogério        | 05/01/2000 | FILHO      |





# Sumário

- **Algoritmo de mapeamento do ER para o Relacional**

Regras de Implementação

- Passo 1: Mapear Conjuntos de Entidades Regulares
- Passo 2: Mapear Conjuntos de Entidades Fracas
- Passo 3: Mapear Conjuntos de Relacionamento Binário 1:1
- Passo 4: Mapear Conjuntos de Relacionamento Binário Regular 1:N
- Passo 5: Mapear Conjuntos de Relacionamento Binário M:N
- Passo 6: Mapear Atributos Multivalorados ou Compostos
- Passo 7: Mapear Conjuntos de Relacionamentos n-ário,  $n > 2$
- Passo 8: Relacionamento Recursivo

- **Mapeamento EER para Relações**

- Passo 9: Opções para mapeamento de especialização ou generalização



# Regras de Implementação

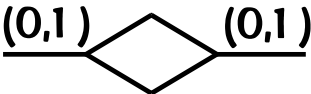
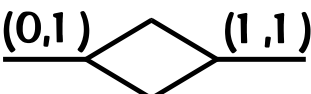
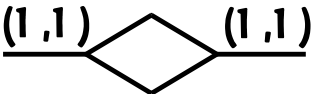
## Objetivos:

- Obter um banco de dados:
  - com boa performance: diminuir o número de acessos a disco
  - que simplifique o desenvolvimento e manutenção de aplicações

## Para alcançar esses objetivos:

- Evitar junções
- Evitar campos opcionais (valores NULL)
- Diminuir o número de chaves

cliente(cod, nome, endereço, complemento)  
OU  
cliente(cod, nome)  
cliend (codcli (FK), endereço, complemento)

| Tipo de Relacionamento   | Regra de Implementação |               |               |
|--|------------------------|---------------|---------------|
|  | Tabela Própria         | Adição Coluna | Fusão Tabelas |
|   | 2                      | ✓             | ✗             |
|   | 3                      | 2             | ✓             |
|  | 3                      | 3             | ✓             |

(Heuser, 2004)

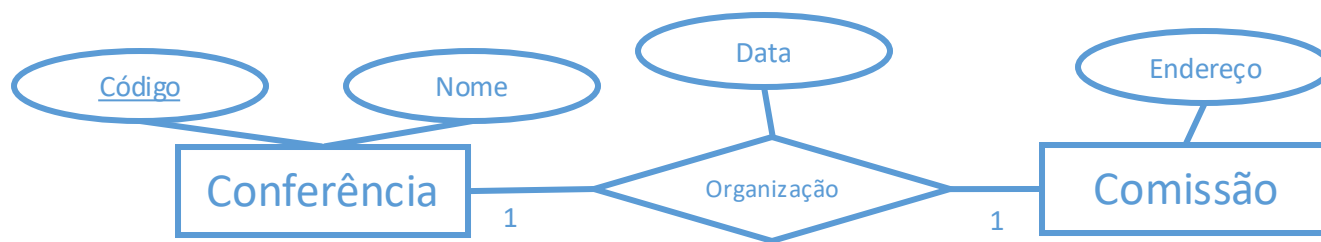


# Regras de Implementação

## Fusão de Tabelas

- **Fusão de tabelas de entidades**

- (1:1) Implementar, em uma única entidade, todos os atributos de ambas as entidades, bem como os atributos eventualmente existentes no relacionamento.



Conferência (CodConferencia, Nome, DtOrganizacao, EndereçoComissao)

CONFERENCIA

| <u>CodConferencia</u> | nome    | DtOrganizacao | EndereçoComissao  |
|-----------------------|---------|---------------|-------------------|
| 1                     | SBIE    | 10/02/11      | Rua Paraná, 1514  |
| 2                     | ACM-SAC | NULL          | Rua Bocaiuva, 111 |
| 3                     | BRACIS  | NULL          | NULL              |

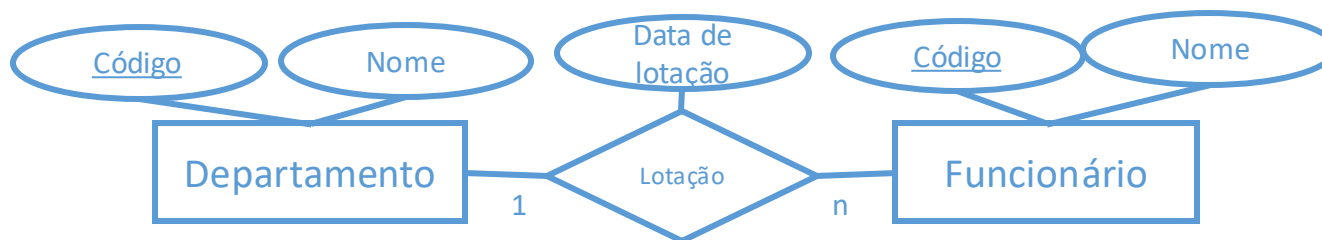


# Regras de Implementação

## Adição de Coluna

- **Adição de colunas**

- Implementação da chave estrangeira



Departamento (CodDepartamento, nome)

Funcionário (CodFuncionario, CodDepartamento (FK), nome, DtLotacao)

DEPARTAMENTO

| <u>CodDepartamento</u> | Nome           |
|------------------------|----------------|
| 1                      | Administrativo |
| 2                      | Financeiro     |
| 3                      | Jurídico       |
| 4                      | Comercial      |

FUNCIONARIO

| <u>CodFuncionario</u> | Nome   | CodDepartamento | DtLotacao |
|-----------------------|--------|-----------------|-----------|
| 1                     | Carlos | 1               | 12/01/17  |
| 2                     | Ana    | 1               | NULL      |
| 3                     | Arthur | 1               | 20/06/18  |
| 4                     | Renato | 2               | NULL      |
| 5                     | Thaís  | 3               | 20/06/19  |

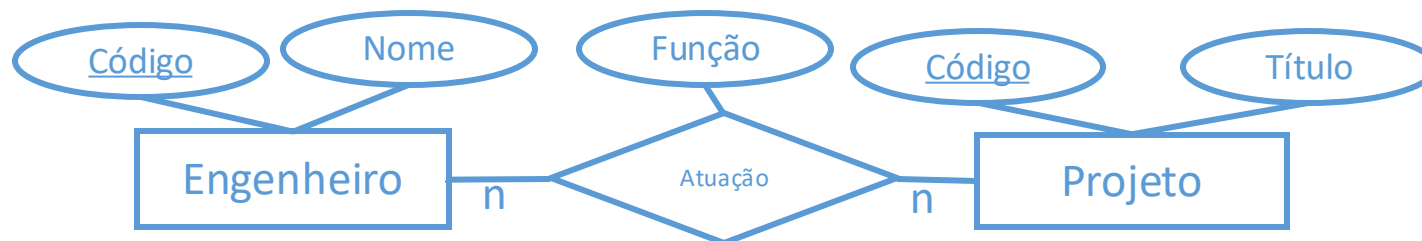


# Regras de Implementação

## Tabela própria

- **Tabela própria**

- O relacionamento é implementado por meio de uma tabela própria que contém as seguintes colunas:
  - Colunas correspondentes aos identificadores das entidades relacionadas
  - Colunas correspondentes aos atributos do relacionamento



Engenheiro (CodEngenheiro, nome)

Projeto (CodProjeto, titulo)

Atuacao (CodEngenheiro (FK), CodProjeto (FK), funcao)

ENGENHEIRO

| <u>CodEngenheiro</u> | nome    |
|----------------------|---------|
| 1                    | Carlos  |
| 2                    | Miguel  |
| 3                    | Augusto |

ATUACAO

| <u>CodEngenheiro</u> | <u>CodProjeto</u> | funcao  |
|----------------------|-------------------|---------|
| 1                    | 1                 | Gerente |
| 1                    | 2                 | NULL    |
| 2                    | 1                 | NULL    |

PROJETO

| <u>CodProjeto</u> | titulo |
|-------------------|--------|
| 1                 | PPS    |
| 2                 | PMP    |
| 3                 | PSI    |

# Sumário

- **Algoritmo de mapeamento do ER para o Relacional**

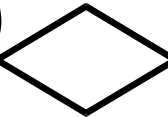
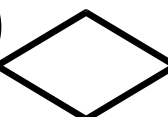
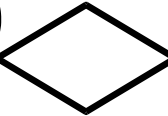
- Passo 1: Mapear Conjuntos de Entidades Regulares
- Passo 2: Mapear Conjuntos de Entidades Fracas
- **Passo 3: Mapear Conjuntos de Relacionamento Binário 1:1**
- Passo 4: Mapear Conjuntos de Relacionamento Binário Regular 1:N
- Passo 5: Mapear Conjuntos de Relacionamento Binário M:N
- Passo 6: Mapear Atributos Multivalorados ou Compostos
- Passo 7: Mapear Conjuntos de Relacionamentos n-ário,  $n > 2$
- Passo 8: Relacionamento Recursivo

- **Mapeamento EER para Relações**

- Passo 9: Opções para mapeamento de especialização ou generalização



# Passo 3: Mapear Conjuntos de Relacionamento Binário 1:1

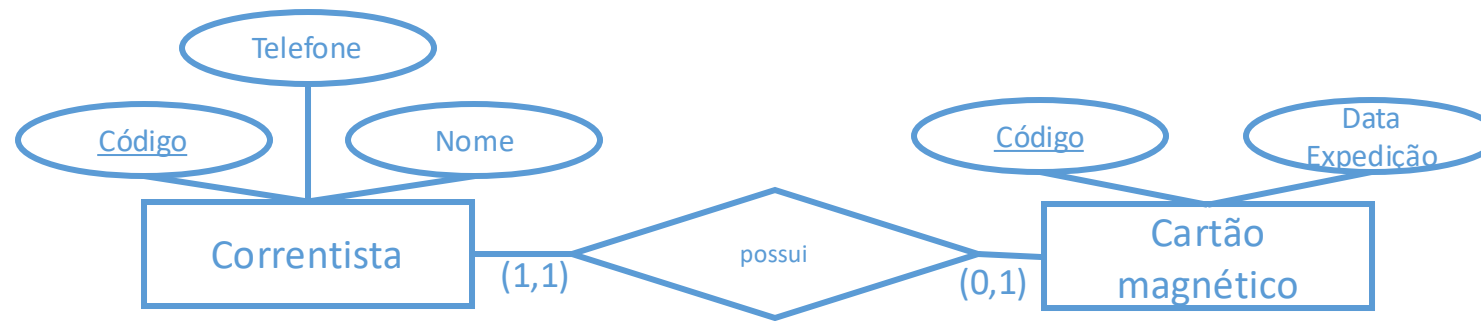
| Tipo de Relacionamento  | Regra de Implementação |               |               |
|---|------------------------|---------------|---------------|
|   | Tabela Própria         | Adição Coluna | Fusão Tabelas |
| <u>(0,1)</u>  <u>(0,1)</u>   | 2                      | ✓             | ✗             |
| <u>(0,1)</u>  <u>(1,1)</u>   | 3                      | 2             | ✓             |
| <u>(1,1)</u>  <u>(1,1)</u> | 3                      | 3             | ✓             |

(Heuser, 2004)



# Passo 3: Mapear Conjuntos de Relacionamento Binário 1:1

- Participação obrigatória e participação opcional


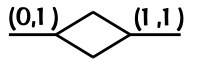



| Tipo de Relacionamento | Regra de Implementação |               |               |
|------------------------|------------------------|---------------|---------------|
|                        | Tabela Própria         | Adição Coluna | Fusão Tabelas |
|                        | 2                      | ✓             | ✗             |
|                        | 3                      | 2             | ✓             |
|                        | 3                      | 3             | ✓             |



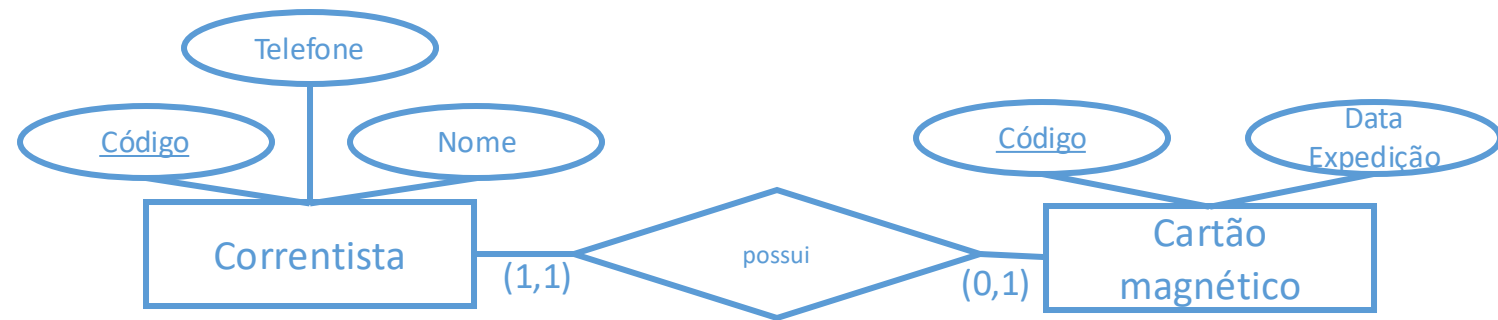
# Passo 3: Mapear Conjuntos de Relacionamento Binário 1:1

- Participação obrigatória e participação opcional

| Tipo de Relacionamento  | Regra de Implementação |               |               |
|---|------------------------|---------------|---------------|
|   | Tabela Própria         | Adição Coluna | Fusão Tabelas |
|  | 2                      | ✓             | ✗             |
|  | 3                      | 2             | ✓             |
|  | 3                      | 3             | ✓             |

Fusão de tabela (relacionamento incorporado):

- Implementa-se em uma única tabela todos os atributos de ambas as entidades, bem como os atributos eventualmente existentes no relacionamento.




