

# Mapeamento: Modelo Entidade Relacionamento para o Modelo Relacional

Banco de Dados

Profa. Dra. Cristina Dutra de Aguiar Ciferri

# Modelo Relacional

- ◆ Relação esquema R:
  - utilizada para descrever uma relação
  - denotada por  $R(A_1, A_2, \dots, A_n)$
  - formada por
    - ◆ um nome de relação R
    - ◆ uma lista de atributos  $A_1, A_2, \dots, A_n$
  - para cada atributo  $A_i$  ( $1 \leq i \leq n$ )
    - ◆  $\text{dom}(A_i)$ : domínio de  $A_i$
    - ◆ domínio: conjunto de valores **atômicos**
  - caracteriza a **intenção** do BD

# Modelo Relacional

## ◆ Relação $r$ da relação esquema

$R(A_1, A_2, \dots, A_n)$

– representa a instância da relação

– denotada por  $r(R)$

– formada por um conjunto de  $n$ -tuplas

$r = \{t_1, t_2, \dots, t_m\}$

◆ cada  $n$ -tupla  $t$  é uma lista de  $n$  valores

$t = \langle v_1, v_2, \dots, v_n \rangle$

◆  $v_i$  ( $1 \leq i \leq n$ ) é um elemento de  $\text{dom}(A_i)$  ou um valor nulo (i.e., null)

– caracteriza a **extensão** do BD

# Restrições sobre uma Relação

## ◆ Domínio

- dentro de cada tupla, o valor de cada atributo A deve ser um valor **atômico** de  $\text{dom}(A)$

## ◆ Chave primária

- identifica de forma única cada tupla da relação

## ◆ Valor nulo

- permitido: **null** (default)
- não permitido: **not null**

## ◆ Integridade de entidade

- **nenhum** valor de chave primária pode ser nulo

# Restrições entre duas Relações

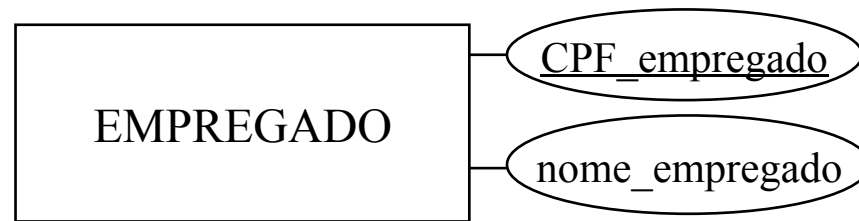
## ♦ Integridade referencial

- mantém a consistência entre as tuplas nas duas relações
- declara que uma tupla em uma relação, a qual faz referência a uma outra relação, deve se referir a uma tupla existente nessa segunda relação
- definida entre a **chave estrangeira** (FK) de uma relação esquema  $R_1$  e a **chave primária** (PK) de uma relação esquema  $R_2$

# Restrições entre duas Relações

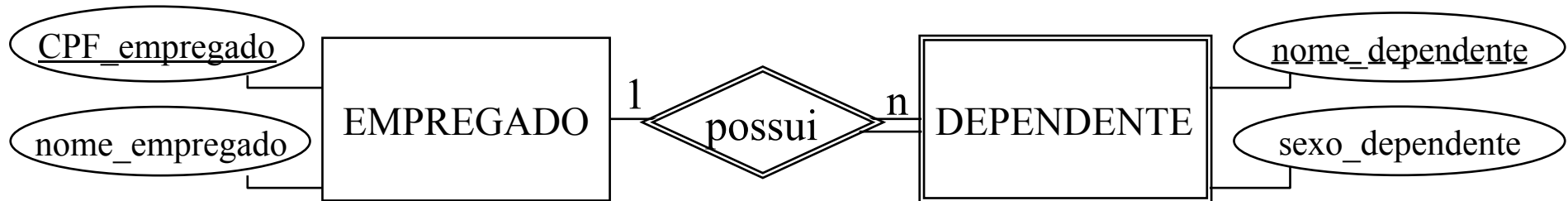
- ◆ FK de  $R_1$  é **chave estrangeira** de  $R_1$ , que faz referência à PK de  $R_2$ , se:
  - os atributos de FK têm os mesmos domínios que os atributos de PK
  - um valor de FK em uma tupla  $t_1$  do estado corrente de  $r_1(R_1)$ 
    - ◆ ocorre como um valor de PK para alguma tupla  $t_2$  no estado corrente  $r_2(R_2)$  *ou*
    - ◆ tem o valor null

# Tipo-Entidade Forte



empregado (CPF\_empregado, nome\_empregado)

# Tipo-Entidade Fraca

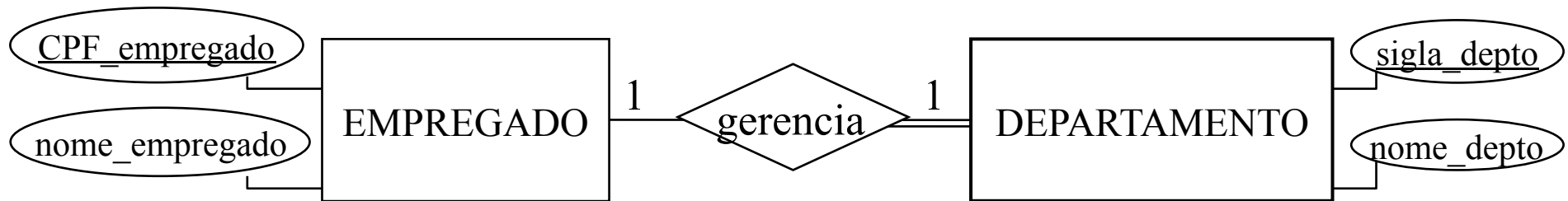


empregado (CPF\_empregado, nome\_empregado)

dependente (CPF\_empregado, nome\_dependente, sexo\_dependente)



# Tipo-Relacionamento (1:1)

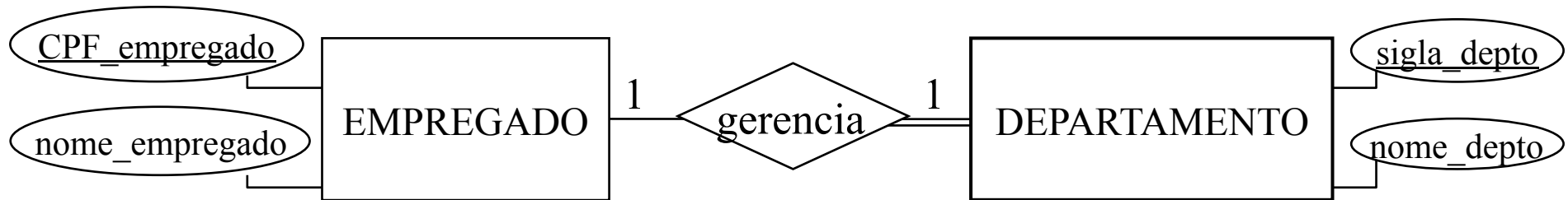


empregado (CPF empregado, nome\_empregado, sigla\_depto)  
departamento (sigla\_depto, nome\_depto, CPF\_empregado)

empregado (CPF empregado, nome\_empregado)  
departamento (sigla\_depto, nome\_depto, CPF\_empregado)

empregado (CPF empregado, nome\_empregado, sigla\_depto)  
departamento (sigla\_depto, nome\_depto)

