

**INSTITUTO FEDERAL**

Goiano

Campus Morrinhos

# Banco de Dados Mapeamento DER para Relacional

Professor Marcel Melo

[marcel.melo@ifgoiano.edu.br](mailto:marcel.melo@ifgoiano.edu.br)

Entidades

# Passo 01: Mapeamento entidades regulares

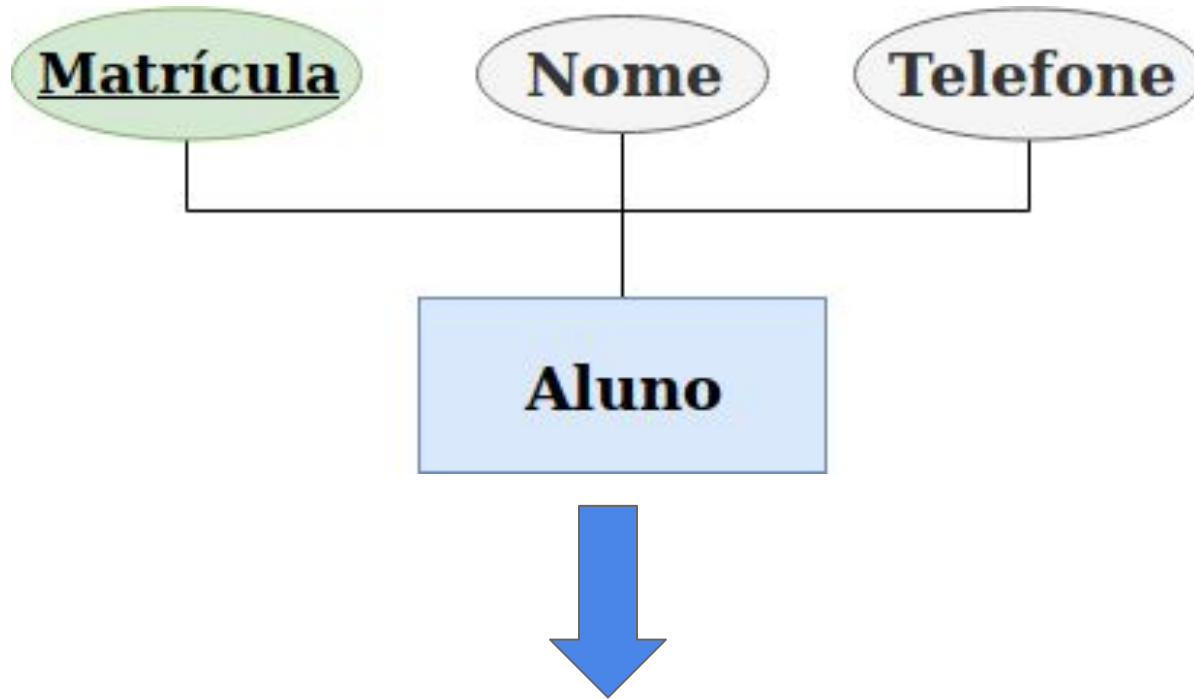
**Toda entidade E** presente no diagrama entidade-relacionamento **se torna uma tabela R** no modelo relacional;

Cada **atributo simples** da Entidade E se torna **uma coluna** da tabela R;

Escolher um **atributo-chave de E** para **chave primária de R**;

- Caso a chave escolhida de E for um atributo composto, os atributos simples do atributo composto vão, juntos, formar a chave primária de R

# Passo 01: Mapeamento entidades regulares



Aluno (Matrícula, Nome, Telefone)

# Atributos

# Passo 1.1: Atributos compostos

Atributos compostos e multivalorados podem ser transformados de duas formas, dependendo do contexto:

- Criar uma nova tabela **R1** que levará o nome do atributo composto e adicionar os atributos simples como colunas dessa tabela;
  - Escolher um **atributo-chave** para **chave primária de R1**;
  - Adicionar uma chave estrangeira FK na tabela R1 onde originalmente estava ligado o atributo composto
- **Atributos simples dos atributos compostos** da Entidade **E** tornam-se atributos da tabela **R**

## Passo 1.2: Atributos multivalorados

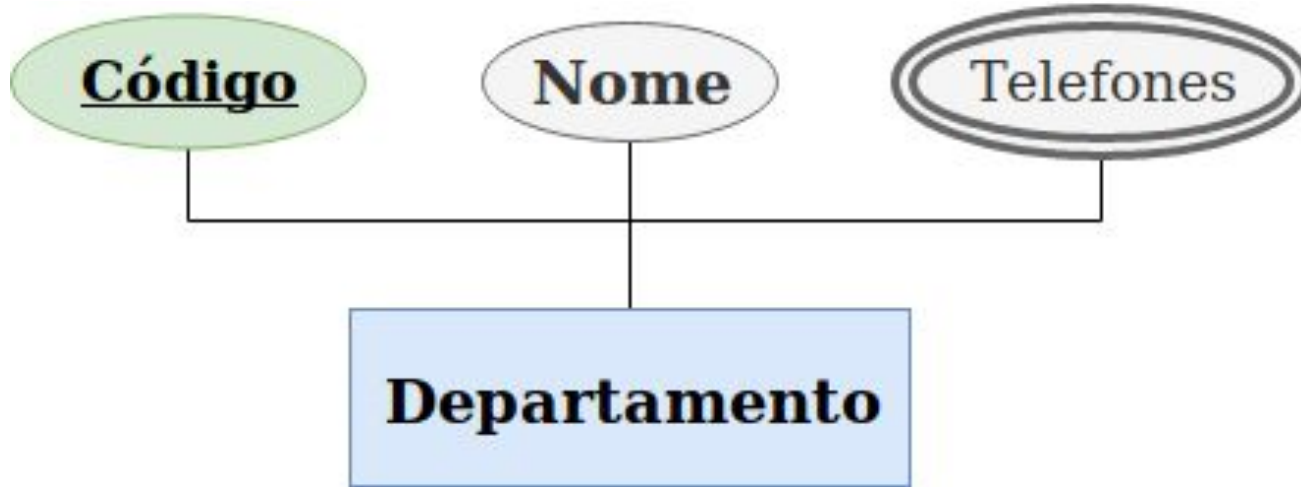
### **Modelo relacional não permite atributos multivalorados;**

Para cada atributo multivalorado A, criar uma tabela R1 que contém o atributo A mas a chave primária da tabela R, como **chave estrangeira**.

A chave primária de R é a combinação do atributo A + chave primária de E.

- Se o atributo multivalorado for também composto, todos atributos devem pertencer a tabela R1, mas somente alguns devem fazer parte da chave primária.