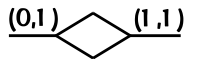

CORRENTISTA

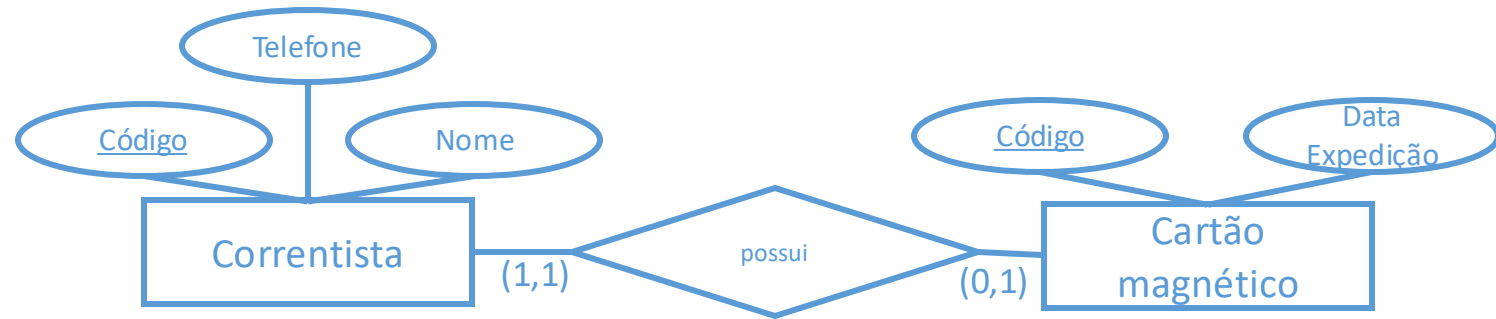
| <u>CodCorrent</u> | Nome       | Telefone | CodCartao | DataExp  |
|-------------------|------------|----------|-----------|----------|
| 1                 | Carlos     | 22343221 | 1         | 12/01/17 |
| 2                 | Ana        | 17453457 | NULL      | NULL     |
| 3                 | Arthur     | 78541257 | 2         | 20/06/17 |
| 4                 | Renato     | 78954214 | NULL      | NULL     |
| 5                 | Francielle | 86473542 | 4         | 15/01/16 |

Correntista (CodCorrent, Nome, Telefone, CodCartao, DataExp)

# Passo 3: Mapear Conjuntos de Relacionamento Binário 1:1

- Participação obrigatória e participação opcional

| Tipo de Relacionamento  | Regra de Implementação |               |               |
|---|------------------------|---------------|---------------|
|   | Tabela Própria         | Adição Coluna | Fusão Tabelas |
|  | 2                      | ✓             | ✗             |
|  | 3                      | 2             | ✓             |
|  | 3                      | 3             | ✓             |



Correntista (CodCorrent, Telefone, Nome)  
Cartao (CodCartao, DataExp, CodCorrent (FK))

2

Adição de coluna (chave estrangeira):

- Chave primária de uma das relações torna-se chave estrangeira da outra

CORRENTISTA

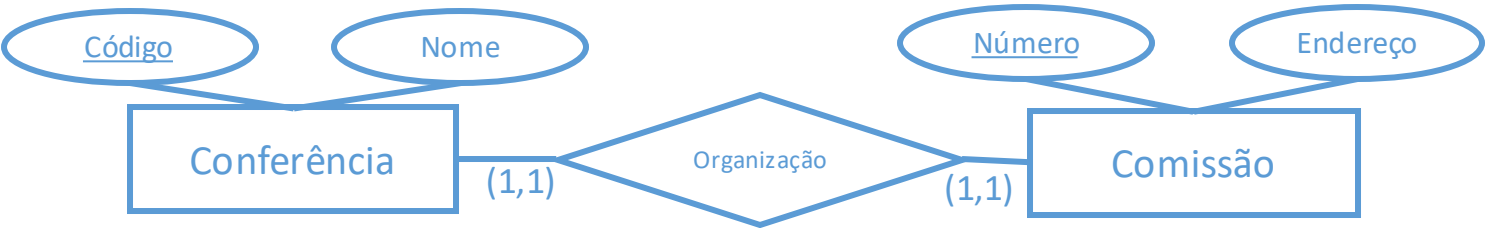
| <u>CodCorrent</u> | Nome       | Telefone |
|-------------------|------------|----------|
| 1                 | Carlos     | 22343221 |
| 2                 | Ana        | 17453457 |
| 3                 | Arthur     | 78541257 |
| 4                 | Renato     | 78954214 |
| 5                 | Francielle | 86473542 |



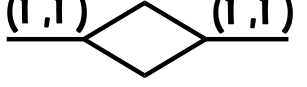
CARTAO

| <u>CodCartao</u> | DataExp  | CodCorrent |
|------------------|----------|------------|
| 1                | 12/01/17 | 1          |
| 2                | 20/06/17 | 3          |
| 4                | 15/01/16 | 5          |

# Passo 3: Mapear Conjuntos de Relacionamento Binário 1:1

- Ambas as entidades com participação total






| Tipo de Relacionamento  | Regra de Implementação |               |               |
|---|------------------------|---------------|---------------|
|   | Tabela Própria         | Adição Coluna | Fusão Tabelas |
|  | 2                      | ✓             | ✗             |
|  | 3                      | 2             | ✓             |
|  | 3                      | 3             | ✓             |



# Passo 3: Mapear Conjuntos de Relacionamento Binário 1:1

- Participação obrigatória e participação opcional

| Tipo de Relacionamento  | Regra de Implementação |               |               |
|---|------------------------|---------------|---------------|
|   | Tabela Própria         | Adição Coluna | Fusão Tabelas |
|  | 2                      | ✓             | ✗             |
|  | 3                      | 2             | ✓             |
|  | 3                      | 3             | ✓             |

Fusão de tabela (relacionamento incorporado):

- Implementa-se em uma única tabela todos os atributos de ambas as entidades, bem como os atributos eventualmente existentes no relacionamento.



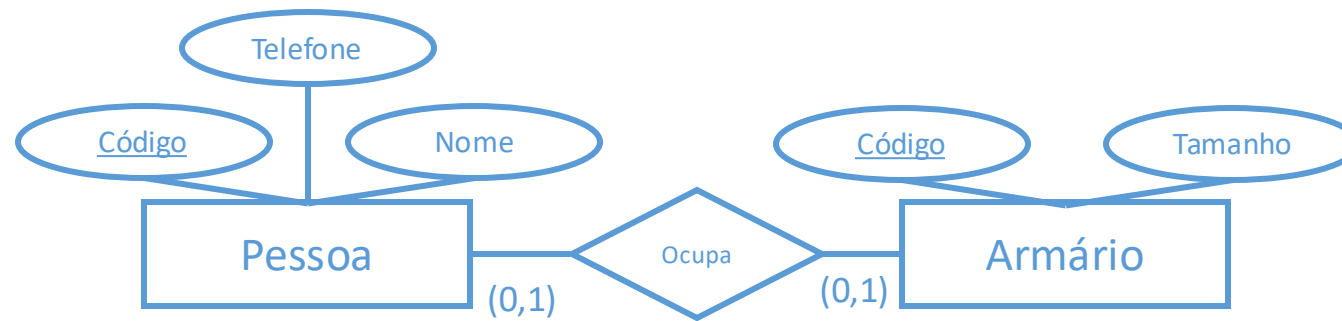
conferencia\_comissao (codigo, nome, numero, endereco)  
OU  
conferencia\_comissao (codigo, nome, numero, endereco)




CONFERENCIA\_COMISSAO

| <u>codigo</u> | numero | nome | endereco |
|---------------|--------|------|----------|
| A             | 1      | RBIE | Rua A    |
| B             | 2      | SBIE | NULL     |
| C             | 10     | FIE  | Rua C    |
| D             | 5      | SAC  | NULL     |

# Passo 3: Mapear Conjuntos de Relacionamento Binário 1:1

- Ambas as entidades com participação opcional



| Tipo de Relacionamento  | Regra de Implementação |               |               |
|---|------------------------|---------------|---------------|
|   | Tabela Própria         | Adição Coluna | Fusão Tabelas |
| <u>(0,1)</u>  <u>(0,1)</u> | 2                      | ✓             | ✗             |
| <u>(0,1)</u>  (1,1)        | 3                      | 2             | ✓             |
| (1,1)  (1,1)               | 3                      | 3             | ✓             |

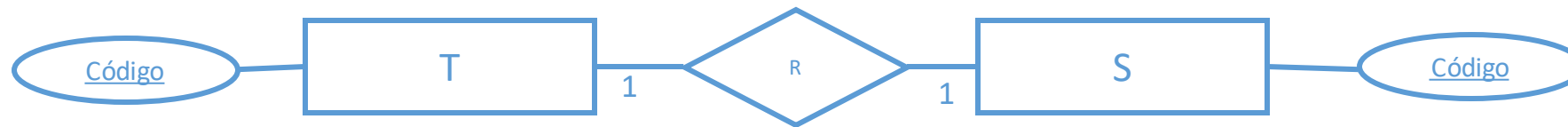


# Passo 3: Mapear Conjuntos de Relacionamento Binário 1:1

## Adição de Coluna

- **Abordagem Adição de Coluna (técnica de chave estrangeira)**

- Escolha uma das relações, por exemplo a relação S e inclua como chave estrangeira em S a chave primária de T;
- Se houver tipo entidade com participação total envolvido no relacionamento, escolhê-lo para o papel de S será a melhor opção;
- Inclua todos os atributos simples (ou componentes simples dos atributos compostos) do tipo relacionamento como atributos de S.



T(codT, ....)

S (codS, codT (FK), <atributos de relacionamento, ...>)



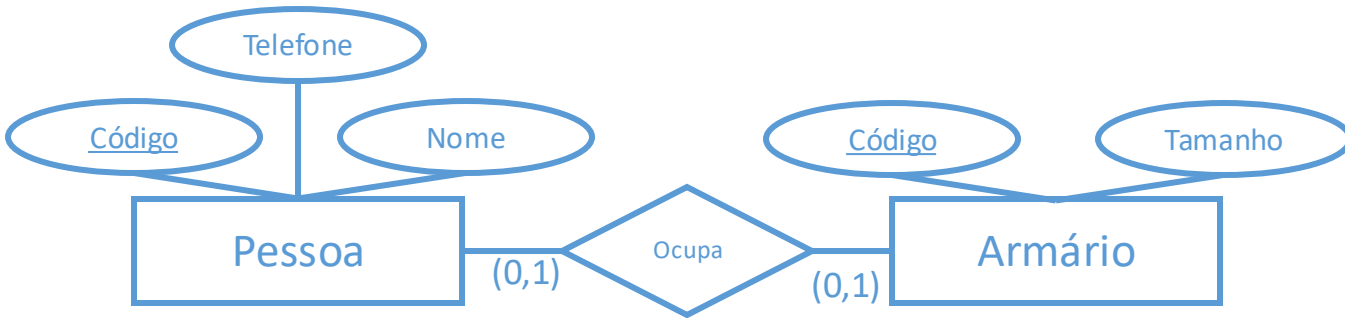
# Passo 3: Mapear Conjuntos de Relacionamento Binário 1:1

- Ambas as entidades com participação opcional

| Tipo de Relacionamento      | Regra de Implementação |               |               |
|-----------------------------|------------------------|---------------|---------------|
|                             | Tabela Própria         | Adição Coluna | Fusão Tabelas |
| <u>(0,1)</u> ◊ <u>(0,1)</u> | 2                      | ✓             | ✗             |
| <u>(0,1)</u> ◊ <u>(1,1)</u> | 3                      | 2             | ✓             |
| <u>(1,1)</u> ◊ <u>(1,1)</u> | 3                      | 3             | ✓             |