# Tipo-Relacionamento (1:1)

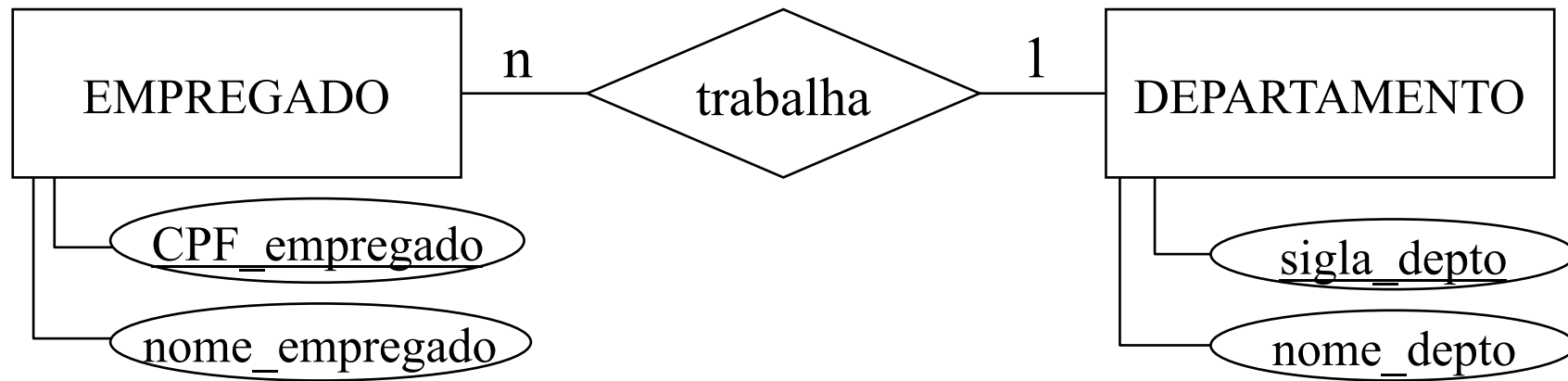


- não pode existir departamento sem gerente
- pode existir empregado que não gerencia o departamento

empregado (CPF empregado, nome\_empregado)  
departamento (sigla\_depto, nome\_depto, CPF\_empregado)

- entidades de departamento: participação total
- entidades de empregado: participação parcial

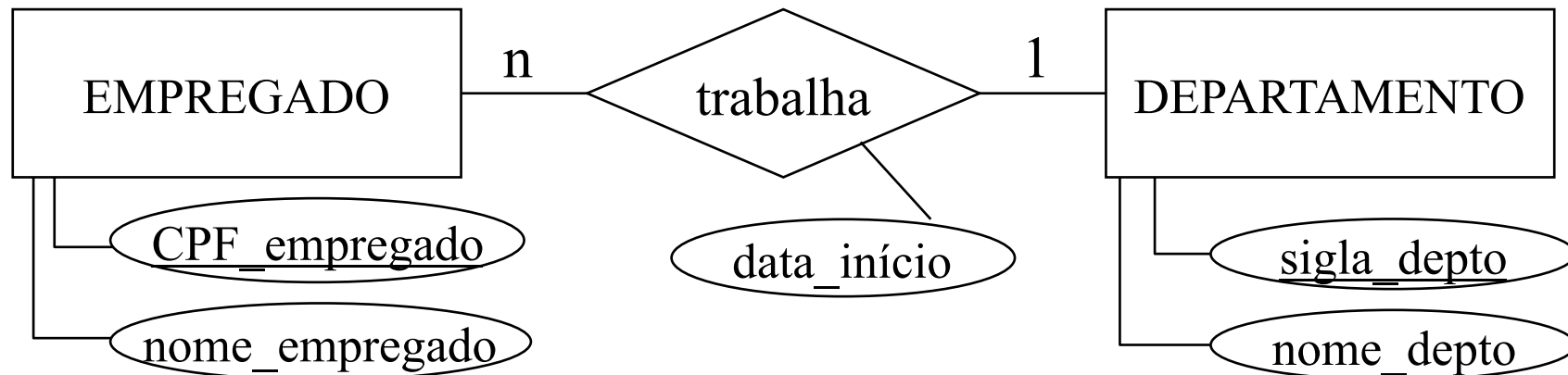
# Tipo-Relacionamento (1:n)



empregado (CPF\_empregado, nome\_empregado, sigla\_depto)

departamento (sigla\_depto, nome\_depto)

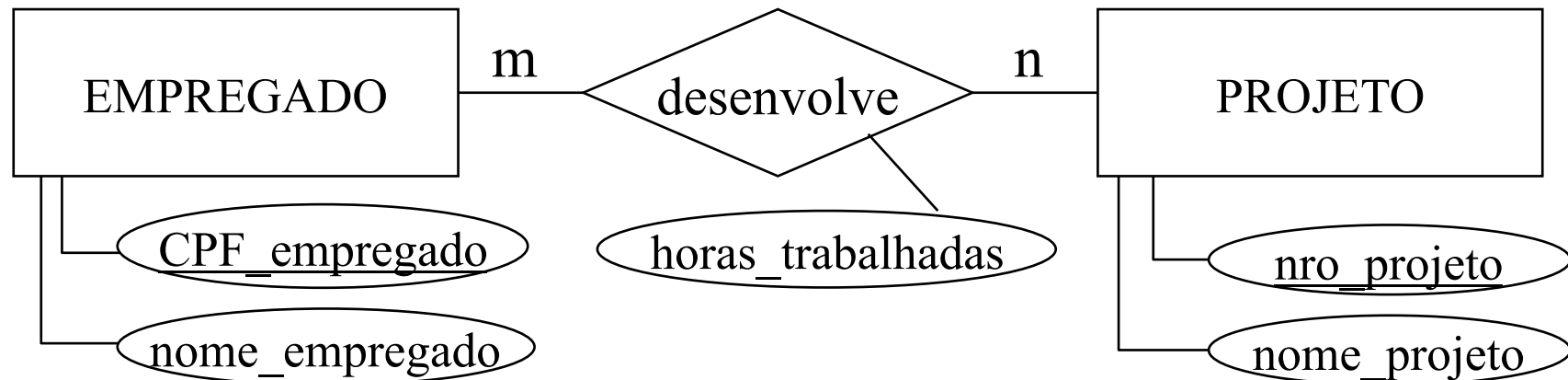
# Atributo de Tipo-Relacionamento (1:1 e 1:n)



empregado (CPF\_empregado, nome\_empregado, sigla\_depto, data\_início)

departamento (sigla\_depto, nome\_depto)