## Passo 1.2: Atributos multivalorados



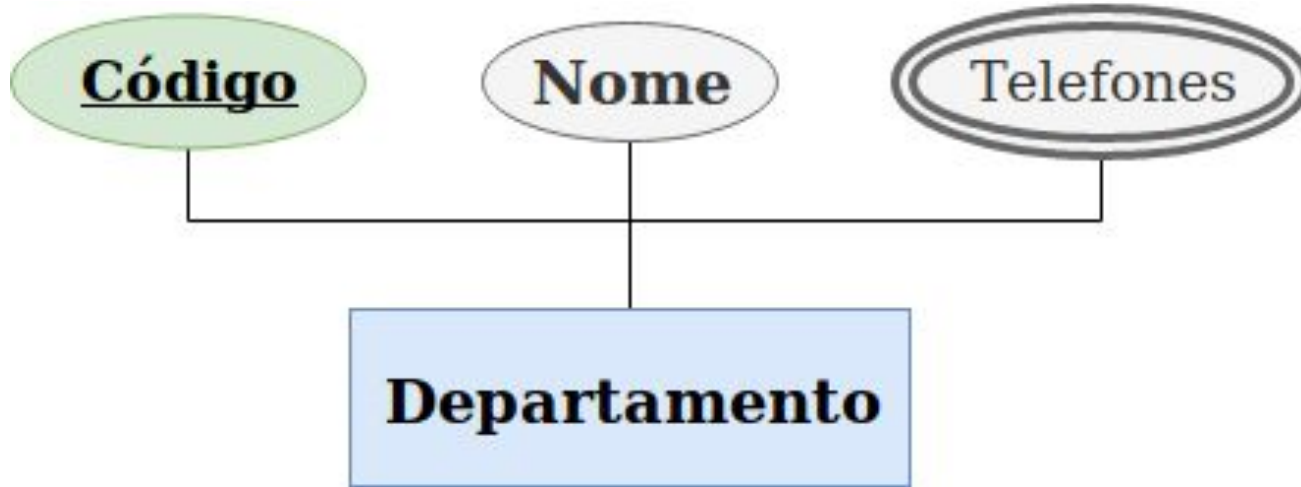
Departamento(Código, Nome)

Telefones(CódigoDepto#, nroTelefone)

CódigoDepto referencia Departamento(Código)

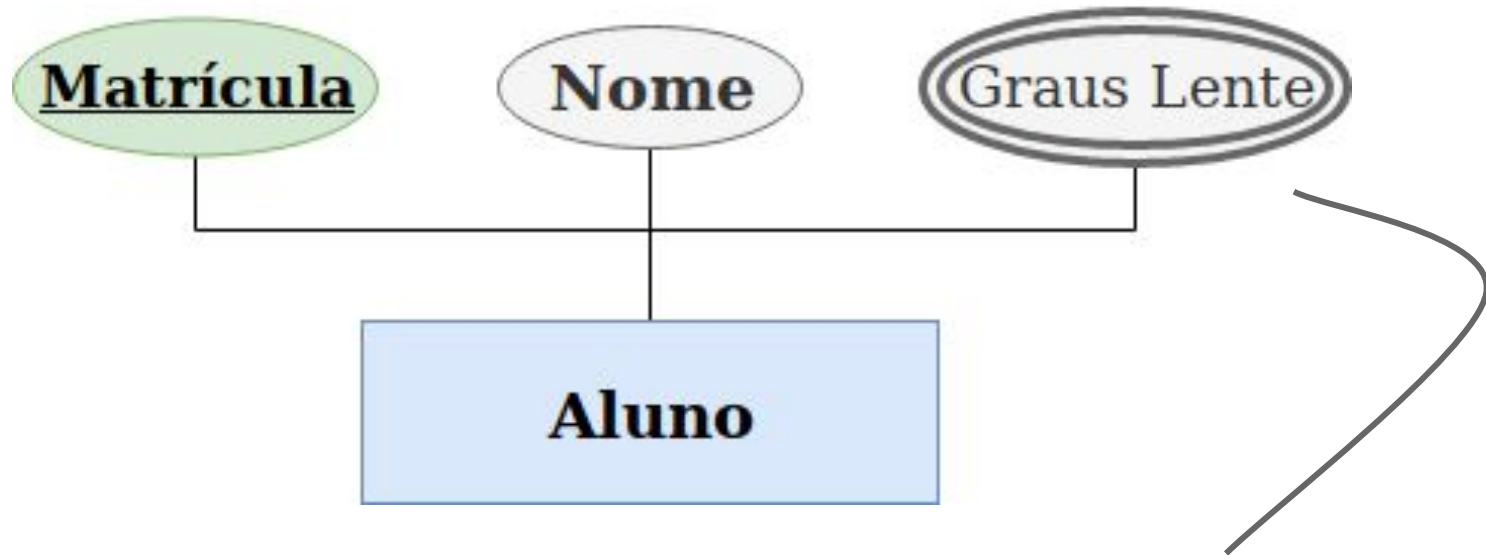


## Passo 1.2: Atributos multivalorados



Departamento(Código, Nome, TelefoneFixo, TelefoneCelular, TelefoneComercial)

## Passo 1.2: Atributos multivalorados



Podemos decompor o atributo em duas partes:

- Grau olho direito
- Grau olho esquerdo

**Aluno**(matrícula, nome, grau\_direito, grau\_esquerdo)

# Relacionamentos

# Passo 3: Mapeamento tipo-relacionamento (1:1)

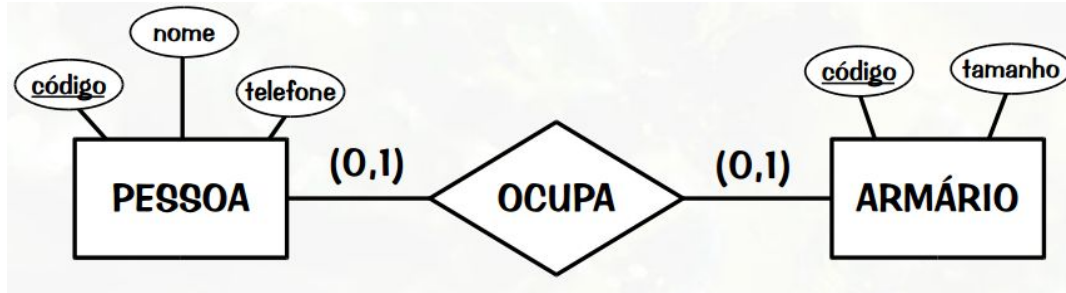
## Modelo Entidade-Relacionamento

**Tipo-relacionamento Binário:** E1 relacionando-se com E2

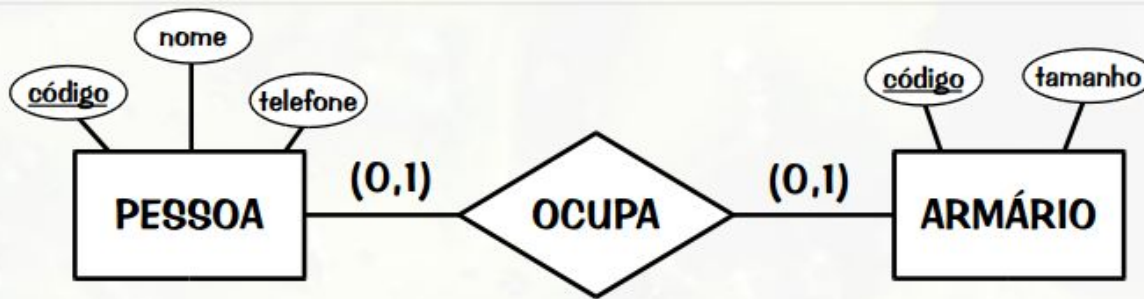
**Cardinalidade:** 1:1 com cardinalidade mínima de 0 em ambas as relações

## Modelo Relacional

- **Chave Estrangeira:** Escolher uma Relação (R1) e nela inserir, como chave estrangeira, a chave primária da outra Relação (R2).  
A opção mais utilizada. Deve ser utilizada a não ser em casos especiais



# Passo 3: Mapeamento tipo-relacionamento (1:1)



PESSOA(Código, Nome, Telefone)

ARMÁRIO(Código, Tamanho)