Adição de coluna:

- Chave primária de uma das relações torna-se chave estrangeira da outra



Pessoa (CodPessoa, Nome, Telefone)  
Armario (CodArmario, Tamanho, CodPessoa (FK))

PESSOA

| <u>CodPessoa</u> | Nome       | Telefone |
|------------------|------------|----------|
| 1                | Carlos     | 22343221 |
| 2                | Ana        | 17453457 |
| 3                | Arthur     | 78541257 |
| 4                | Renato     | 78954214 |
| 5                | Francielle | 86473542 |

ARMARIO

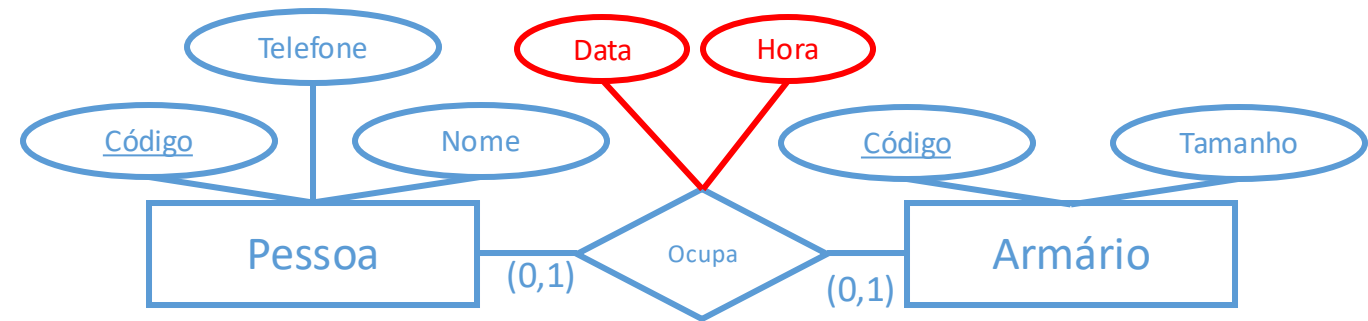
| <u>CodArmario</u> | Tamanho | CodPessoa |
|-------------------|---------|-----------|
| A                 | Simples | 1         |
| B                 | Duplo   | NULL      |
| C                 | Simples | NULL      |
| D                 | Duplo   | 3         |

# Passo 3: Mapear Conjuntos de Relacionamento Binário 1:1

## Atributos de relacionamento

- Ambas as entidades com participação opcional e atributos de relacionamento

| Tipo de Relacionamento | Regra de Implementação |               |               |
|------------------------|------------------------|---------------|---------------|
|                        | Tabela Própria         | Adição Coluna | Fusão Tabelas |
| $(0,1) \diamond (0,1)$ | 2                      | ✓             | ✗             |
| $(0,1) \diamond (1,1)$ | 3                      | 2             | ✓             |
| $(1,1) \diamond (1,1)$ | 3                      | 3             | ✓             |



Pessoa (CodPessoa, Nome, Telefone)

Armario (CodArmario, Tamanho, CodPessoa (FK), Data, Hora)

Adição de coluna:

- Chave primária de uma das relações torna-se chave estrangeira da outra

PESSOA

| <u>CodPessoa</u> | Nome       | Telefone |
|------------------|------------|----------|
| 1                | Carlos     | 22343221 |
| 2                | Ana        | 17453457 |
| 3                | Arthur     | 78541257 |
| 4                | Renato     | 78954214 |
| 5                | Francielle | 86473542 |

ARMARIO

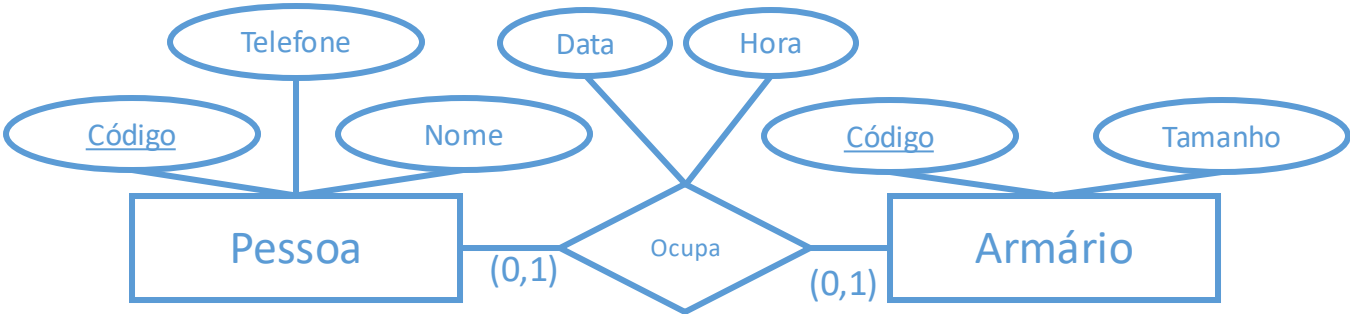
| <u>CodArmario</u> | Tamanho | CodPessoa | Data     | Hora  |
|-------------------|---------|-----------|----------|-------|
| A                 | Simples | 1         | 11/03/20 | 11:40 |
| B                 | Duplo   | NULL      | 15/10/17 | 21:30 |
| C                 | Simples | NULL      | 01/12/18 | 18:02 |
| D                 | Duplo   | 3         | 03/02/21 | 10:00 |



# Passo 3: Mapear Conjuntos de Relacionamento Binário 1:1

- Ambas as entidades com participação opcional

| Tipo de Relacionamento      | Regra de Implementação |               |               |
|-----------------------------|------------------------|---------------|---------------|
|                             | Tabela Própria         | Adição Coluna | Fusão Tabelas |
| <u>(0,1)</u> ◊ <u>(0,1)</u> | 2                      | ✓             | ✗             |
| <u>(0,1)</u> ◊ (1,1)        | 3                      | 2             | ✓             |
| (1,1) ◊ (1,1)               | 3                      | 3             | ✓             |



Pessoa (CodPessoa, Nome, Telefone)  
Ocupa (CodPessoa (FK), CodArmario (FK), Data, Hora)  
Armário (CodArmario, Tamanho)

2 Relacionamento se transforma em terceira relação

| PESSOA           |            |          | OCUPA             |                  |          |       | ARMARIO           |         |
|------------------|------------|----------|-------------------|------------------|----------|-------|-------------------|---------|
| <u>CodPessoa</u> | Nome       | Telefone | <u>CodArmario</u> | <u>CodPessoa</u> | Data     | Hora  | <u>CodArmario</u> | Tamanho |
| 1                | Carlos     | 22343221 | A                 | 1                | 11/03/20 | 11:40 | A                 | Simples |
| 2                | Ana        | 17453457 | B                 | NULL             | 15/10/17 | 21:30 | B                 | Duplo   |
| 3                | Arthur     | 78541257 | C                 | NULL             | 01/12/18 | 18:02 | C                 | Simples |
| 4                | Renato     | 78954214 | D                 | 3                | 03/02/21 | 10:00 | D                 | Duplo   |
| 5                | Francielle | 86473542 |                   |                  |          |       |                   |         |

# Sumário

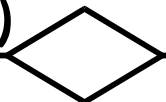
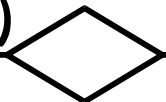
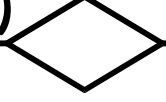
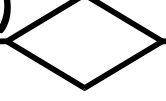
- **Algoritmo de mapeamento do ER para o Relacional**

- Passo 1: Mapear Conjuntos de Entidades Regulares
- Passo 2: Mapear Conjuntos de Entidades Fracas
- Passo 3: Mapear Conjuntos de Relacionamento Binário 1:1
- **Passo 4: Mapear Conjuntos de Relacionamento Binário Regular 1:N**
- Passo 5: Mapear Conjuntos de Relacionamento Binário M:N
- Passo 6: Mapear Atributos Multivalorados ou Compostos
- Passo 7: Mapear Conjuntos de Relacionamentos n-ário,  $n > 2$
- Passo 8: Relacionamento Recursivo

- **Mapeamento EER para Relações**

- Passo 9: Opções para mapeamento de especialização ou generalização

## Passo 4: Mapear Conjuntos de Relacionamento Binário Regular 1:n

| Tipo de Relacionamento  | Regra de Implementação |               |               |
|---|------------------------|---------------|---------------|
|   | Tabela Própria         | Adição Coluna | Fusão Tabelas |
| <u>(0,1)</u>  <u>(0,n)</u>   | ②                      | ✓             | ✗             |
| <u>(0,1)</u>  <u>(1,n)</u>   | ②                      | ✓             | ✗             |
| <u>(1,1)</u>  <u>(0,n)</u>  | ③                      | ✓             | ✗             |
| <u>(1,1)</u>  <u>(1,n)</u> | ③                      | ✓             | ✗             |

(Heuser, 2004)

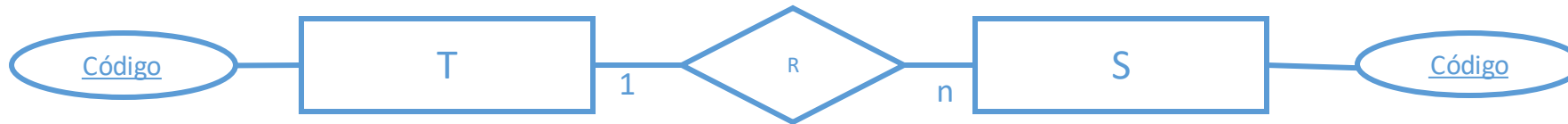


## Passo 4:

# Mapear Conjuntos de Relacionamento Binário Regular 1:n

- **Abordagem Adição de Coluna (técnica de chave estrangeira)**

- Identifique a relação S que representa o tipo de entidade participante no lado N do tipo de relacionamento. Inclua como chave estrangeira em S a chave primária da relação T que representa o outro tipo de entidade participante em R;
- Inclua todos os atributos simples (ou componentes simples dos atributos compostos) do tipo relacionamento como atributos de S.

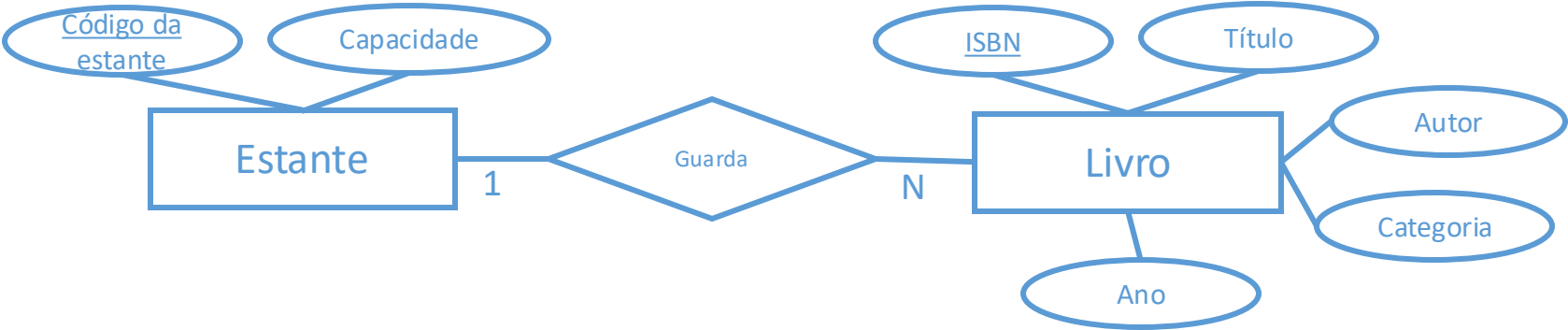


T(codT, ....)