# Tipo-Relacionamento (m:n)

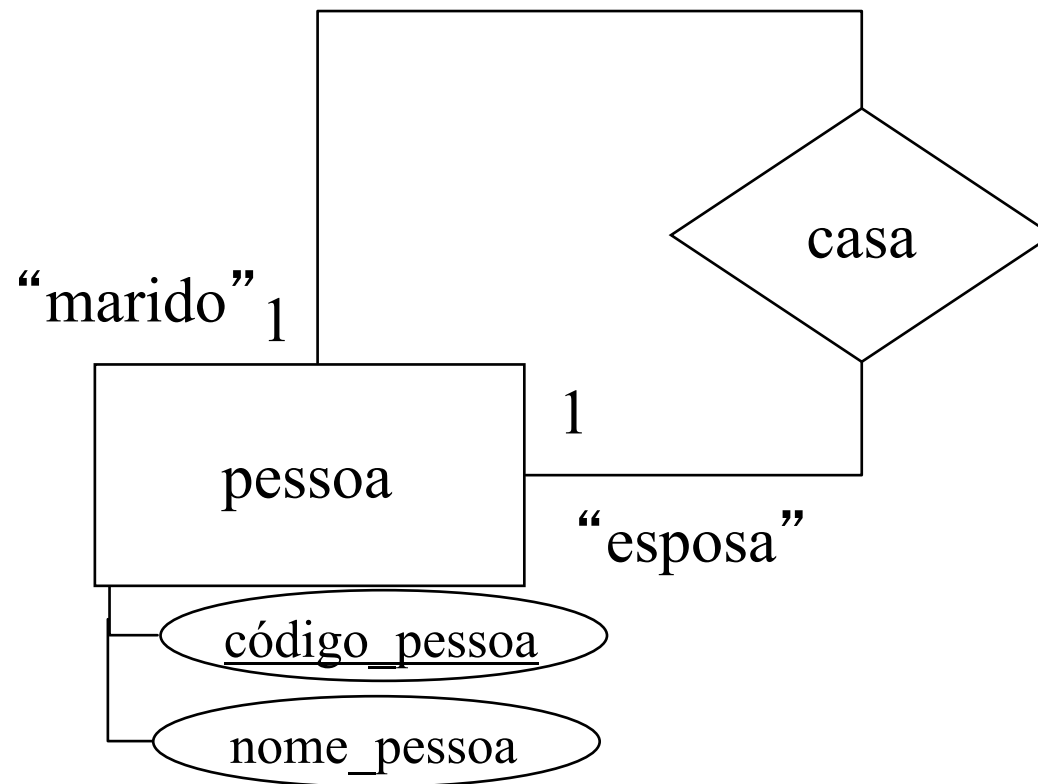


empregado (CPF\_empregado, nome\_empregado)

projeto (nro\_projeto, nome\_projeto)

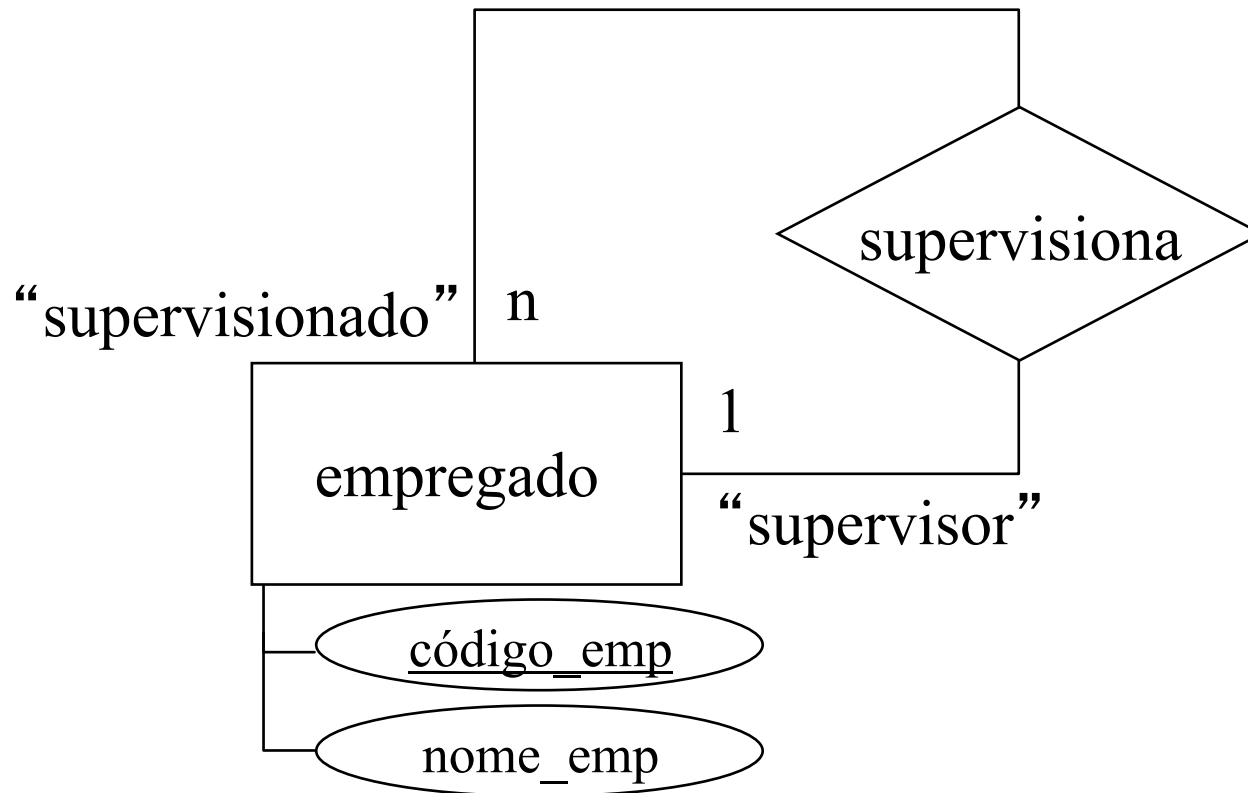
desenvolve (CPF\_empregado, nro\_projeto, horas\_trabalhadas)

# Tipo-relacionamento Unário (1:1)



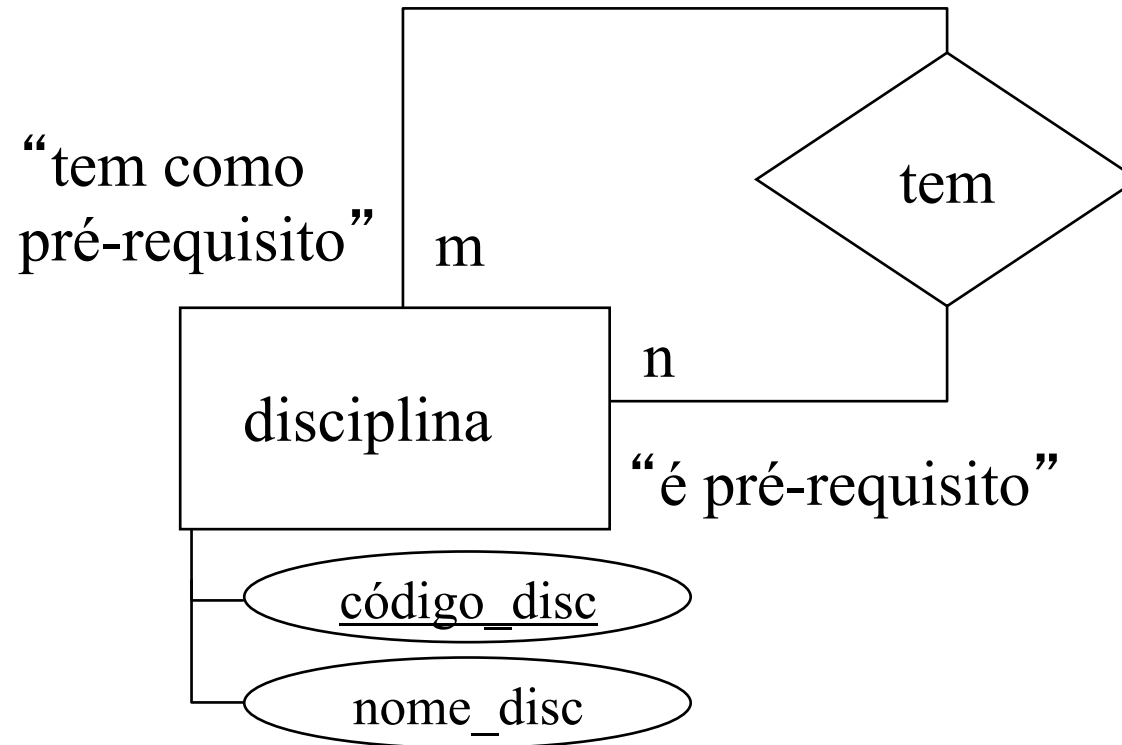
pessoa (código\_pessoa, nome\_pessoa, código\_cônjuge)

# Tipo-relacionamento Unário (1:n)



empregado (código\_emp, nome\_emp, código\_supervisor)