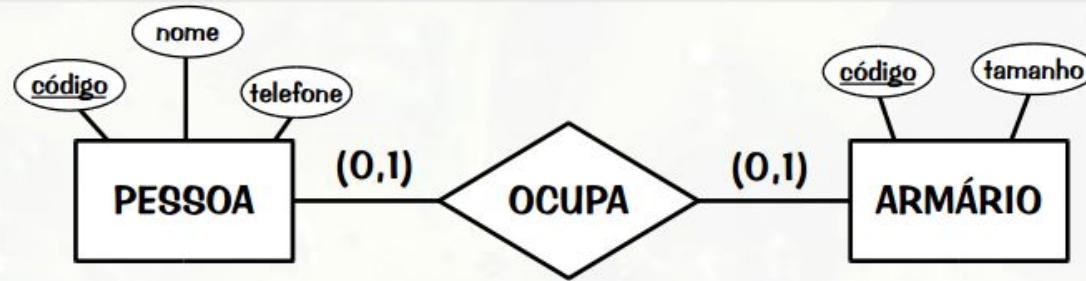
**PESSOA**

<u>Código</u>	Nome	Telefone
1525	Asdrúbal	5432-1098
1637	Doriana	9876-5432
1701	Quincas	8765-4321
2042	Melissa	7654-3210
2111	Horácio	6543-2109

**ARMÁRIO**

<u>Código</u>	Tamanho
1A	simples
2A	duplo
1B	simples
2B	duplo

# Passo 3: Mapeamento tipo-relacionamento (1:1)



PESSOA (Código, Nome, Telefone)

ARMÁRIO (Código, Tamanho, **Ocupante**)

**PESSOA**

<u>Código</u>	Nome	Telefone
1525	Asdrúbal	5432-1098
1637	Doriana	9876-5432
1701	Quincas	8765-4321
2042	Melissa	7654-3210
2111	Horácio	6543-2109

**ARMÁRIO**

<u>Código</u>	Tamanho	<b>Ocupante</b>
1A	simples	<b>1637</b>
2A	duplo	<b>(nulo)</b>
1B	simples	<b>(nulo)</b>
2B	duplo	<b>2111</b>

# Passo 3: Mapeamento tipo-relacionamento (1:1)

## Modelo Entidade-Relacionamento

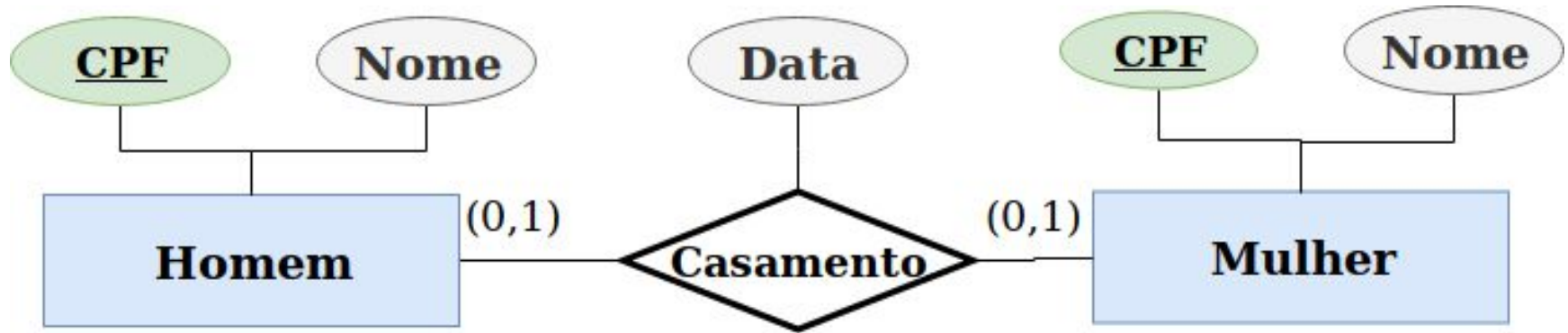
**Tipo-relacionamento Binário:** E1 relacionando-se com E2

**Cardinalidade:** 1:1 com cardinalidade mínima de 0 em ambas as relações

## Modelo Relacional

- **Tabela adicional para o relacionamento:** Cria-se uma nova tabela R para o relacionamento.
  - A chave primária desta tabela R é composta pelas chaves-primárias da Tabela E1 e E2 (adicionadas em R também como chaves-estrangeiras).

## Passo 3: Mapeamento tipo-relacionamento (1:1)



**Homem**(CPF, Nome)

**Mulher**(CPF, Nome)

**Casamento**(CPF<sub>H</sub>, CPF<sub>M</sub>, Data)



# Passo 3: Mapeamento tipo-relacionamento (1:1)

## Modelo Entidade-Relacionamento

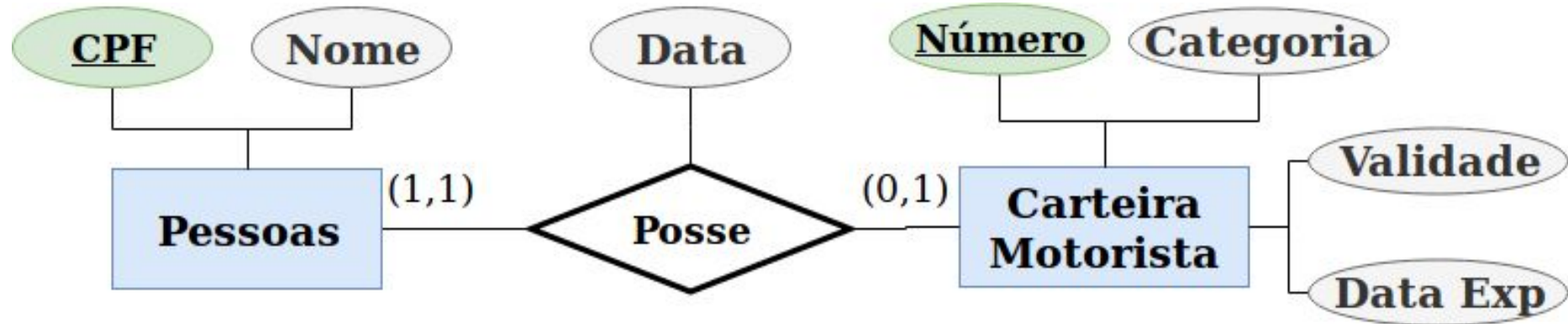
**Tipo-relacionamento Binário:** E1 relacionando-se com E2

**Cardinalidade:** 1:1 com cardinalidade mínima de 1 em uma ou ambas as relações

## Modelo Relacional

- Funde-se a ambas tabelas criando apenas uma com os atributos de ambas as tabelas.
- Quando apenas um dos lados possui cardinalidade obrigatória (mínimo de 1), mantém a tabela com cardinalidade mínima 1.