S (codS, codT (FK), <atributos de relacionamento, ...>)

# Passo 4:

## Mapear Conjuntos de Relacionamento Binário Regular 1:n

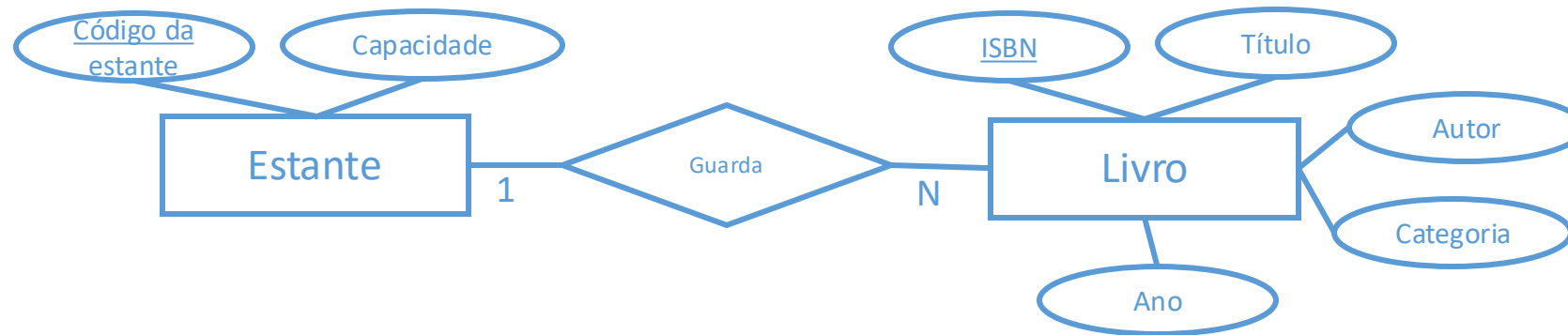


| Tipo de Relacionamento | Regra de Implementação |               |               |
|------------------------|------------------------|---------------|---------------|
|                        | Tabela Própria         | Adição Coluna | Fusão Tabelas |
| <u>(0,1)</u> —◇— (0,n) | 2                      | ✓             | ✗             |
| <u>(0,1)</u> —◇— (1,n) | 2                      | ✓             | ✗             |
| <u>(1,1)</u> —◇— (0,n) | 3                      | ✓             | ✗             |
| <u>(1,1)</u> —◇— (1,n) | 3                      | ✓             | ✗             |



## Passo 4:

# Mapear Conjuntos de Relacionamento Binário Regular 1:n



Estante (CodEstante, capacidade)

Livro (ISBN, título, autor, categoria, ano, CodEstante (FK))

ESTANTE

| <u>codEstante</u> | Capacidade |
|-------------------|------------|
| 1                 | 10         |
| 2                 | 30         |
| 3                 | 100        |

LIVRO

| <u>ISBN</u> | Título             | Autor              | Categoria | Ano  | codEstante |
|-------------|--------------------|--------------------|-----------|------|------------|
| 1           | Dom Casmurro       | Machado de Assis   | Ficção    | 1899 | 1          |
| 2           | A Moreninha        | Joaquim Macedo     | Romance   | 1844 | 1          |
| 4           | A máquina do tempo | Simon Wells        | Ação      | 2002 | 3          |
| 5           | O Cortiço          | Aluísio de Azevedo | Romance   | 1890 | 2          |
| 6           | A Coisa            | Stephen King       | Terro     | 1986 | 3          |



# Sumário

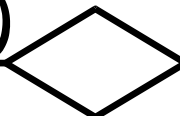
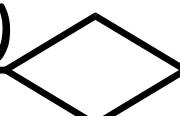
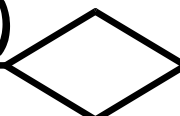
- **Algoritmo de mapeamento do ER para o Relacional**

- Passo 1: Mapear Conjuntos de Entidades Regulares
- Passo 2: Mapear Conjuntos de Entidades Fracas
- Passo 3: Mapear Conjuntos de Relacionamento Binário 1:1
- Passo 4: Mapear Conjuntos de Relacionamento Binário Regular 1:N
- **Passo 5: Mapear Conjuntos de Relacionamento Binário M:N**
- Passo 6: Mapear Atributos Multivalorados ou Compostos
- Passo 7: Mapear Conjuntos de Relacionamentos n-ário,  $n > 2$
- Passo 8: Relacionamento Recursivo

- **Mapeamento EER para Relações**

- Passo 9: Opções para mapeamento de especialização ou generalização

## Passo 5: Mapear Conjuntos de Relacionamento Binário m:n

| Tipo de Relacionamento   | Regra de Implementação |               |               |
|--|------------------------|---------------|---------------|
|  | Tabela Própria         | Adição Coluna | Fusão Tabelas |
| <u>(0,n)</u>  <u>(0,n)</u>  | ✓                      | ✗             | ✗             |
| <u>(0,n)</u>  <u>(1,n)</u>  | ✓                      | ✗             | ✗             |
| <u>(1,n)</u>  <u>(1,n)</u> | ✓                      | ✗             | ✗             |

(Heuser, 2004)



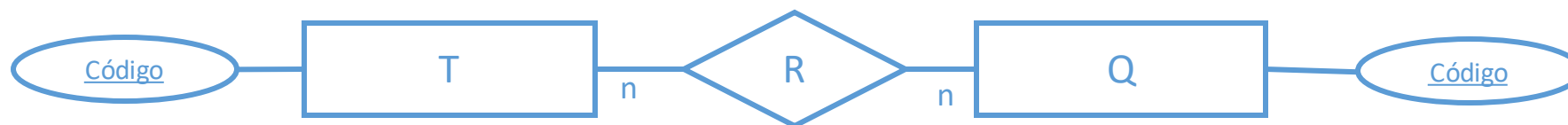


# Regras de Implementação

## Tabela própria

- **Tabela Própria**

- Crie uma nova relação S para representar R
- Inclua como atributos de chave estrangeira em S as chaves primárias das relações que representam os tipos de entidade participantes; sua combinação formará a chave primária de S
- Inclua também quaisquer atributos simples do tipo de relacionamento M:N (ou componentes simples dos atributos compostos) como atributos de S

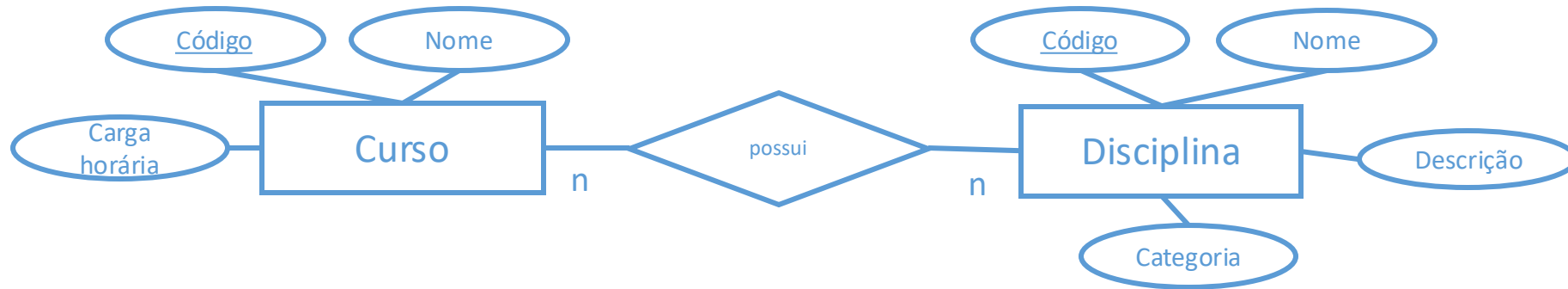


T(codT, ....)

S(codT (FK), codQ(FK) , <atributos de relacionamento, ...>)

Q (codQ, ...)

# Passo 5: Mapear Conjuntos de Relacionamento Binário m:n



Curso (CodCurso, nome, cargaHoraria)

Disciplina (CodDisciplina, nome, descricao, categoria)

CursoDisciplina (CodCurso (FK), CodDisciplina (FK))

| CURSO           |      |              | CURSODISCIPLINA |                      | DISCIPLINA           |         |           |             |
|-----------------|------|--------------|-----------------|----------------------|----------------------|---------|-----------|-------------|
| <u>CodCurso</u> | Nome | cargaHoraria | <u>CodCurso</u> | <u>CodDisciplina</u> | <u>CodDisciplina</u> | Nome    | Descricao | Categoria   |
| 1               | ES   | 22343221     | 1               | 11                   | 11                   | BD      | A         | Obrigatória |
| 2               | CC   | 17453457     | 1               | 22                   | 22                   | LBD     | B         | Optativa    |
| 3               | SI   | 78541257     | 2               | 11                   | 33                   | AlgProg | C         | Obrigatória |
| 4               | TADS | 45678944     | 3               | 33                   |                      |         |           |             |
|                 |      |              | 3               | 22                   |                      |         |           |             |



# Sumário

- **Algoritmo de mapeamento do ER para o Relacional**

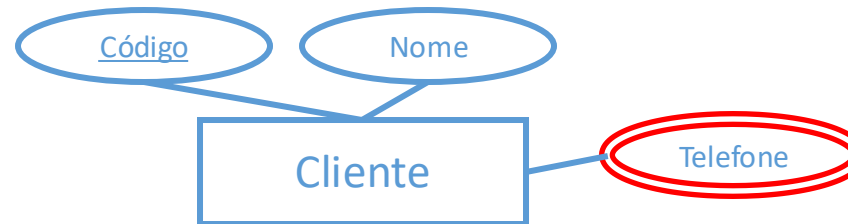
- Passo 1: Mapear Conjuntos de Entidades Regulares
- Passo 2: Mapear Conjuntos de Entidades Fracas
- Passo 3: Mapear Conjuntos de Relacionamento Binário 1:1
- Passo 4: Mapear Conjuntos de Relacionamento Binário Regular 1:N
- Passo 5: Mapear Conjuntos de Relacionamento Binário M:N
- **Passo 6: Mapear Atributos Multivalorados ou Compostos**
- Passo 7: Mapear Conjuntos de Relacionamentos n-ário,  $n > 2$
- Passo 8: Relacionamento Recursivo

- **Mapeamento EER para Relações**

- Passo 9: Opções para mapeamento de especialização ou generalização

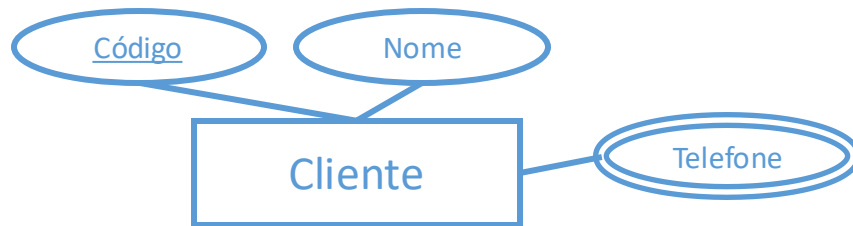
# Passo 6: Mapear Atributos Multivalorados e Compostos

Modelo relacional não permite atributo multivalorado



# Passo 6: Mapear Atributos Multivalorados

- Atributo vira tabela M
- Chave primária de entidade vira chave estrangeira de M
- Chave primária de M
  - Chave primária da entidade + Atributo multivalorado



Cliente (CodCliente, Nome)

Telefone (CodCliente (FK), NumTelefone)

CLIENTE

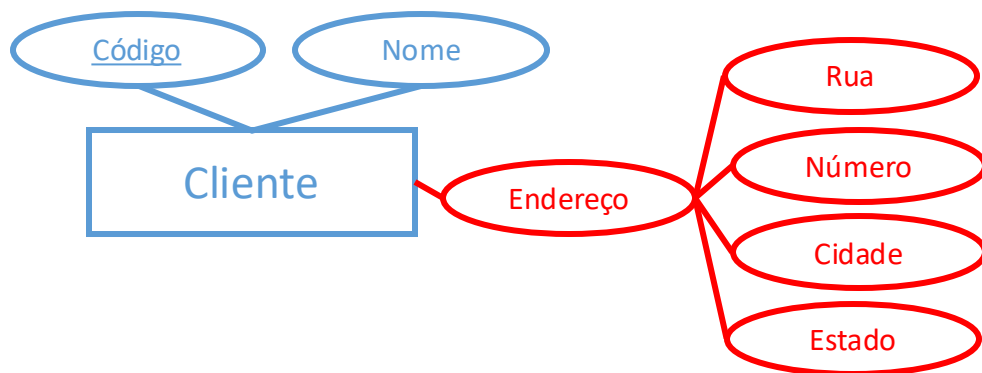
| <u>CodCliente</u> | Nome       |
|-------------------|------------|
| 1                 | Carlos     |
| 2                 | Ana        |
| 3                 | Arthur     |
| 4                 | Renato     |
| 5                 | Francielle |

TELEFONE

| <u>CodCliente</u> | <u>NumTelefone</u> |
|-------------------|--------------------|
| 1                 | 11111111           |
| 1                 | 22222222           |
| 4                 | 33333333           |
| 4                 | 44444444           |
| 4                 | 55555555           |

# Passo 6: Mapear Atributos Compostos

- Se atributo for composto, componentes viram coluna de **M**



Cliente (CodCliente, Nome, Rua, Numero, Cidade, Estado)