# Tipo-relacionamento Unário (m:n)

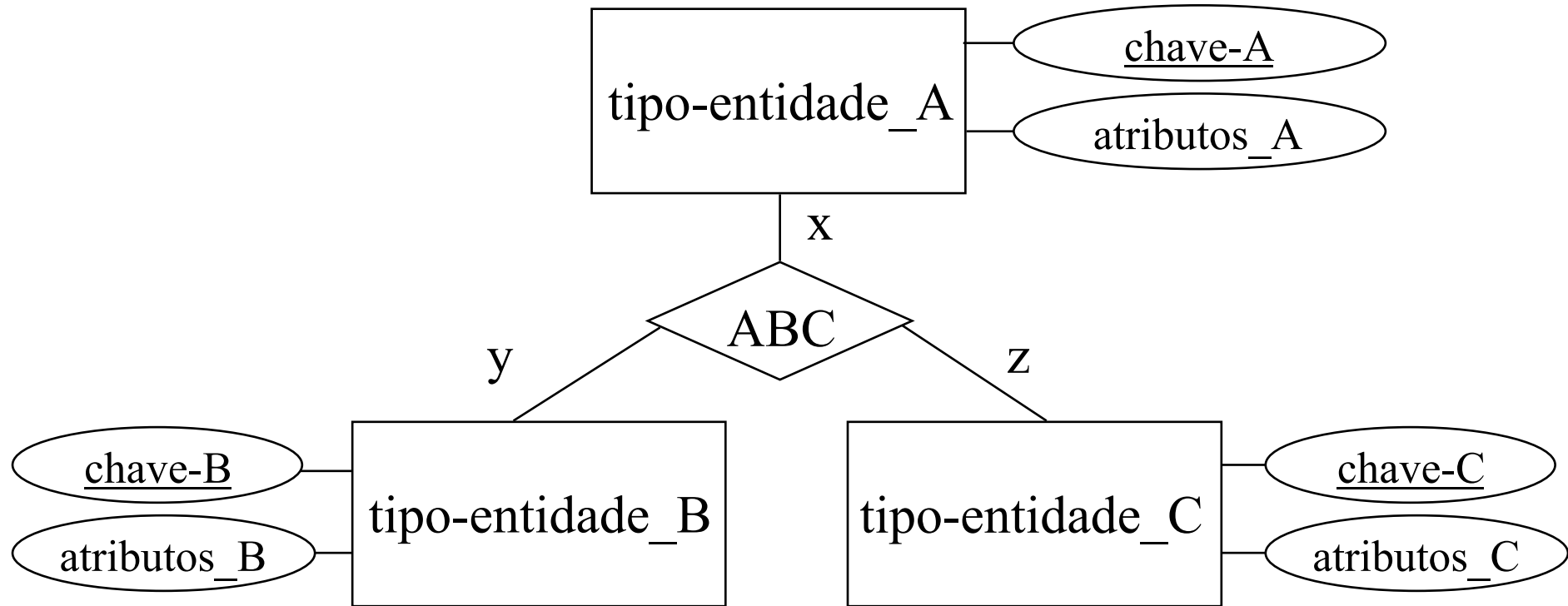


disciplina (código\_disc, nome\_disc)

pré\_requisito (código\_disc, código pré requisito)