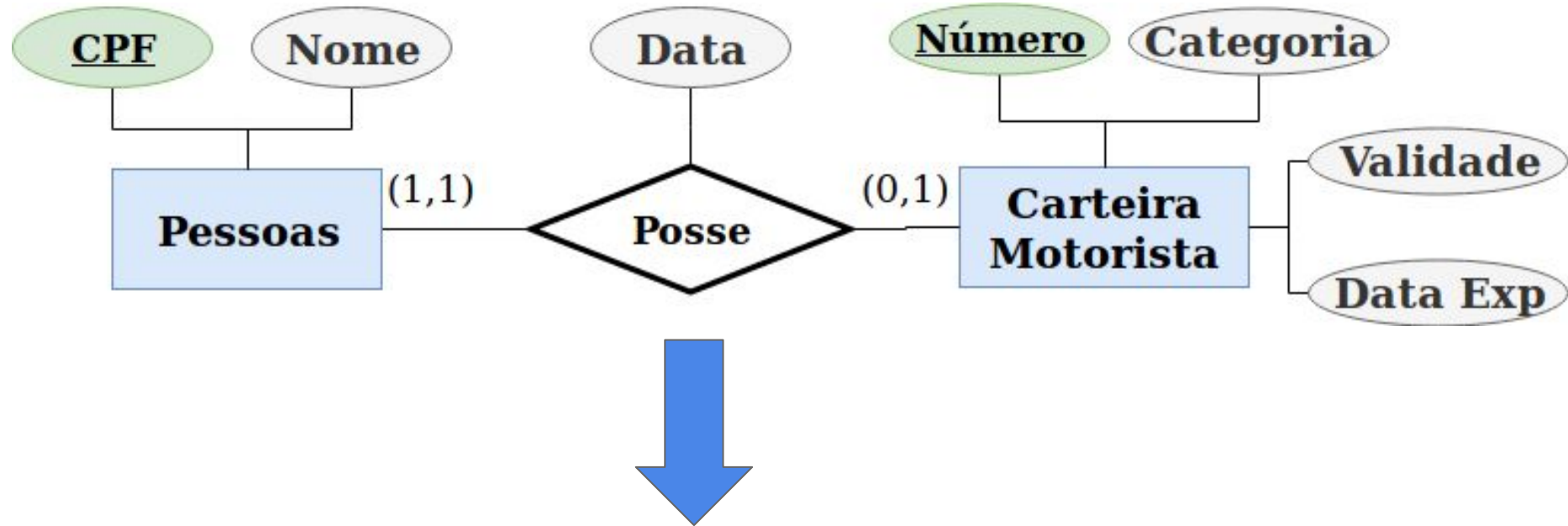
## Passo 3: Mapeamento tipo-relacionamento (1:1)



**Pessoas**(CPF, Nome)

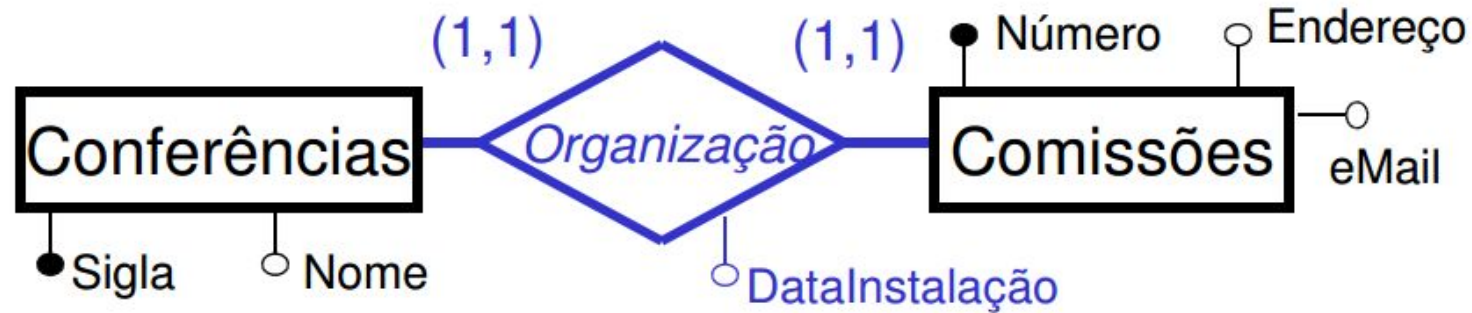
**CarteiraMotorista**(Número, Categoria, Validade, Data\_Expedicao, CPF\_Motorista, DataPosse)

## Passo 3: Mapeamento tipo-relacionamento (1:1)



**Pessoas**(CPF, Nome, NúmeroCarteira, CategoriaCarteira, ValidadeCarteira, Data\_Expedicao\_Carteira, DataPosse)

## Passo 3: Mapeamento tipo-relacionamento (1:1)



Conferências (Sigla, Nome, DataInstCom, NroCom, EndereçoCom, eMailCom)

# Passo 4: Mapeamento tipo-relacionamento (1:N)

## Modelo Entidade-Relacionamento

**Tipo-relacionamento Binário:** E1 relacionando-se com E2

**Cardinalidade:** 1:N

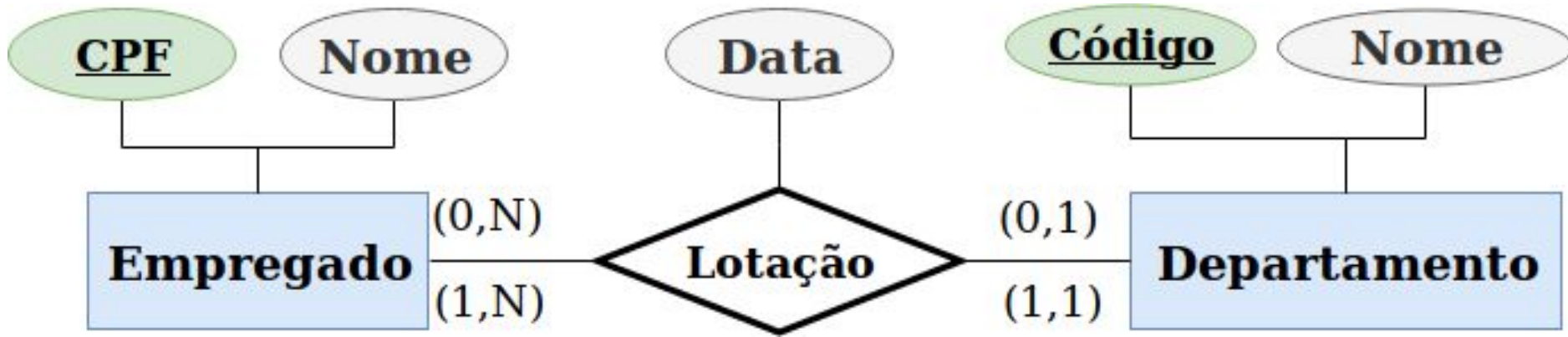
**E1** - Lado (0,1) ou (1,1)

**E2** - Lado (0,N) ou (1,N)

## Modelo Relacional

- **A relação R1** terá somente os **atributos de E1**
- **A relação R2** terá:
  - os atributos de R2
  - a chave-primária de E1 como **chave estrangeira**
  - os atributos do relacionamento

## Passo 4: Mapeamento tipo-relacionamento (1:N)



Departamento(Código, Nome)

Empregado(CPF, Nome, CódigoDept, DataLotação)

# Passo 4: Mapeamento tipo-relacionamento (1:N)

## Modelo Entidade-Relacionamento

**Tipo-relacionamento Binário:** E1 relacionando-se com E2

**Cardinalidade:** 1:N

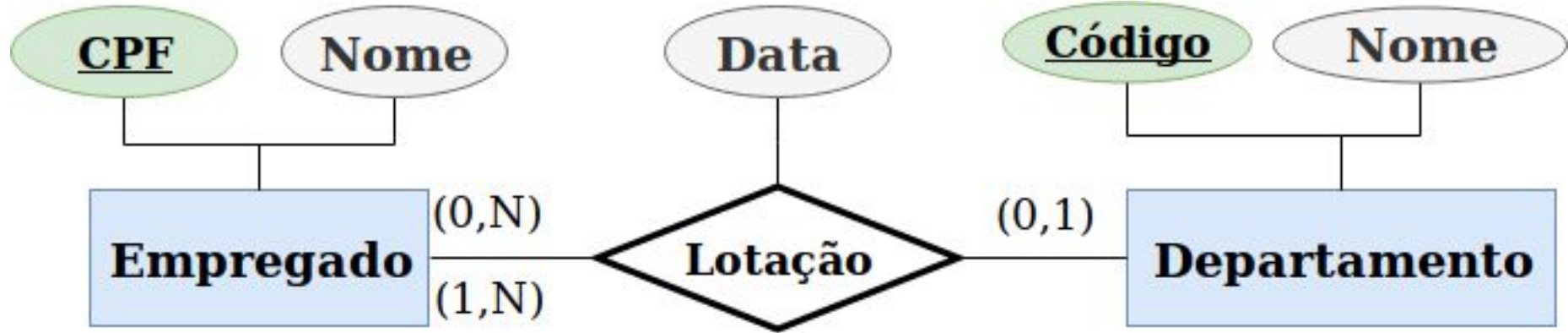
**E1 - Lado (0,1)**

**E2 - Lado (0,N) ou (1,N)**

## Modelo Relacional

Se na Entidade E1 o relacionamento for opcional, cardinalidades (0,1) , é possível também criar uma nova tabela T1 para representar o relacionamento.