CLIENTE

| <u>CodCliente</u> | Nome       | Rua             | Numero | Cidade | Estado |
|-------------------|------------|-----------------|--------|--------|--------|
| 1                 | Carlos     | Cândido Mariano | 11     | CG     | MS     |
| 2                 | Ana        | Teodoro Sampaio | 15     | SP     | SP     |
| 3                 | Arthur     | Sabiá           | 2      | POA    | NULL   |
| 4                 | Renato     | NULL            | NULL   | NULL   | NULL   |
| 5                 | Francielle | Itaipu          | 8      | CG     | MS     |

OU

Cliente (CodCliente, Nome)

Endereco (CodCliente (FK), Rua, Numero, Cidade, Estado)

# Sumário

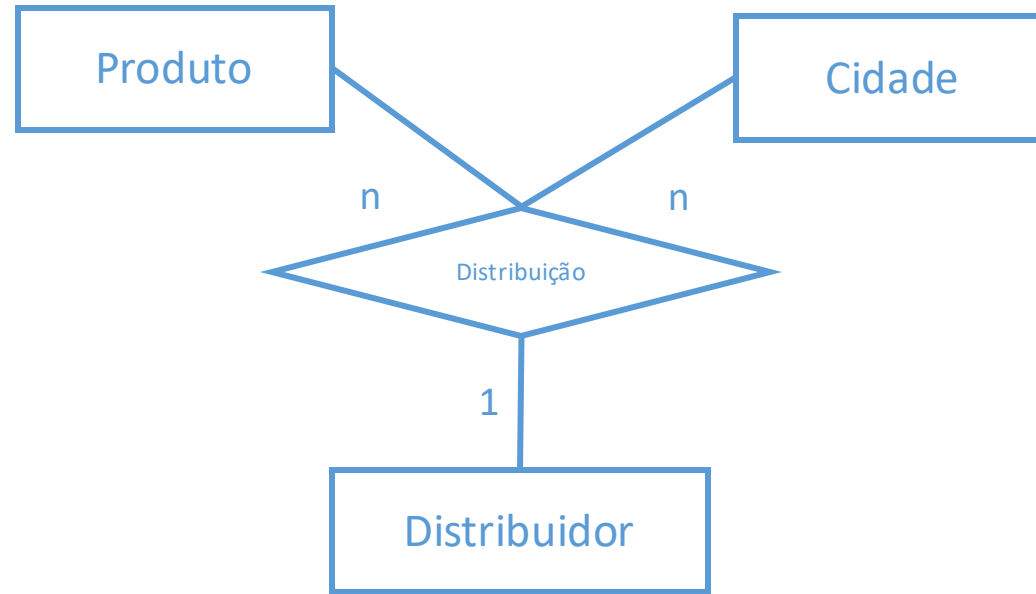
- **Algoritmo de mapeamento do ER para o Relacional**

- Passo 1: Mapear Conjuntos de Entidades Regulares
- Passo 2: Mapear Conjuntos de Entidades Fracas
- Passo 3: Mapear Conjuntos de Relacionamento Binário 1:1
- Passo 4: Mapear Conjuntos de Relacionamento Binário Regular 1:N
- Passo 5: Mapear Conjuntos de Relacionamento Binário M:N
- Passo 6: Mapear Atributos Multivalorados ou Compostos
- **Passo 7: Mapear Conjuntos de Relacionamentos n-ário,  $n > 2$**
- Passo 8: Relacionamento Recursivo

- **Mapeamento EER para Relações**

- Passo 9: Opções para mapeamento de especialização ou generalização

## Passo 7: Mapear Conjuntos de Relacionamentos n-ário, $n > 2$



**Relação de relacionamento (tabela própria)**



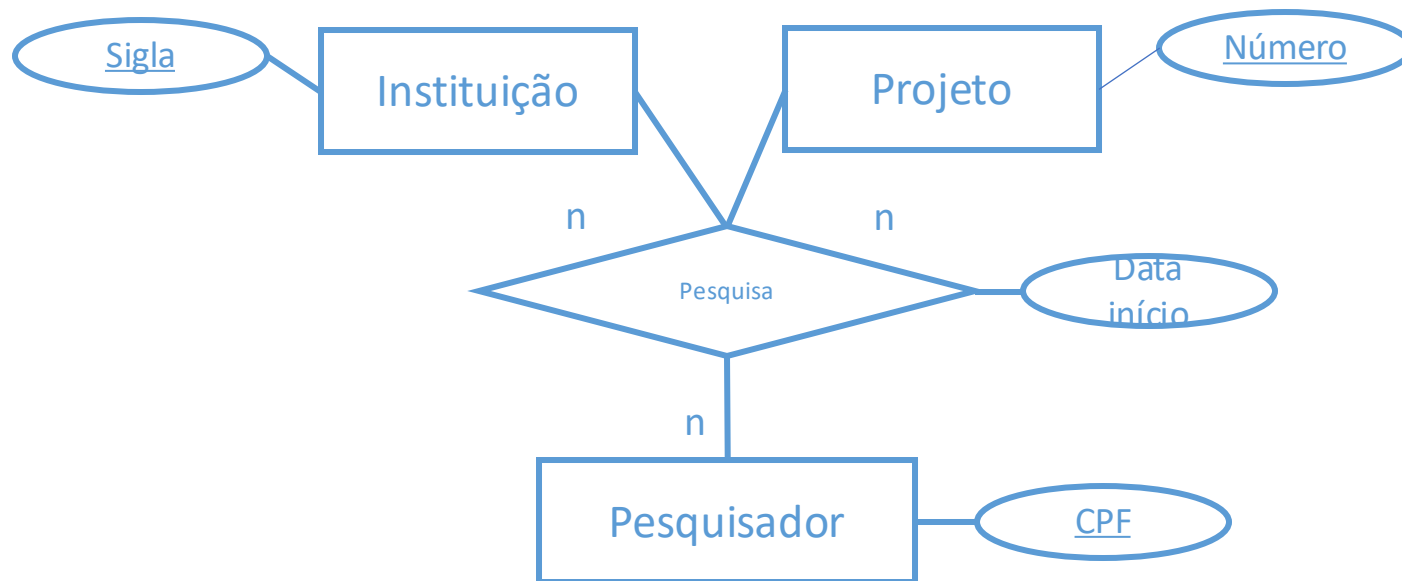


# Passo 7: Mapear Conjuntos de Relacionamentos n-ário, $n > 2$

## Segundo Navathe:

1. Para cada tipo de relacionamento n-ário  $R$ , onde  $n > 2$ , crie uma relação  $S$  para representar  $R$ .
2. Inclua como atributos de chave estrangeira em  $S$  as chaves primárias das relações que representam os tipos de entidade participantes.
3. Inclua também quaisquer atributos simples do tipo de relacionamento n-ário (ou componentes simples de atributos compostos) como atributos de  $S$ .
4. A chave primária de  $S$  normalmente é uma combinação de todas as chaves estrangeiras que referenciam as relações representando os tipos de entidade participantes.
  - Porém, se as restrições de cardinalidade sobre qualquer **um dos tipos** de entidade  $E$  participantes em  $R$  for 1, então a chave primária de  $S$  **não** deve incluir o atributo de chave estrangeira que referencia a relação  $E'$  correspondente a  $E$

## Passo 7: Mapear Conjuntos de Relacionamentos n-ário, $n > 2$ (n:n:n)



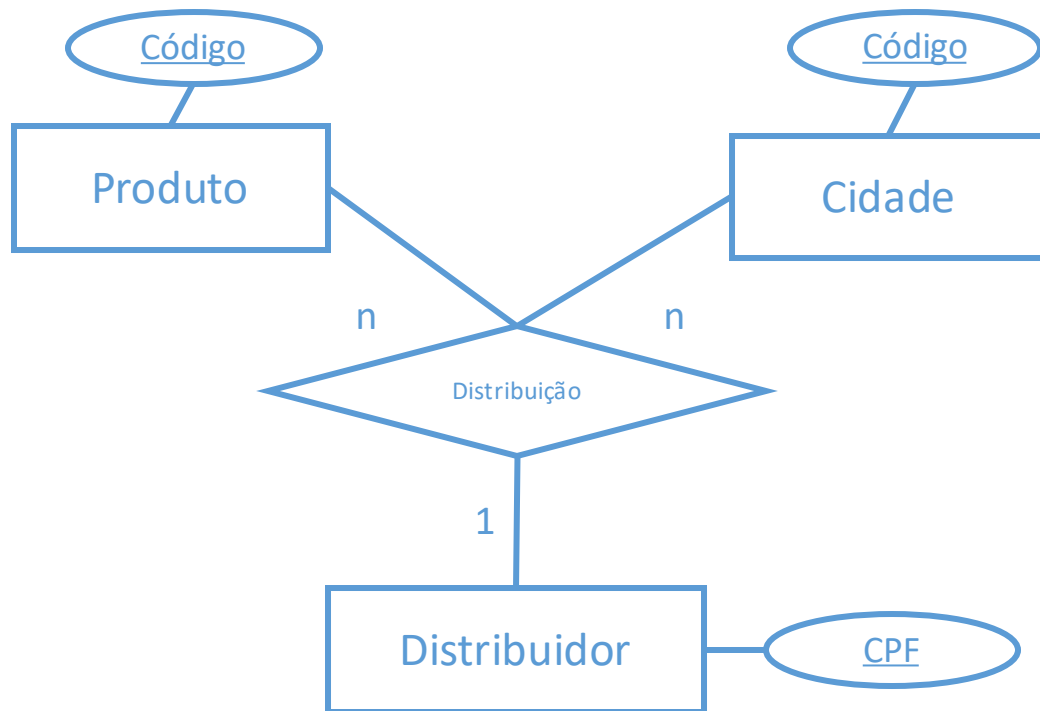
Instituicao (Sigla, ...)  
Projeto (Numero, ...)  
Pesquisador (CPF, ...)  
Pesquisa (Sigla (FK), Numero(FK), CPF(FK), DataInício)



**Relação de relacionamento (tabela própria)**



## Passo 7: Mapear Conjuntos de Relacionamentos n-ário, $n > 2$ (1:n:n)



Produto (CodProduto, ...)

Cidade (CodCidade, ...)

Distribuidor (CPF, ...)

Distribuicao (CodProduto(FK), CodCidade(FK), CPF(FK))

CodProduto, CodCidade → CPF



**Relação de relacionamento (tabela própria)**



# Sumário

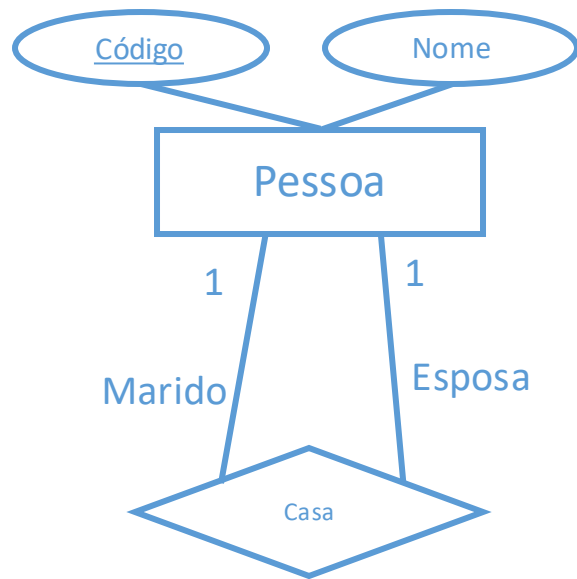
- **Algoritmo de mapeamento do ER para o Relacional**

- Passo 1: Mapear Conjuntos de Entidades Regulares
- Passo 2: Mapear Conjuntos de Entidades Fracas
- Passo 3: Mapear Conjuntos de Relacionamento Binário 1:1
- Passo 4: Mapear Conjuntos de Relacionamento Binário Regular 1:N
- Passo 5: Mapear Conjuntos de Relacionamento Binário M:N
- Passo 6: Mapear Atributos Multivalorados ou Compostos
- Passo 7: Mapear Conjuntos de Relacionamentos n-ário,  $n > 2$
- Passo 8: Relacionamento Recursivo

- **Mapeamento EER para Relações**

- Passo 9: Opções para mapeamento de especialização ou generalização

## Passo 8: Mapear Conjuntos de Relacionamento Recursivo 1:1



Pessoa (CodPessoa, NomePessoa, CodConjuje(FK))