# Tipo-relacionamento Ternário

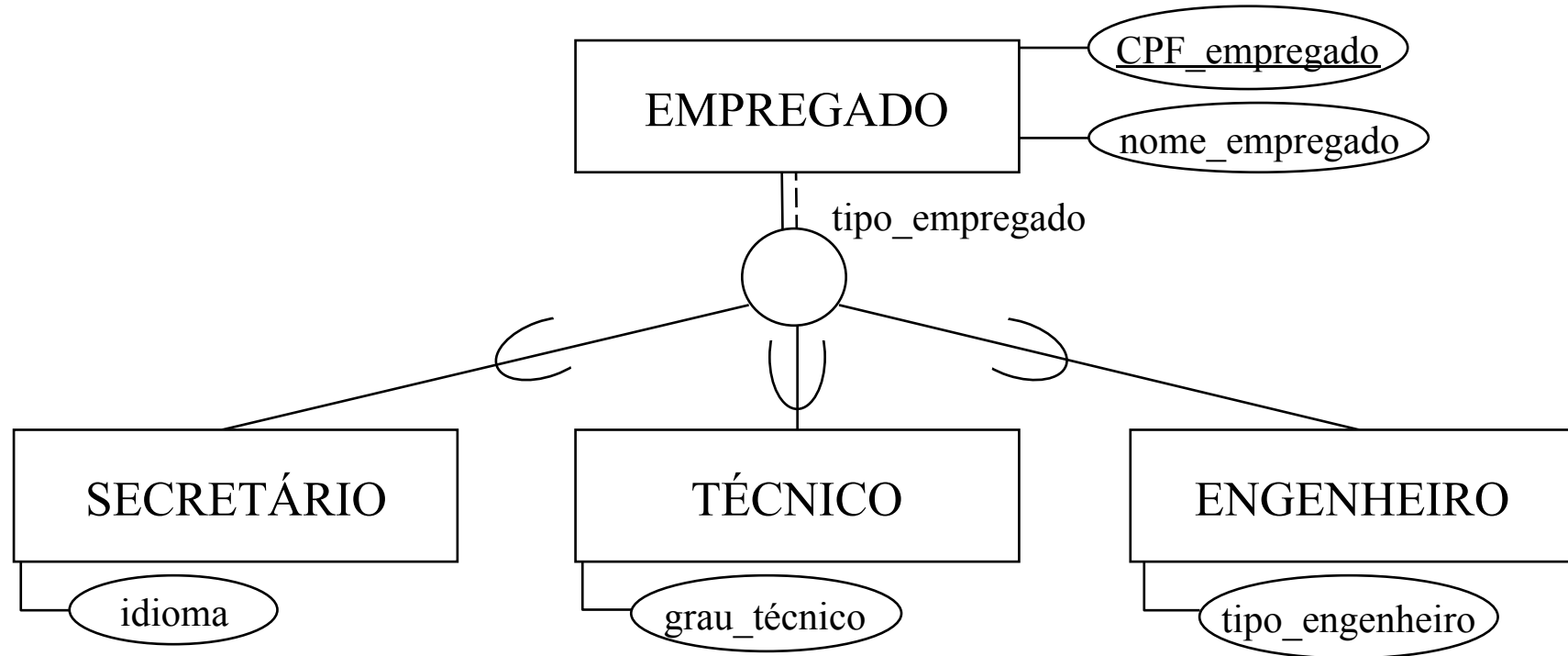


tabelas relativas  
aos tipos-entidade

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

ABC (chave-A, chave-B, chave-C) + integridade referencial

# Generalização/Especialização



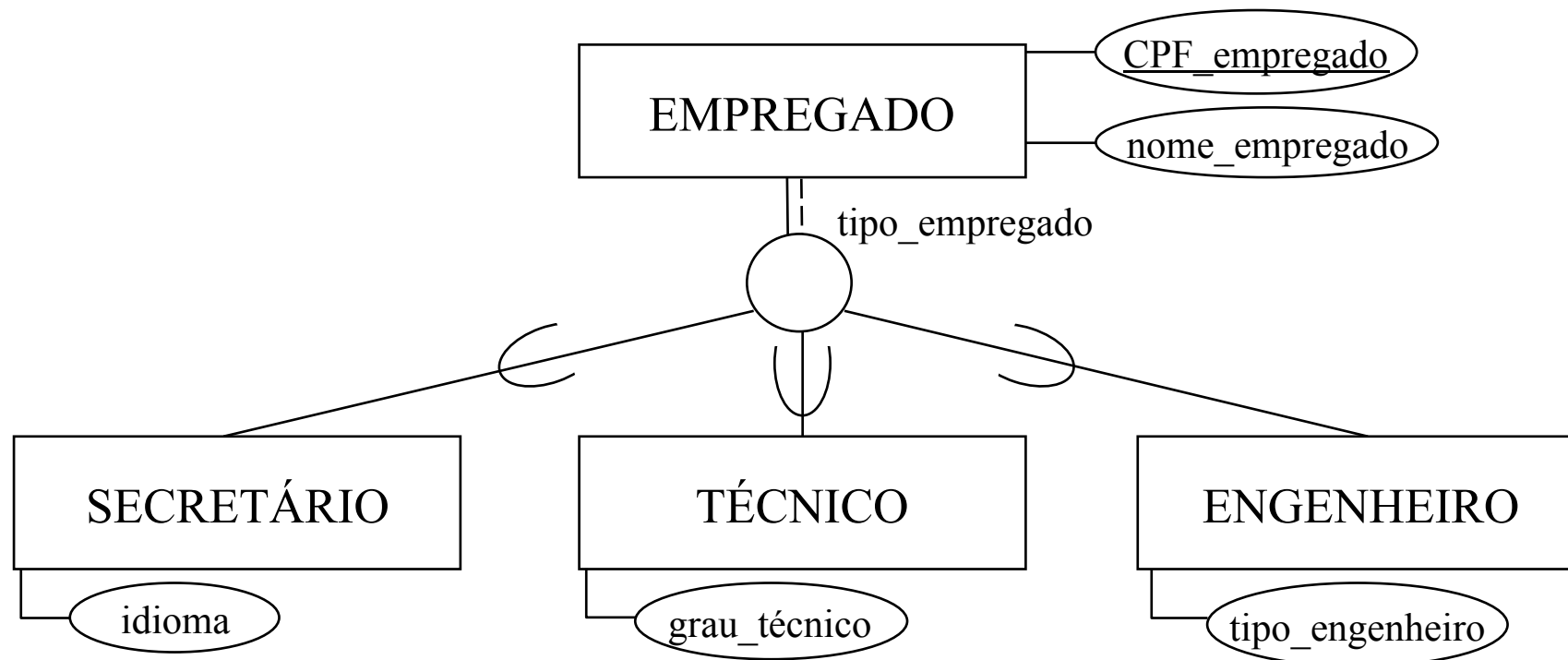
empregado (CPF\_empregado, nome\_empregado, tipo\_empregado)

secretário (CPF\_empregado, idioma)

técnico (CPF\_empregado, grau\_técnico)

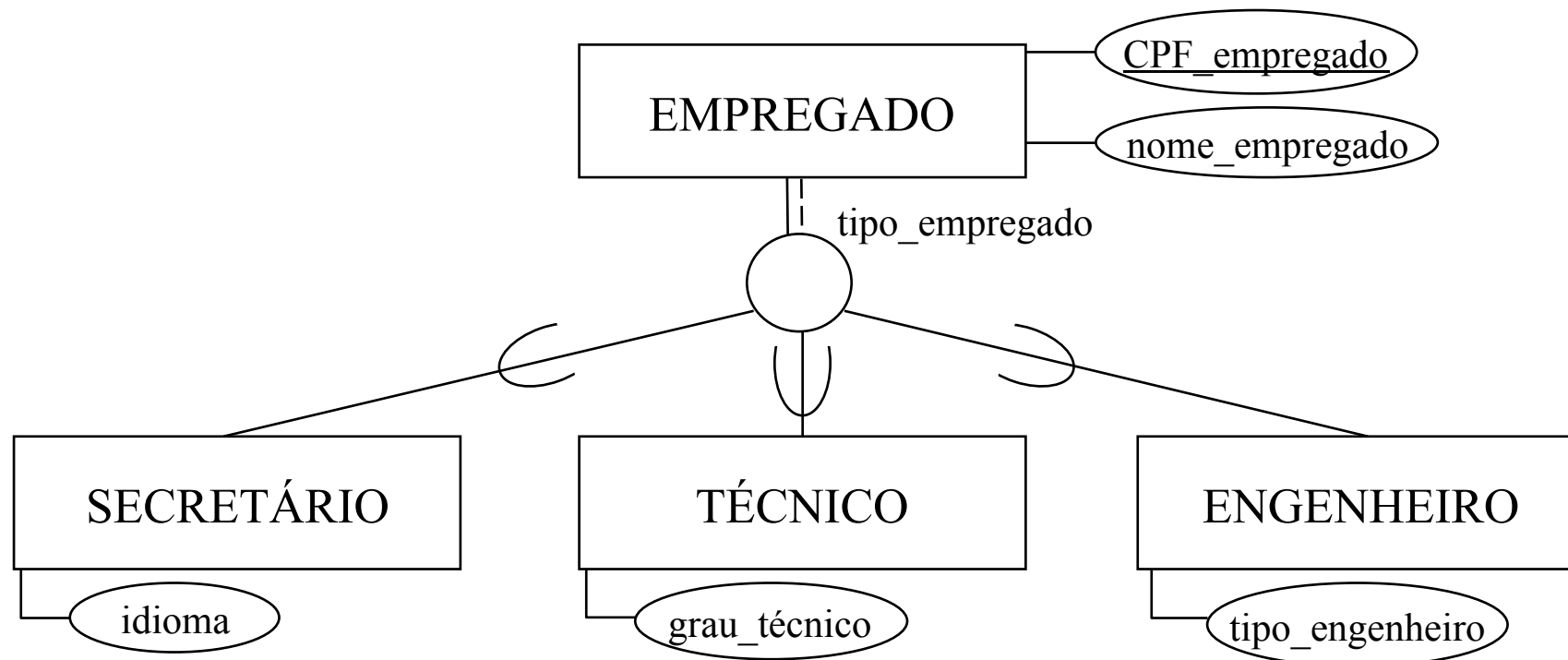
engenheiro (CPF\_empregado, tipo\_engenheiro)