## Passo 4: Mapeamento tipo-relacionamento (1:N)



Departamento(Código, Nome)

Empregado(CPF, Nome)

Lotação(Código Dept, CPF Empregado, Data)



# Passo 5: Mapeamento tipo-relacionamento (N:M)

## Modelo Entidade-Relacionamento

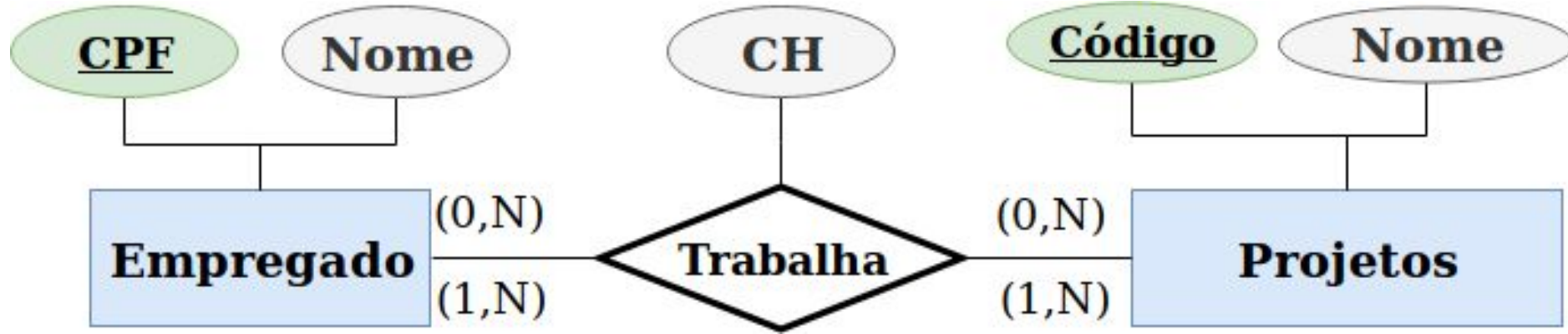
**Tipo-relacionamento Binário:** E1 relacionando-se com E2

**Cardinalidade:** N:M Ambos os lados possui cardinalidade máxima N

## Modelo Relacional

- **A relação R1 terá apenas os atributos de E1**
- **A relação R2 terá apenas os atributos de E2**
- **Criar uma nova tabela R (relativa ao tipo-relacionamento) contendo:**
  - **A chave-primária de E1, como chave estrangeira**
  - **A chave-primária de E2, como chave estrangeira**
  - **os atributos do tipo-relacionamento, caso existir.**
  - **Chave primária: Chave-primária de E1 + Chave-primária E2**

## Passo 5: Mapeamento tipo-relacionamento (N:M)



Empregado(CPF, Nome)

Projetos(Código, Nome)

Trabalha(CPF\_Empregado, Cod\_Projeto, CH)

## passo 06: Mapeamento Entidade-fraca

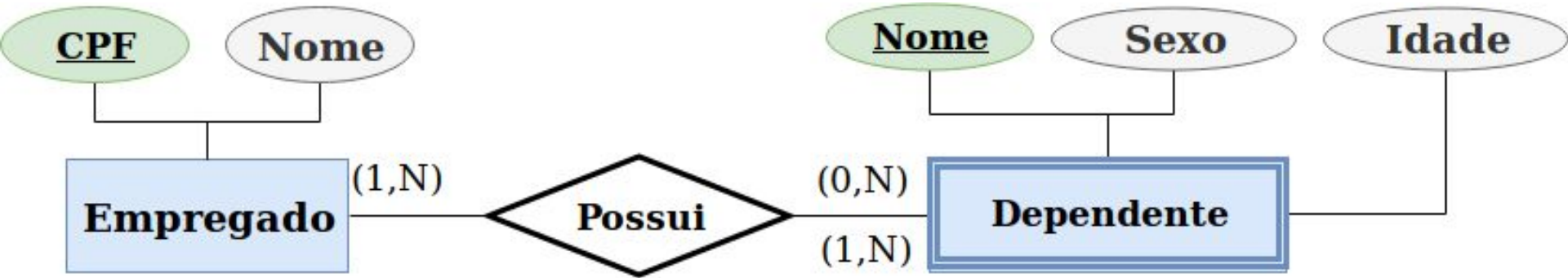
### Modelo Entidade Relacionamento

- **Tipo-Entidade Forte E**: chave-primária **PK**
- **Tipo-Entidade Fraca A**: atributos **a1, a2, a3, ..., an**

Para cada **tipo-entidade Fraca A**, criar uma **Tabela R1** que inclua **todos atributos simples** do tipo-entidade **Fraca A** (inclusive os atributos simples de atributos compostos) mais as **chaves primárias PK** do **tipo-entidade forte E** a qual a **tipo-entidade Fraca A** está ligada.

A **chave-primária de R1** é a combinação das **chaves primárias do tipo-entidade Fraca A** + a **chave primária PK do tipo-Entidade Forte E**

## passo 06: Mapeamento Entidade-fracca



Empregado(CPF, Nome)

Dependente(CPF\_Empregado, Nome, Sexo, Idade)

# Resumo



# Resumo das regras

- **Passo 01:** Entidade (normal, associativa, fraca) se torna uma tabela
- **Passo 02:** Cada atributo se torna uma coluna na tabela correspondente
- **Passo 03:** Em Relacionamentos 1:N, a Entidade do lado N recebe a chave-primária da Entidade do lado 1, como chave-estrangeira;
- **Passo 04:** Em Relacionamentos N:N, cria-se uma nova Tabela com as chaves primárias das entidades relacionadas
- **Passo 05:** Em Relacionamentos 1:1, há duas alternativas:
  1. Escolher uma como “entidade-fraca”, que recebe a chave-primária da outra como chave estrangeira
  2. Os atributos da “entidade-fraca” são adicionados na outra Entidade.
- **Passo 06:** Em caso de entidade fraca, adiciona-se a chave-primária da entidade forte como chave primária e chave estrangeira da entidade fraca

# Mapeamento de Relacionamentos 1:1


Tipo de relacionamento	Regra de implementação		
	Tabela própria	Adição de coluna	Fusão de tabelas
	2ª Opção	✓	✗
	3ª Opção	2ª Opção	✓
	3ª Opção	2ª Opção	✓