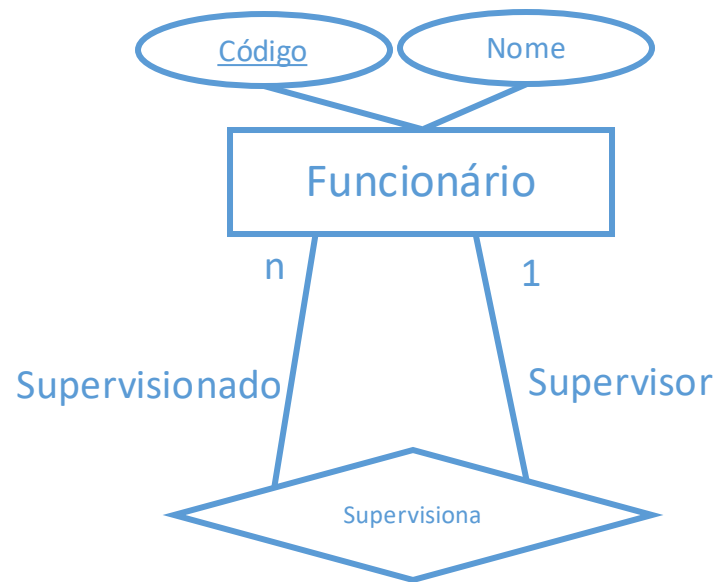
PESSOA

| <u>CodPessoa</u> | nome    | CodConjuje |
|------------------|---------|------------|
| 1                | Carlos  | 7          |
| 2                | Miguel  | NULL       |
| 3                | Augusto | 6          |
| 4                | Ana     | 5          |
| 5                | Bruno   | 4          |
| 6                | Alice   | 3          |
| 7                | Renata  | 1          |



## Passo 8:

# Mapear Conjuntos de Relacionamento Recursivo 1:n



Funcionario(CodFuncionario, NomeFuncionario, CodSupervisor (FK))

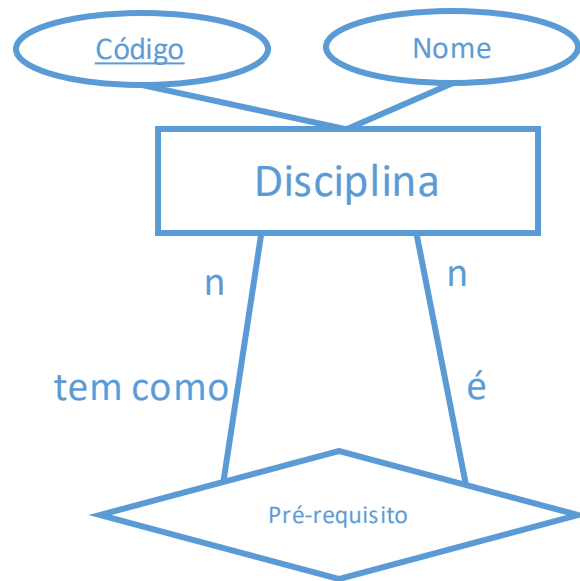
FUNCIONARIO

| <u>CodFuncionario</u> | nome    | CodSupervisor |
|-----------------------|---------|---------------|
| 1                     | Carlos  | 7             |
| 2                     | Miguel  | 7             |
| 3                     | Augusto | NULL          |
| 4                     | Ana     | 7             |
| 5                     | Bruno   | 3             |
| 6                     | Alice   | 3             |
| 7                     | Renata  | 7             |



# Passo 8:

## Mapear Conjuntos de Relacionamento Recursivo n:n



Disciplina (codDisciplina, nomeDisciplina)

Prerequisito (codDisciplina (FK), codDisciplinaPrerequisito (FK))

DISCIPLINA

| <u>CodDisciplina</u> | nome     |
|----------------------|----------|
| 1                    | BD       |
| 2                    | LBD      |
| 3                    | ALGPROG1 |
| 4                    | ALGPROG2 |
| 5                    | LFA      |
| 6                    | FTC      |
| 7                    | ED       |
| 8                    | ARQC1    |
| 9                    | ARQC2    |

PREREQUISITO

| <u>CodDisciplina</u> | <u>CodDisciplinaPrerequisito</u> |
|----------------------|----------------------------------|
| 2                    | 1                                |
| 7                    | 4                                |
| 5                    | 6                                |
| 4                    | 3                                |
| 8                    | 3                                |
| 8                    | 9                                |



# Sumário

- **Algoritmo de mapeamento do ER para o Relacional**

- Passo 1: Mapear Conjuntos de Entidades Regulares
- Passo 2: Mapear Conjuntos de Entidades Fracas
- Passo 3: Mapear Conjuntos de Relacionamento Binário 1:1
- Passo 4: Mapear Conjuntos de Relacionamento Binário Regular 1:N
- Passo 5: Mapear Conjuntos de Relacionamento Binário M:N
- Passo 6: Mapear Atributos Multivalorados ou Compostos
- Passo 7: Mapear Conjuntos de Relacionamentos n-ário,  $n > 2$
- Passo 8: Relacionamento Recursivo

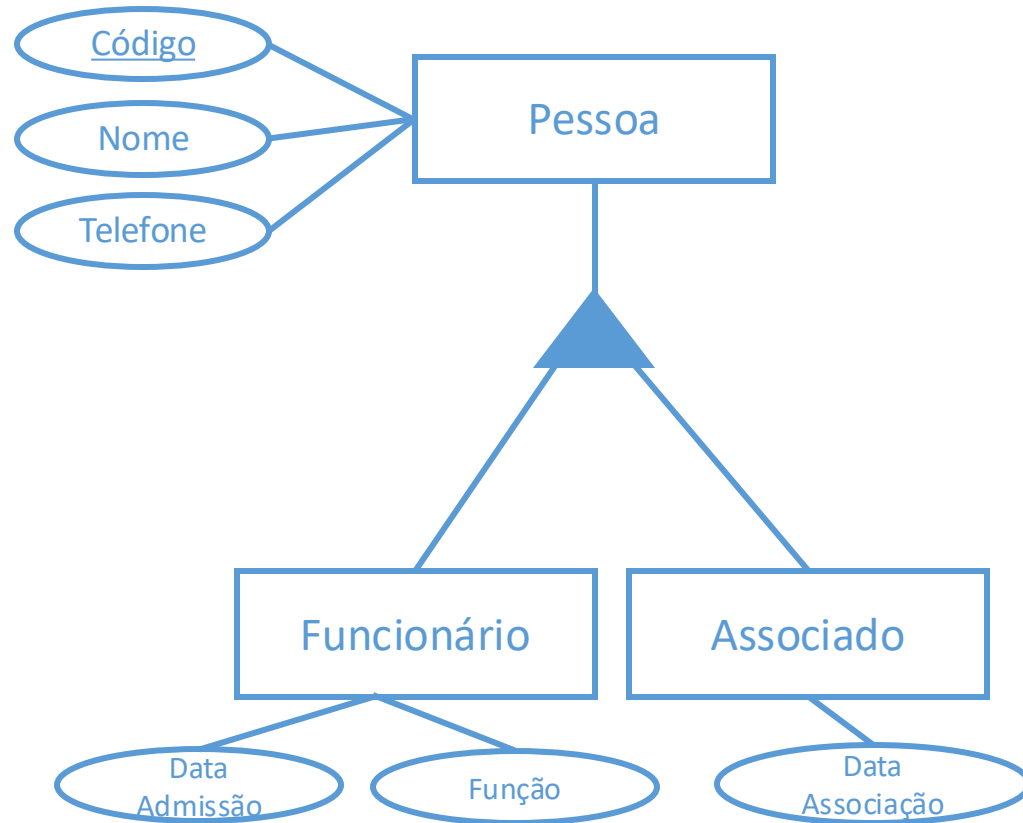
- **Mapeamento EER para Relações**

- Passo 9: Opções para mapeamento de especialização ou generalização

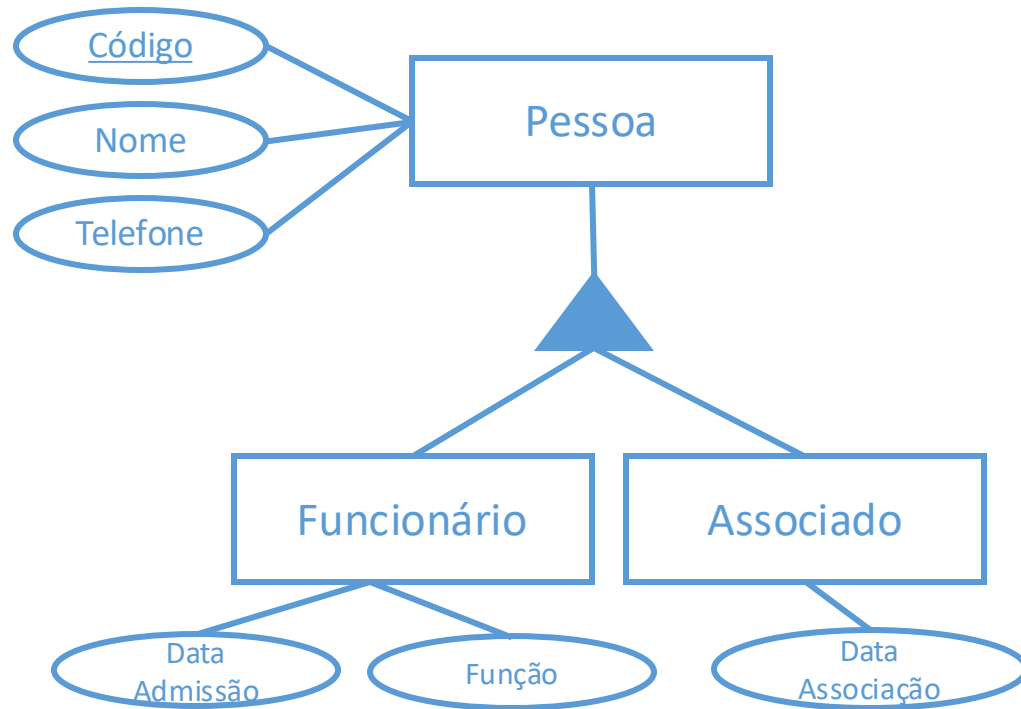




# Passo 9: Mapear Generalização e Especialização



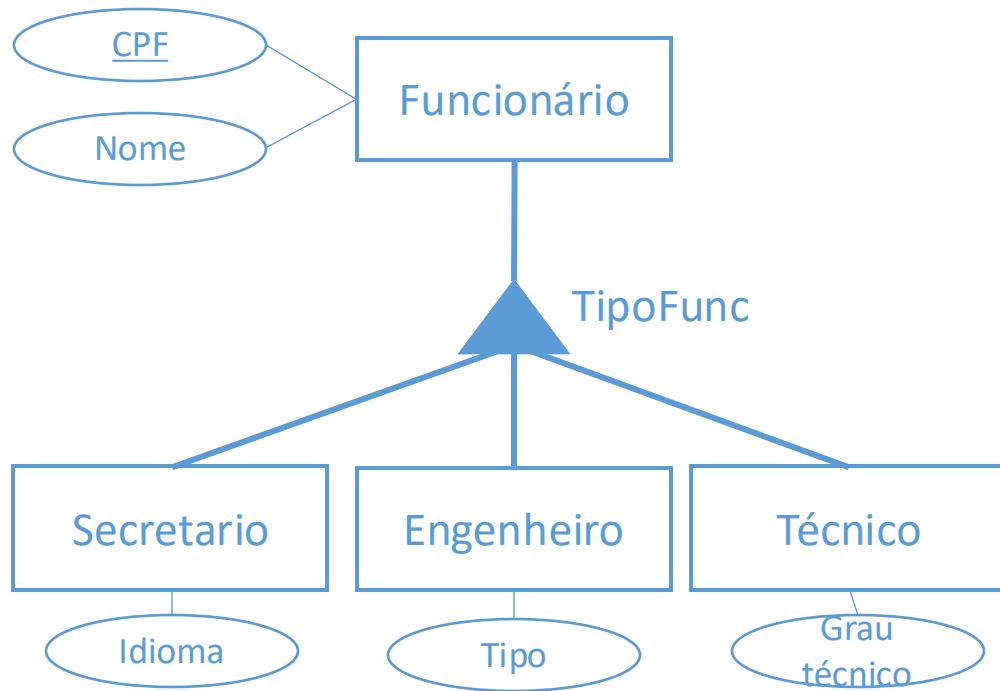
# Passo 9: Mapear Generalização e Especialização



1. Uma tabela para tudo (por hierarquia)
2. Uma tabela para cada herdeiro (entidade especializada)
3. Uma tabela para a superclasse e uma tabela para as subclasses
4. Uma tabela para cada uma das entidades