# Outras Formas de Mapeamento



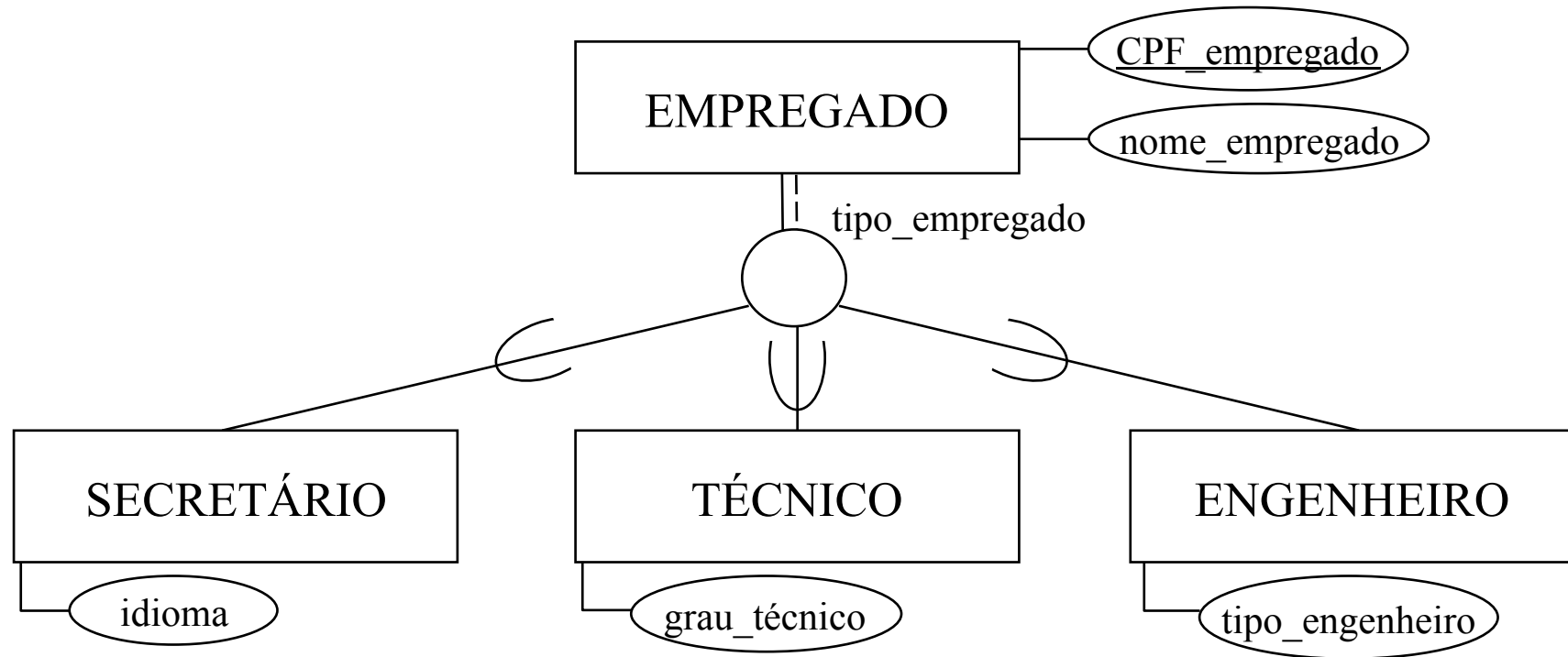
secretário (CPF empregado, nome\_empregado, idioma)  
técnico (CPF empregado, nome\_empregado, grau\_técnico)  
engenheiro (CPF empregado, nome\_empregado, tipo\_engenheiro)

# Outras Formas de Mapeamento



empregado (CPF\_empregado, nome\_empregado, tipo\_empregado, idioma, grau\_técnico, tipo\_engenheiro)

# Outras Formas de Mapeamento



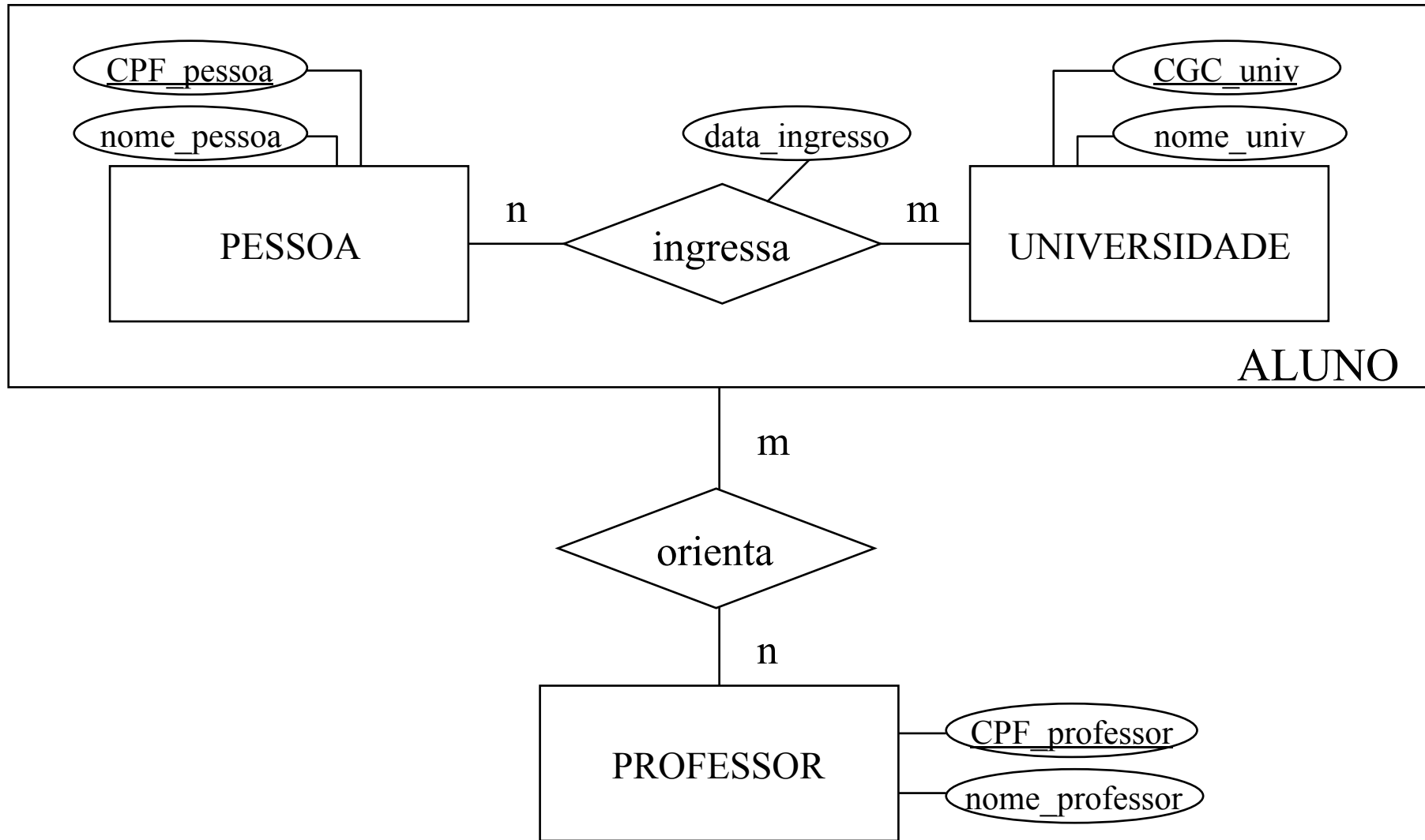
empregado (CPF\_empregado, nome\_empregado)

↑  
SeTeEn (CPF\_empregado, idioma, grau\_técnico, tipo\_engenheiro,  
tipo\_empregado)

# Atributo Tipo-Empregado

- ◆ Atributo único
  - tipo-empregado
  - assume valores diferentes, de acordo com o tipo do empregado
- ◆ Diversos atributos
  - tipo\_empS, tipo\_empT, tipo\_empE, ...
  - cada um dos atributos assume valor 0 ou 1, de acordo com o tipo do empregado
  - abordagem muito mais flexível, principalmente para hierarquias com restrição de sobreposição

# Agregação



# Mapeamento

pessoa (CPF\_pessoa, nome\_pessoa)

universidade (CGC\_univ, nome\_univ)