# Mapeamento de Relacionamentos 1:N

Tipo de relacionamento	Regra de implementação		
	Tabela própria	Adição de coluna	Fusão de tabelas
$(0,1)$  $(\_,n)$	2ª Opção	✓	✗
$(1,1)$  $(\_,n)$	✗	✓	✗



# Mapeamento de Relacionamentos N:N

Tipo de relacionamento	Regra de implementação		
	Tabela própria	Adição de coluna	Fusão de tabelas
	✓	✗	✗

# Relacionamento ternário

# Mapeamentos de relacionamento ternário

## Modelo Entidade-Relacionamento (Ternário)

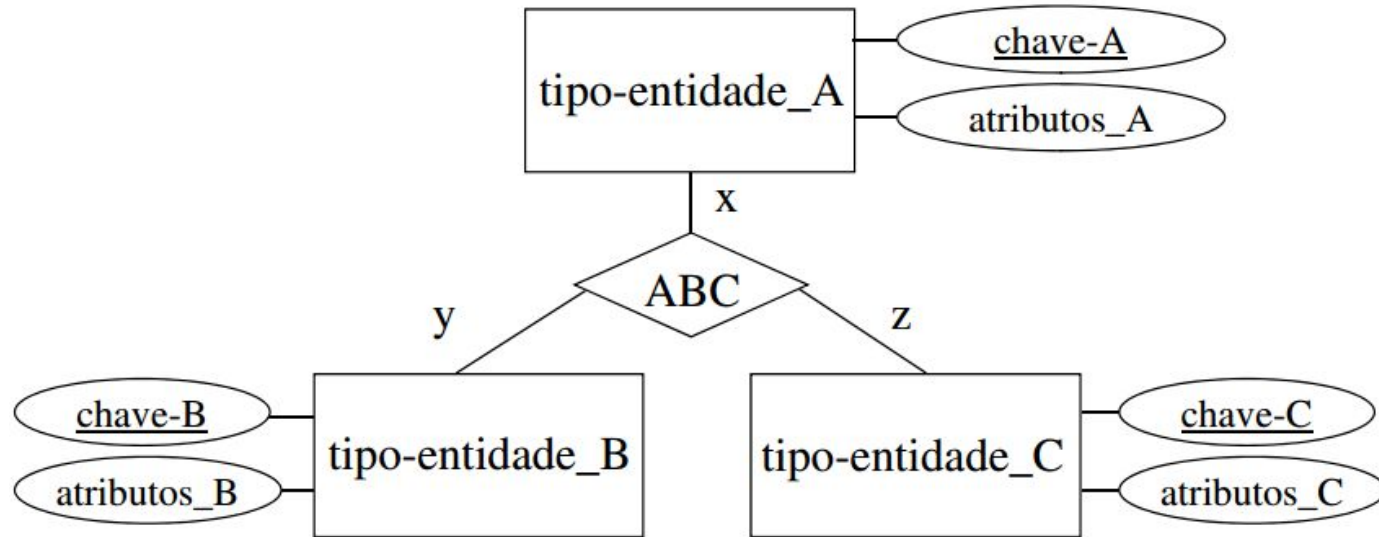
**Tipo-relacionamento** n-ário: E1 relacionando-se com E2 e E3

**Cardinalidade:** m:n:p

## Modelo Relacional

- A tabela T1 possuirá apenas os atributos de E1
- A tabela T2 possuirá apenas os atributos de E2
- A tabela T3 possuirá apenas os atributos de E3
- Criar uma nova relação R (relativa ao relacionamento) contendo:
  - A chave primária de cada relação envolvida no relacionamento
  - Os atributos do tipo-relacionamento

# Mapeamentos de relacionamento ternário



tabelas relativas  
aos tipos-entidade

tipo-entidade\_A (chave-A, atributos\_A)  
tipo-entidade\_B (chave-B, atributos\_B)  
tipo-entidade\_C (chave-C, atributos\_C)

# Mapeamentos de relacionamento ternário

## – Primeiro caso:

$$x = y = z = 1$$

◆ ABC (chaves-A, chaves-B, chaves-C)

◆ ABC (chaves-A, chaves-B, chaves-C)

◆ ABC (chaves-A, chaves-B, chaves-C)

– chaves-A, chaves-B  $\longrightarrow$  chaves-C

– chaves-A, chaves-C  $\longrightarrow$  chaves-B

– chaves-B, chaves-C  $\longrightarrow$  chaves-A

+ integridade  
referencial

## – Segundo caso:

$$x = m; y = n; z = p$$

◆ ABC (chaves-A, chaves-B, chaves-C)

+ integridade  
referencial

# Mapeamentos de relacionamento ternário

– Terceiro caso:

$$x = 1; y = 1; z = m$$

◆ ABC (chaves-A, chaves-B, chaves-C)

◆ ABC (chaves-A, chaves-B, chaves-C)

– chaves-A, chaves-C  $\longrightarrow$  chaves-B

– chaves-B, chaves-C  $\longrightarrow$  chaves-A

+ integridade  
referencial

– Terceiro caso:

$$x = 1; y = m; z = n$$

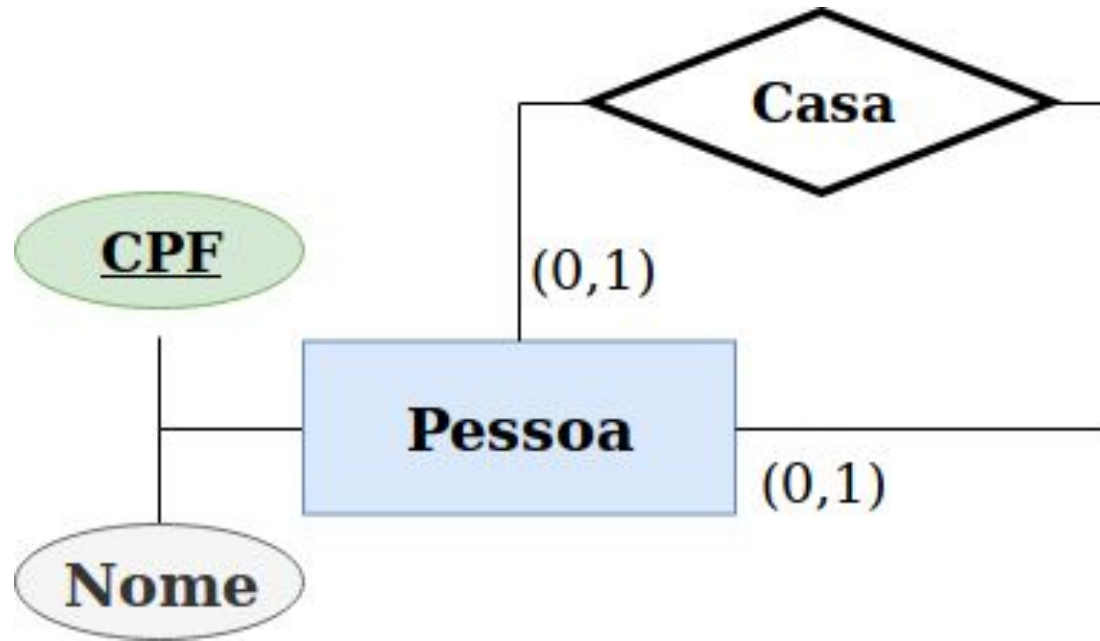
◆ ABC (chaves-A, chaves-B, chaves-C)