# Passo 9: Mapear Generalização e Especialização

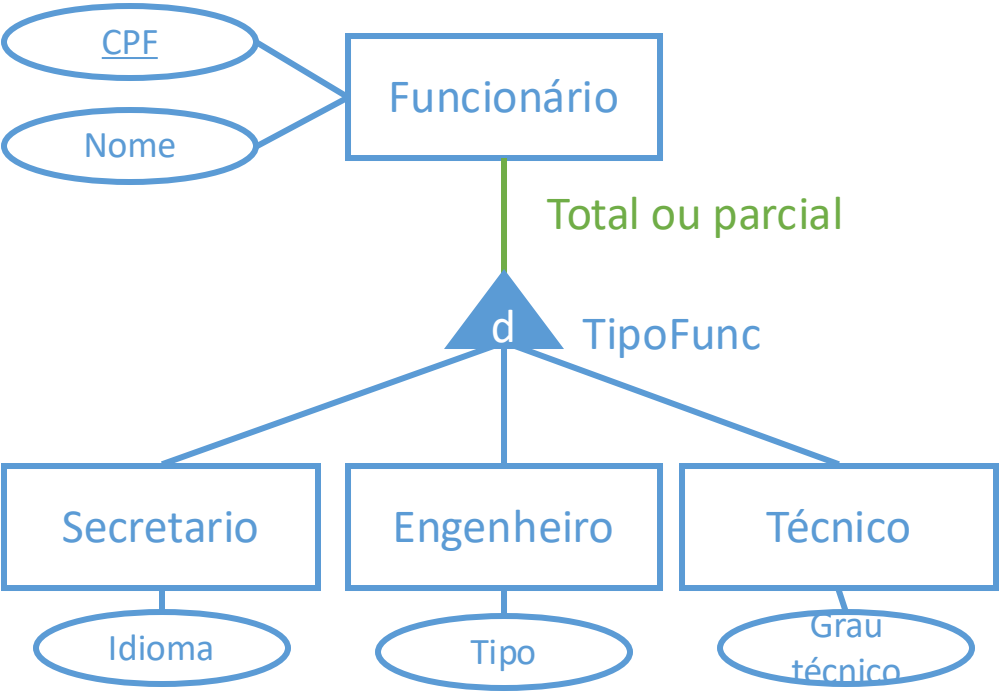


Funcionario (CPFFunc, nomeFunc, tipoFunc, idiomaSec, tipoEng, grauTec)

1. Uma tabela para tudo (por hierarquia)
2. Uma tabela para cada herdeiro (entidade especializada)
3. Uma tabela para a superclasse e uma tabela para as subclasses
4. Uma tabela para cada uma das entidades



# Passo 9: Mapear Generalização e Especialização



Funcionario (CPFFunc, nomeFunc, tipoFunc, idiomaSec, tipoEng, grauTec)

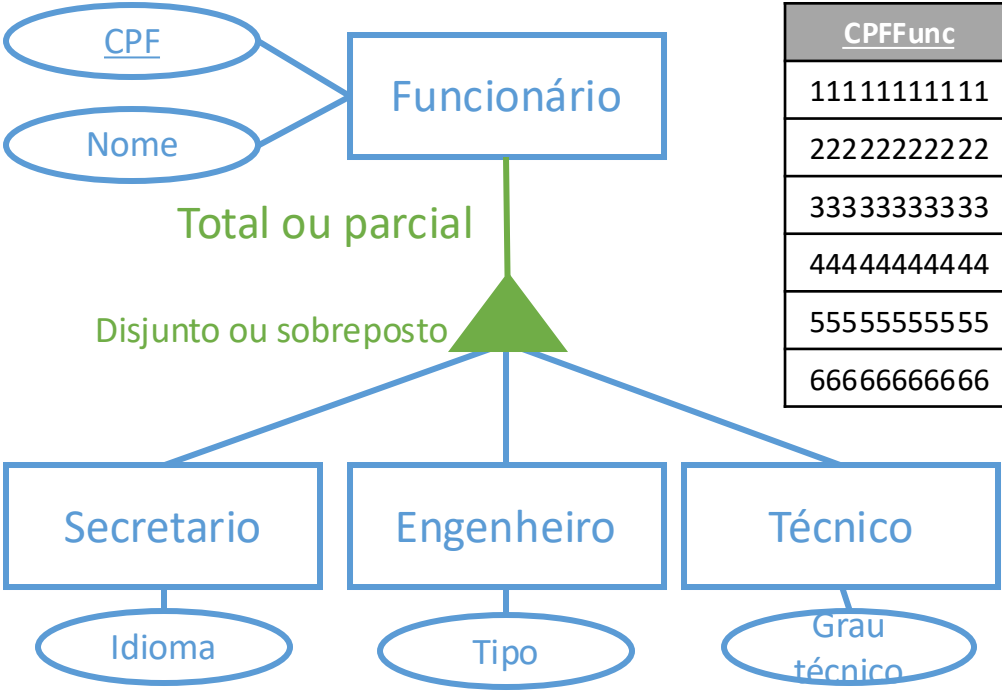
FUNCIONARIO (instância - parcial / disjunto)

| CPFFunc     | nomeFunc | tipoFunc | idiomaSec | tipoEng     | grauTec       |
|-------------|----------|----------|-----------|-------------|---------------|
| 11111111111 | Carlos   | S        | P         | NULL        | NULL          |
| 22222222222 | Ana      | E        | NULL      | Pesquisador | NULL          |
| 33333333333 | Miguel   | T        | NULL      | NULL        | Superior      |
| 44444444444 | Pedro    | T        | NULL      | NULL        | Intermediário |
| 55555555555 | Lucas    | S        | P         | NULL        | NULL          |
| 66666666666 | Antônio  | NULL     | NULL      | NULL        | NULL          |

- Essa opção funciona somente para especializações cujas subclasses são disjuntas
- Pode ser utilizado com completude total ou parcial
- Essa opção pode gerar muitos valores NULL se as subclasses tiverem muitos atributos específicos ou se a restrição de completude for parcial.



# Passo 9: Mapear Generalização e Especialização



**FUNCIONÁRIO (instância – parcial / sobreposto)**

| CPFFunc     | nomeFunc | tipoS | tipoE | TipoT | idiomaSec | tipoEng     | grauTec       |
|-------------|----------|-------|-------|-------|-----------|-------------|---------------|
| 11111111111 | Carlos   | True  | False | False | P         | NULL        | NULL          |
| 22222222222 | Ana      | False | True  | False | NULL      | Pesquisador | NULL          |
| 33333333333 | Miguel   | False | False | True  | NULL      | NULL        | Superior      |
| 44444444444 | Pedro    | False | False | True  | NULL      | NULL        | Intermediário |
| 55555555555 | Lucas    | True  | True  | False | P         | Operador    | NULL          |
| 66666666666 | Antônio  | False | False | False | NULL      | NULL        | NULL          |

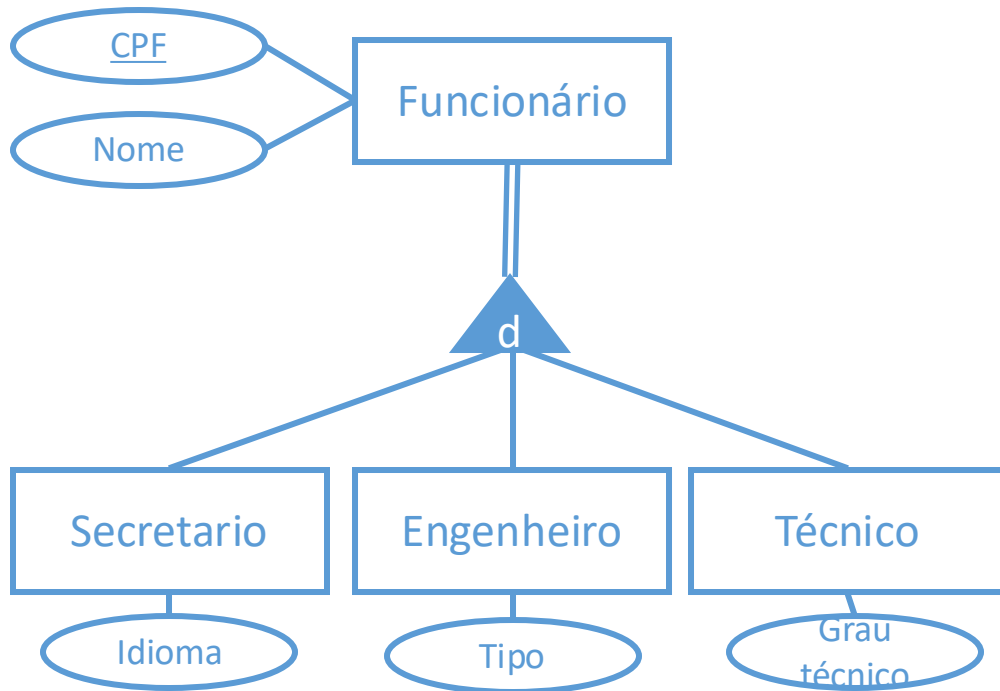
- Ela também funciona para a restrição de disjunção
- Essa opção pode gerar muitos valores NULL se as subclasses tiverem muitos atributos específicos.

Funcionario (CPFFunc, nomeFunc, tipoS, tipoE, tipoT, idiomaSec, tipoEng, grauTec)

- Onde: tipoS, tipoE e tipoT são valores booleanos



# Passo 9: Mapear Generalização e Especialização



Secretario (CPFFunc, NomeFunc, IdiomaSec)

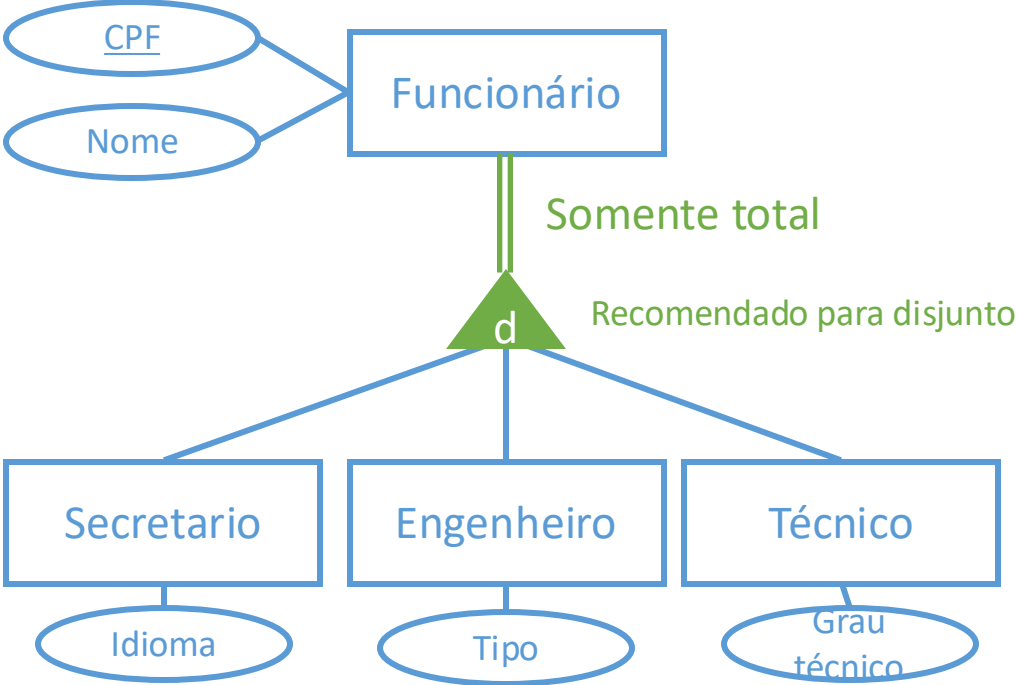
Engenheiro (CPFFunc, NomeFunc, TipoEng)

Tecnico (CPFFunc, NomeFunc, GrauTec)

1. Uma tabela para tudo (por hierarquia)
2. Uma tabela para cada herdeiro (entidade especializada)
3. Uma tabela para a superclasse e uma tabela para as subclasses
4. Uma tabela para cada uma das entidades



# Passo 9: Mapear Generalização e Especialização



Secretario (CPFFunc, NomeFunc, IdiomaSec)  
Engenheiro (CPFFunc, NomeFunc, TipoEng)  
Tecnico (CPFFunc, NomeFunc, GrauTec)

(Instâncias – total / disjunto)

## SECRETARIO

| <u>CPFFunc</u> | nomeFunc | idiomaSec |
|----------------|----------|-----------|
| 11111111111    | Carlos   | P         |
| 55555555555    | Lucas    | P         |

## ENGENHEIRO

| <u>CPFFunc</u> | nomeFunc | tipoEng     |
|----------------|----------|-------------|
| 22222222222    | Ana      | Pesquisador |
| 66666666666    | Vicente  | Operador    |