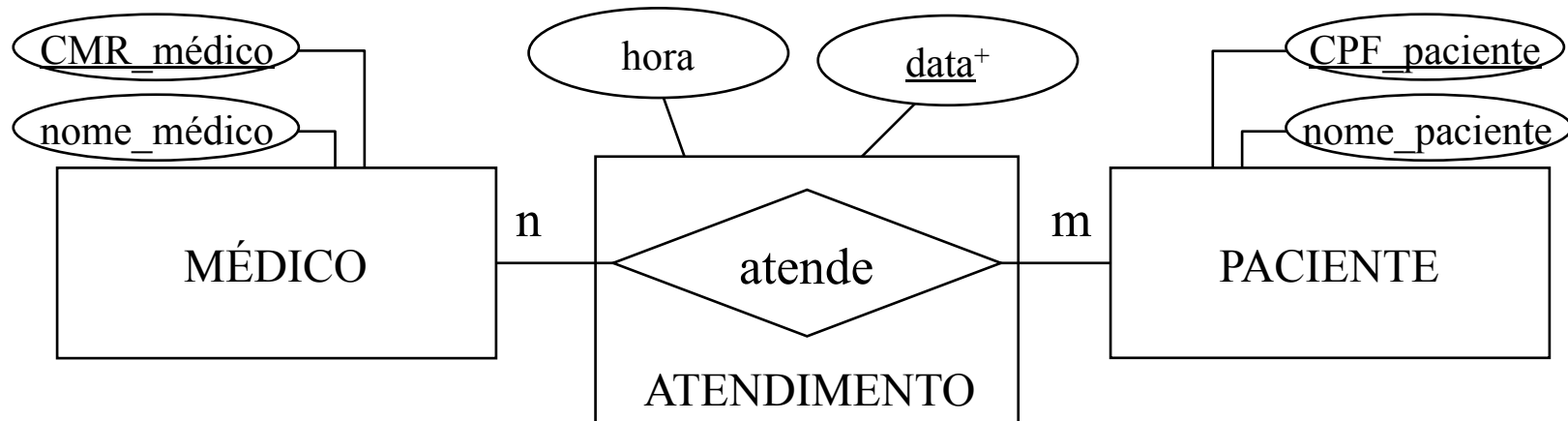
ingressa/aluno (CPF\_pessoa, CGC\_univ, data\_ingresso)

professor (CPF\_professor, nome\_professor)

orienta (CPF\_pessoa, CGC\_univ, CPF\_professor)



# Agregação

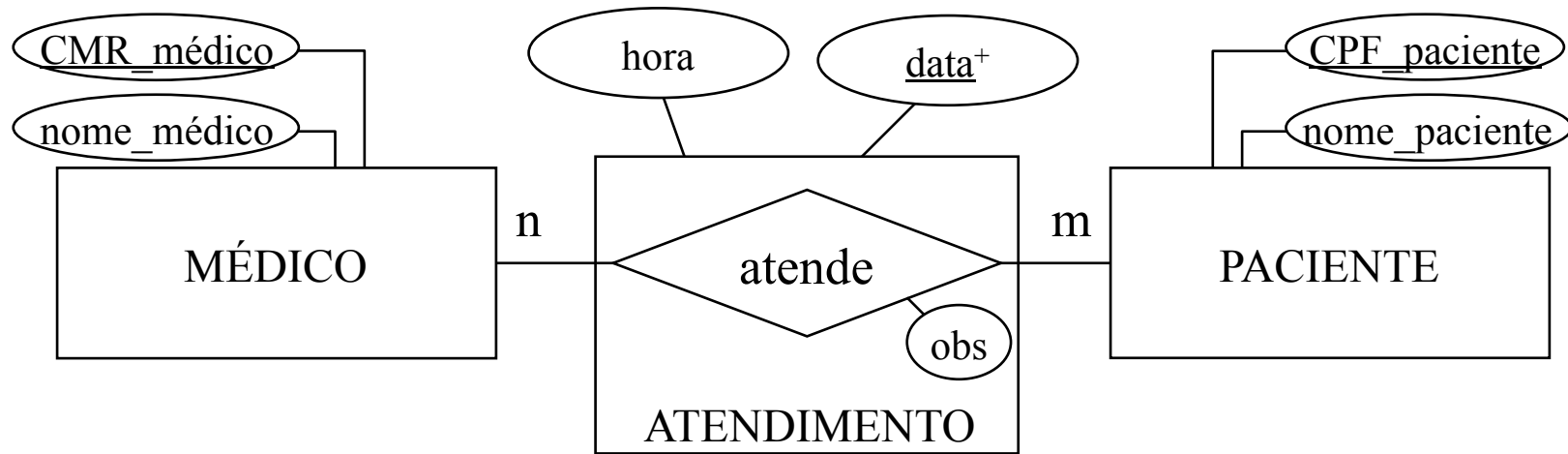


médico (CRM médico, nome\_médico)

paciente (CPF paciente, nome\_paciente)

atendimento (CRM médico, CPF paciente, data, hora)

# Agregação



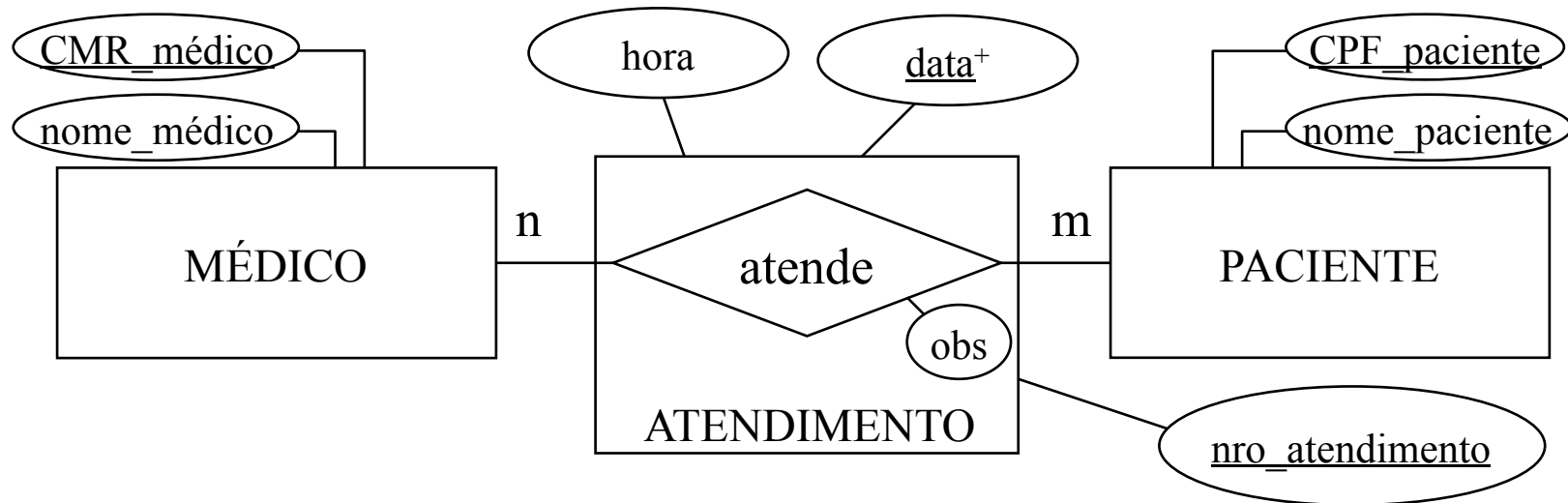
médico (CRM médico, nome\_médico)

paciente (CPF paciente, nome\_paciente)

atendimento (CRM médico, CPF paciente, data, hora)

atende (CRM médico, CPF paciente, obs)

# Agregação



médico (CRM médico, nome\_médico)

paciente (CPF paciente, nome\_paciente)