– chaves-B, chaves-C  $\longrightarrow$  chaves-A

+ integridade  
referencial

Auto-relacionamento

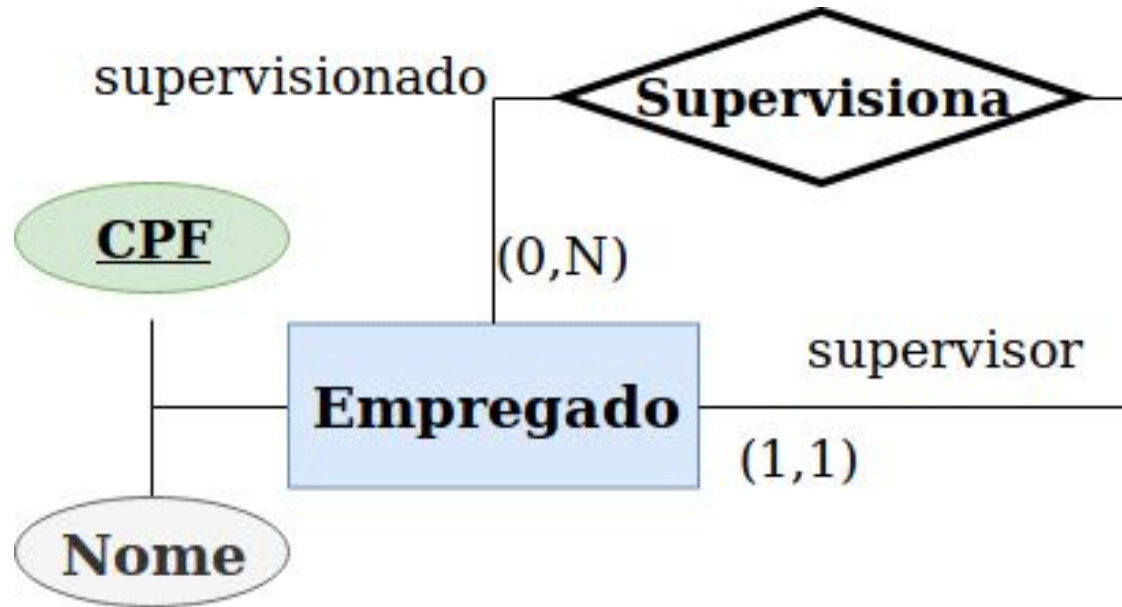
## Passo 8: Auto Relacionamento (Unário) (1:1)



**Pessoa**(CPF, Nome, CPF\_conjuge)

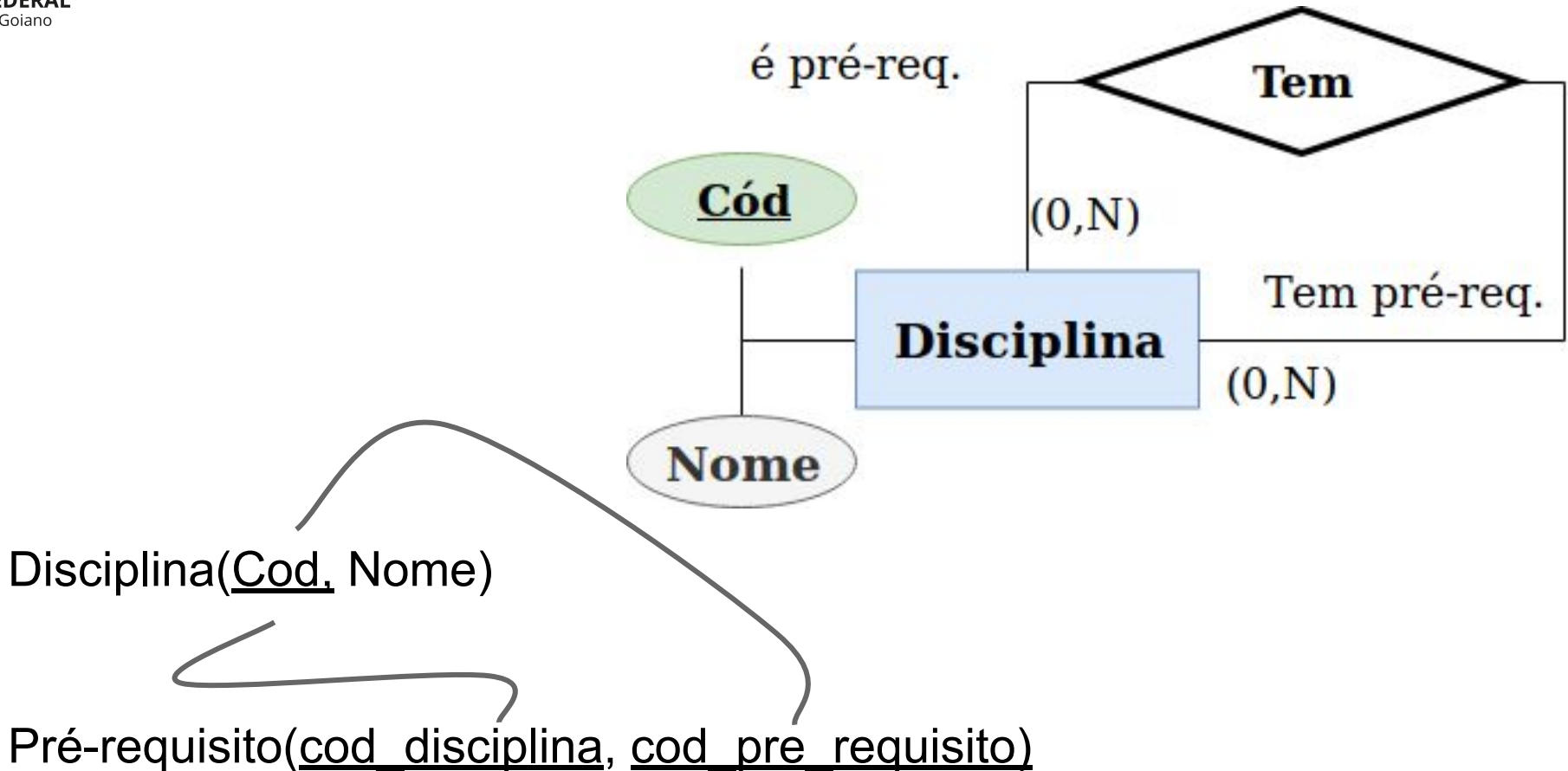


## Passo 8: Auto Relacionamento (Unário) (1:N)



Empregado(CPF, Nome, cpf\_supervisor)