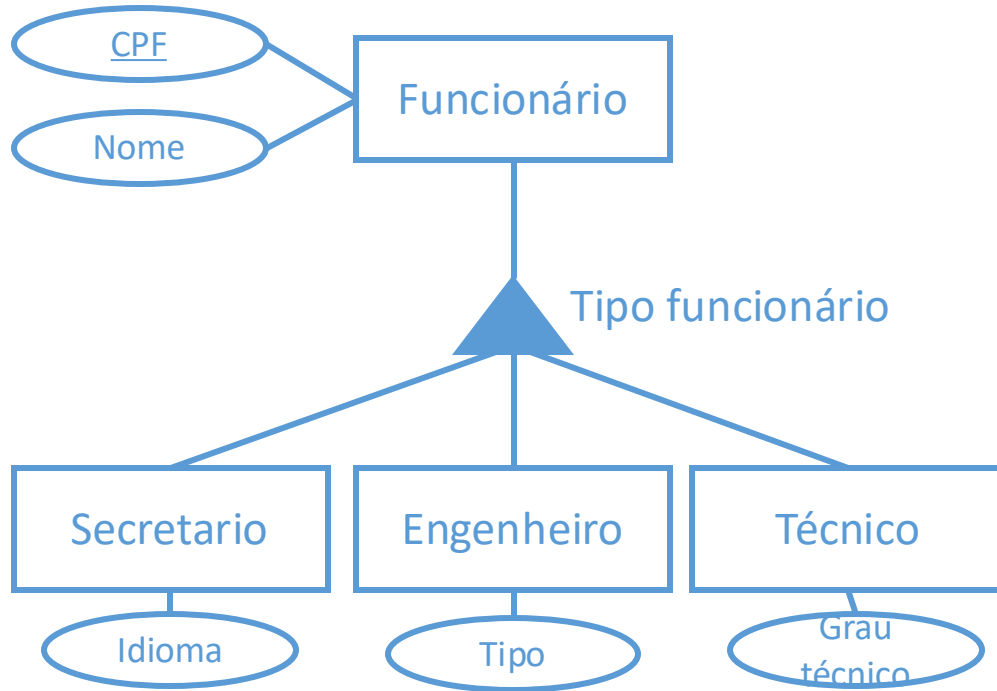
## TECNICO

| <u>CPFFunc</u> | nomeFunc | grauTec       |
|----------------|----------|---------------|
| 33333333333    | Miguel   | Superior      |
| 44444444444    | Pedro    | Intermediário |

- Somente para especializações de participação total
- É recomendada para especializações com restrição de disjunção
  - Poderá haver duplicação de dados em diferentes relações (restrição de sobreposição)



# Passo 9: Mapear Generalização e Especialização



Funcionario (CPFFunc, NomeFunc)

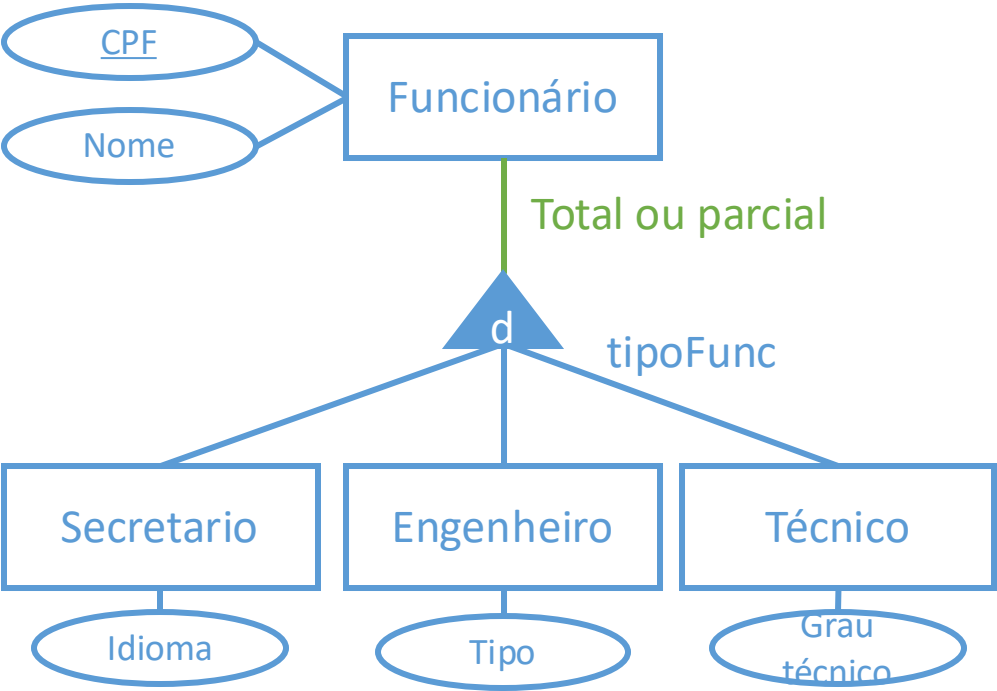
SeEnTe (CPFFunc (FK), tipoFunc, idiomaSec, tipoEng, GrauTec)

1. Uma tabela para tudo (por hierarquia)
2. Uma tabela para cada herdeiro (entidade especializada)
3. Uma tabela para a superclasse e uma tabela para as subclasses
4. Uma tabela para cada uma das entidades





# Passo 9: Mapear Generalização e Especialização



Funcionario (CPFFunc, NomeFunc)  
SeEnTe (CPFFunc (FK), tipoFunc, idiomaSec, tipoEng, GrauTec)

FUNCIONARIO (instância – total / disjunto)

| CPFFunc     | nomeFunc |
|-------------|----------|
| 11111111111 | Carlos   |
| 22222222222 | Ana      |
| 33333333333 | Miguel   |
| 44444444444 | Pedro    |
| 55555555555 | Lucas    |

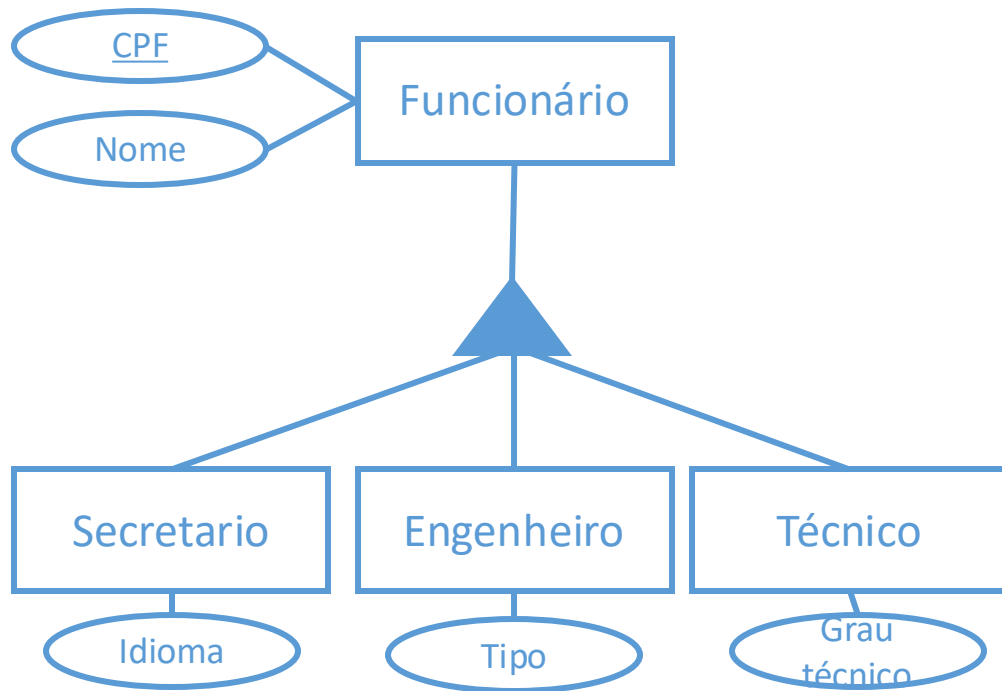
SEENTE

| CPFFunc     | tipoFunc | idiomaSec | tipoEng     | grauTec       |
|-------------|----------|-----------|-------------|---------------|
| 11111111111 | S        | P         | NULL        | NULL          |
| 22222222222 | E        | NULL      | Pesquisador | NULL          |
| 33333333333 | T        | NULL      | NULL        | Superior      |
| 44444444444 | T        | NULL      | NULL        | Intermediário |
| 55555555555 | S        | P         | NULL        | NULL          |

- Essa opção funciona somente para especializações cujas subclasses são disjuntas
- Poderia ser restrição de completude total ou parcial
- Essa opção pode gerar muitos valores NULL se as subclasses tiverem muitos atributos específicos.



# Passo 9: Mapear Generalização e Especialização

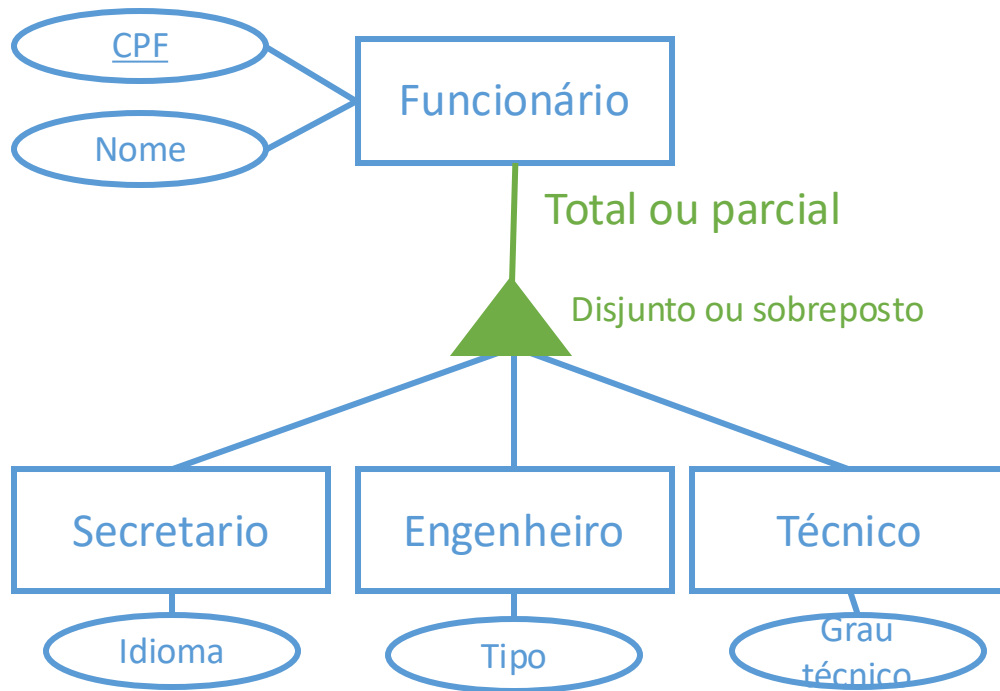


Funcionario (CPFFunc, NomeFunc)  
Secretario (CPFFunc (FK), IdiomaSev)  
Engenheiro (CPFFunc (FK), TipoEng)  
Tecnico (CPFFunc (FK), GrauTec)

1. Uma tabela para tudo (por hierarquia)
2. Uma tabela para cada herdeiro (entidade especializada)
3. Uma tabela para a superclasse e uma tabela para as subclasses
4. Uma tabela para cada uma das entidades



# Passo 9: Mapear Generalização e Especialização



Funcionario (CPFFunc, NomeFunc)  
Secretario (CPFFunc (FK), IdiomaSec)  
Engenheiro (CPFFunc (FK), TipoEng)  
Tecnico (CPFFunc (FK), GrauTec)

FUNCIONARIO (instância – parcial/sobreposto)

| <u>CPFFunc</u> | nomeFunc |
|----------------|----------|
| 111111111111   | Carlos   |
| 222222222222   | Ana      |
| 333333333333   | Miguel   |
| 444444444444   | Pedro    |
| 555555555555   | Lucas    |
| 666666666666   | Antônio  |

SECRETARIO

| <u>CPFFunc</u> | idiomaSec |
|----------------|-----------|
| 111111111111   | P         |
| 555555555555   | P         |

ENGENHEIRO

| <u>CPFFunc</u> | tipoEng     |
|----------------|-------------|
| 222222222222   | Pesquisador |
| 555555555555   | Operador    |

TÉCNICO

| <u>CPFFunc</u> | grauTec       |
|----------------|---------------|
| 333333333333   | Superior      |
| 444444444444   | Intermediário |

- Essa opção funciona para qualquer tipo de especialização (total ou parcial, com restrições de disjunção ou de sobreposição)



Realize o mapeamento do modelo entidade relacionamento estendido para o modelo relacional de forma otimizada. Especifique todas as chaves primárias e estrangeiras. Exemplo: considere a relação R com os atributos a1, a2, a3 um esquema de uma relação deve ser definido como: R(a1, a2, a3) onde a1 é o atributo a chave da relação.