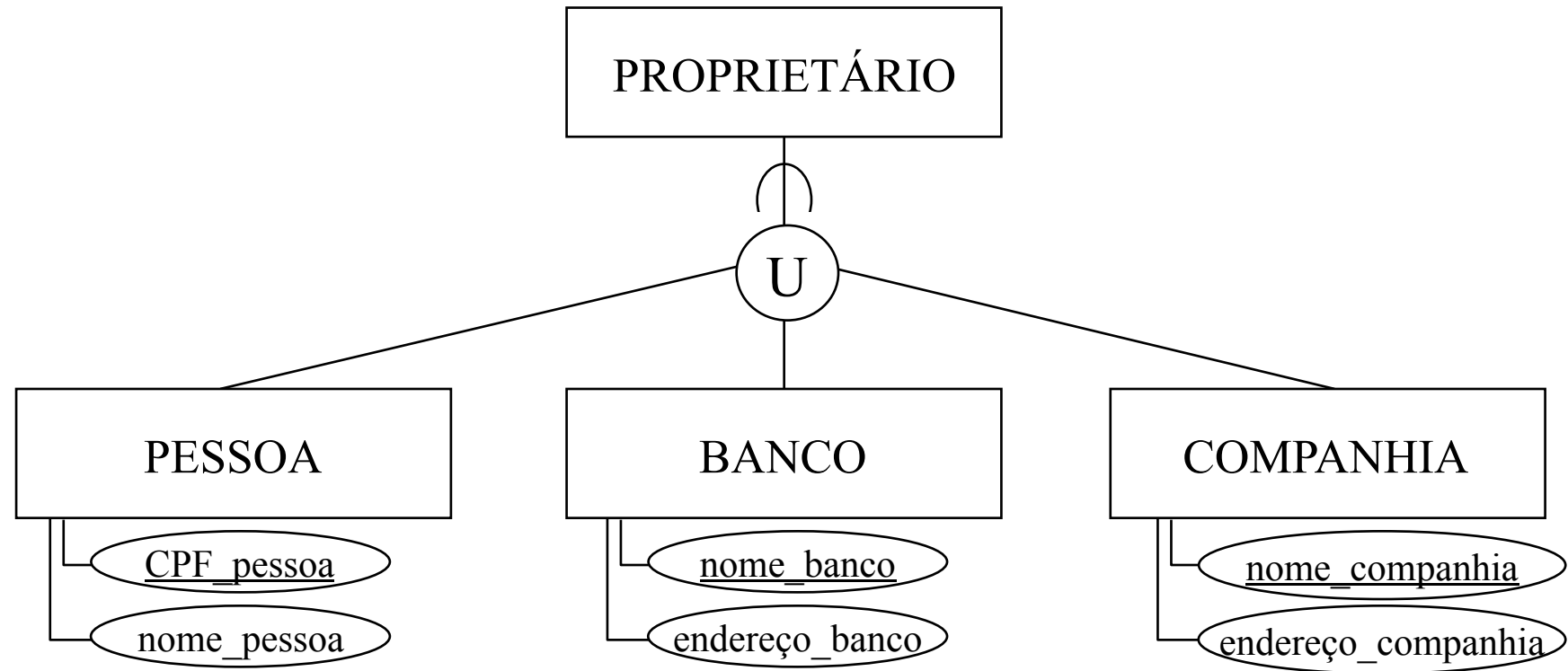
atendimento (CRM médico, CPF paciente, data, hora,  
nro atendimento)

atende (CRM médico, CPF paciente, obs)

# Categoria

- ◆ Modelo entidade relacionamento
  - C : categoria
  - $E_2, \dots, E_n$  : superclasses de C, com chaves primárias diferentes
- ◆ Modelo relacional
  - a tabela de C possuirá:
    - ◆ uma chave primária própria: **chave substituta**
    - ◆ os seus atributos específicos
  - as tabelas de  $E_2$  a  $E_n$  possuirão:
    - ◆ sua própria chave primária
    - ◆ os seus atributos específicos
    - ◆ uma chave estrangeira referenciando a chave primária de C

# Categoria



proprietário (proprietário\_ID)

pessoa (CPF\_pessoa, nome\_pessoa, proprietário\_ID)

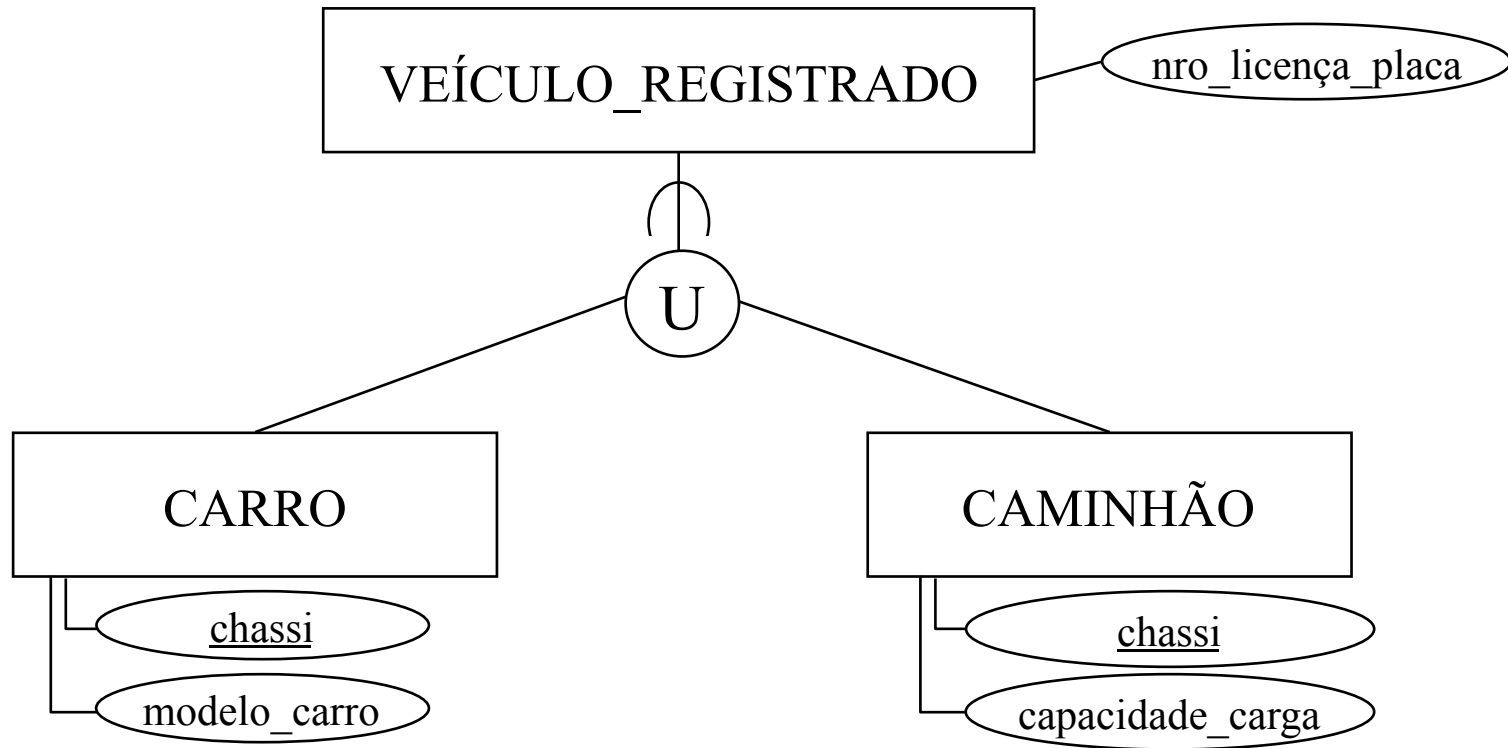
banco (nome\_banco, endereço\_banco, proprietário\_ID)

companhia (nome\_companhia, endereço\_companhia, proprietário\_ID)

# Categoria

- ◆ Modelo entidade relacionamento
  - C : categoria
  - $E_2, \dots, E_n$  : superclasses de C, com chaves primárias idênticas
- ◆ Modelo relacional
  - a tabela de C possuirá:
    - ◆ a chave primária de qualquer superclasse
    - ◆ os seus atributos específicos
  - as tabelas de  $E_2$  a  $E_n$  possuirão:
    - ◆ a sua chave primária
    - ◆ os seus atributos específicos

# Categoria



veículo\_registrado (chassi, nro\_licença\_placa)  
carro (chassi, modelo\_carro)  
caminhão (chassi, capacidade\_carga)