## Passo 8: Auto Relacionamento (Unário) (N:N)



Generalização / Especialização

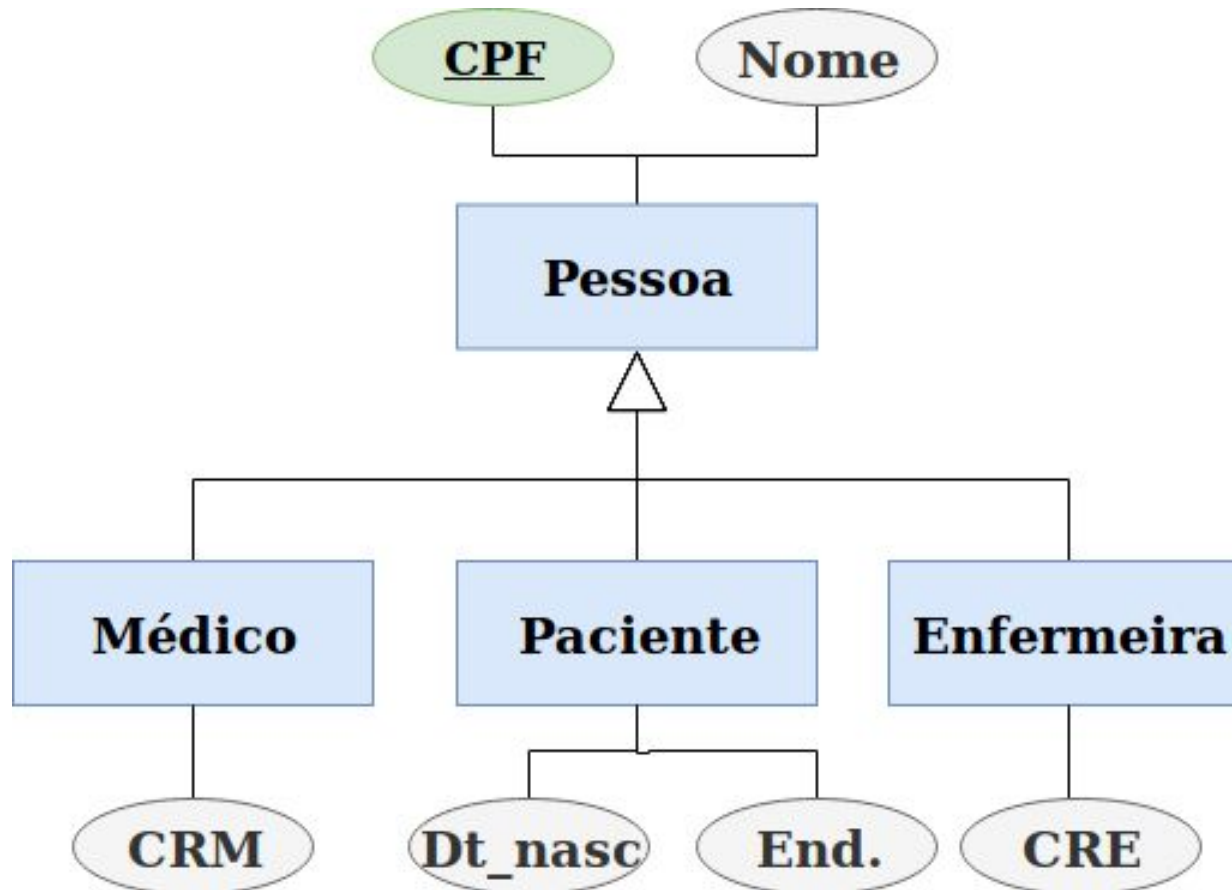
## Modelo Entidade Relacionamento

- **E1**: Entidade Genérica (superclasse)
- **E2...En**: Entidade Especializada (subclasses) de E1

## Modelo Relacional

- Criar uma tabela T1 a partir da Entidade genérica E1 contendo somente os atributos de E1;
- Criar uma tabela T para cada Entidade E especializada da entidade genérica contendo:
  - Os atributos específico de **E**
  - A chave primária da tabela **T1**
  - **Chave primária das tabelas especilizadas é a chave primária de E1**

# Generalização/Especialização



# Generalização/Especialização

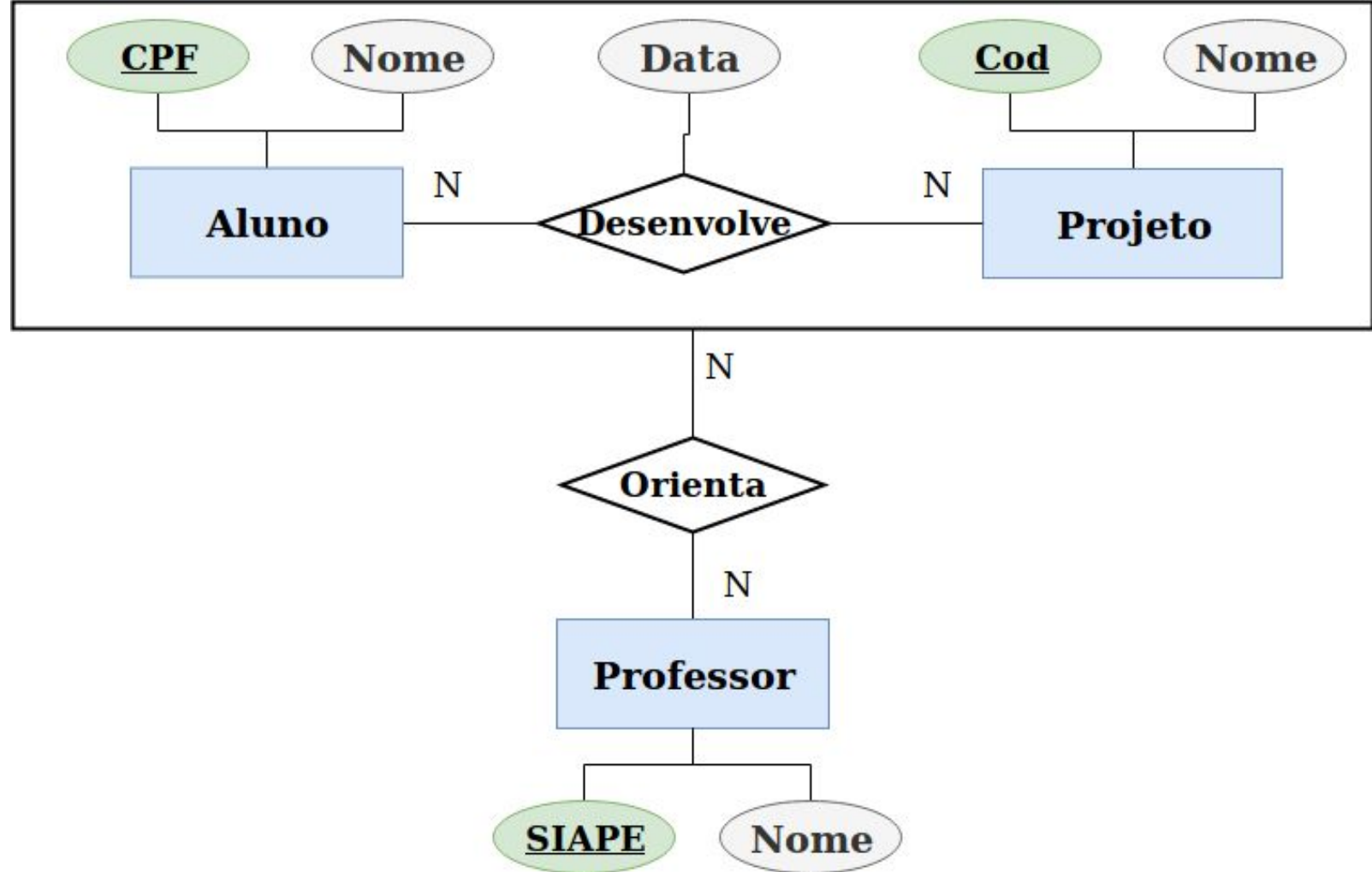
**Pessoa**(CPF, Nome)

**Médico**(CPF, CRM)

**Paciente**(CPF, Dt\_nascimento, Endereco)

**Enfermeira**(CPF, CRE)

Entidade Associativa





# Generalização/Especialização

Aluno(CPF, Nome)

Projeto(Cod, Nome)

Desenvolve(CPF, Cod\_projeto, data)

- CPF referencia Aluno(CPF)
- Cod\_projeto referencia Projeto(Cod)

Orienta(CPF, Cod\_projeto#, SIAPE\_professor#)

CPF, Cod\_projeto referencia Desenvolve(CPF, Cod\_projeto)

SIAPE\_professor referencia Professor(Siape)

Professor(SIAPE, Nome)

# Exercícios

TO Do