

Conceitos Básicos de Banco de Dados

Laboratório de Bases de Dados

Profa. Dra. Cristina Dutra de Aguiar Ciferri

Sistema de Banco de Dados (SBD)

- ♦ Sistema de armazenamento de dados

- ♦ Objetivos:

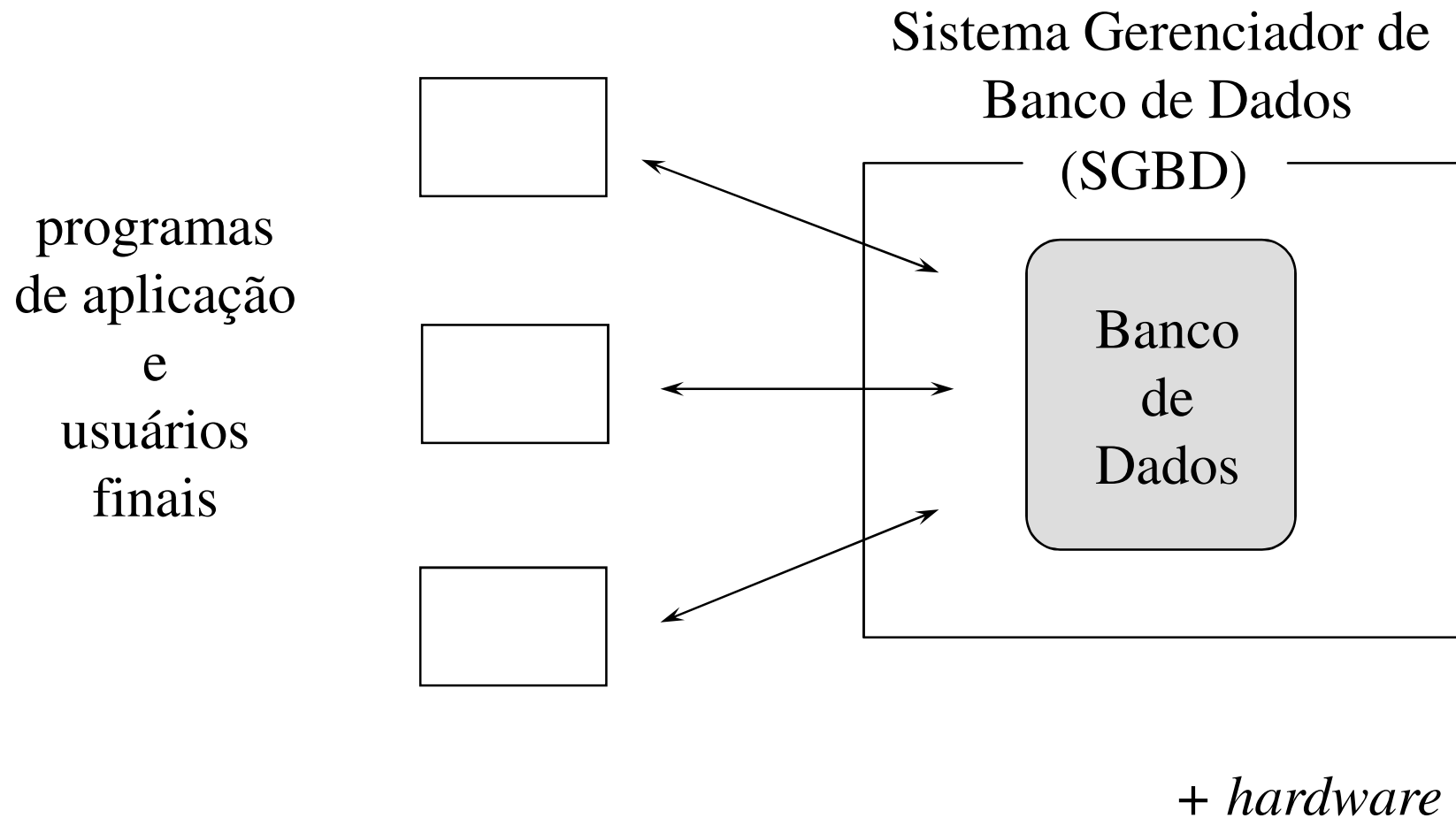
- manter informações
- torná-las disponível quando necessário

- ♦ Armazenamento não volátil

- ♦ Componentes:

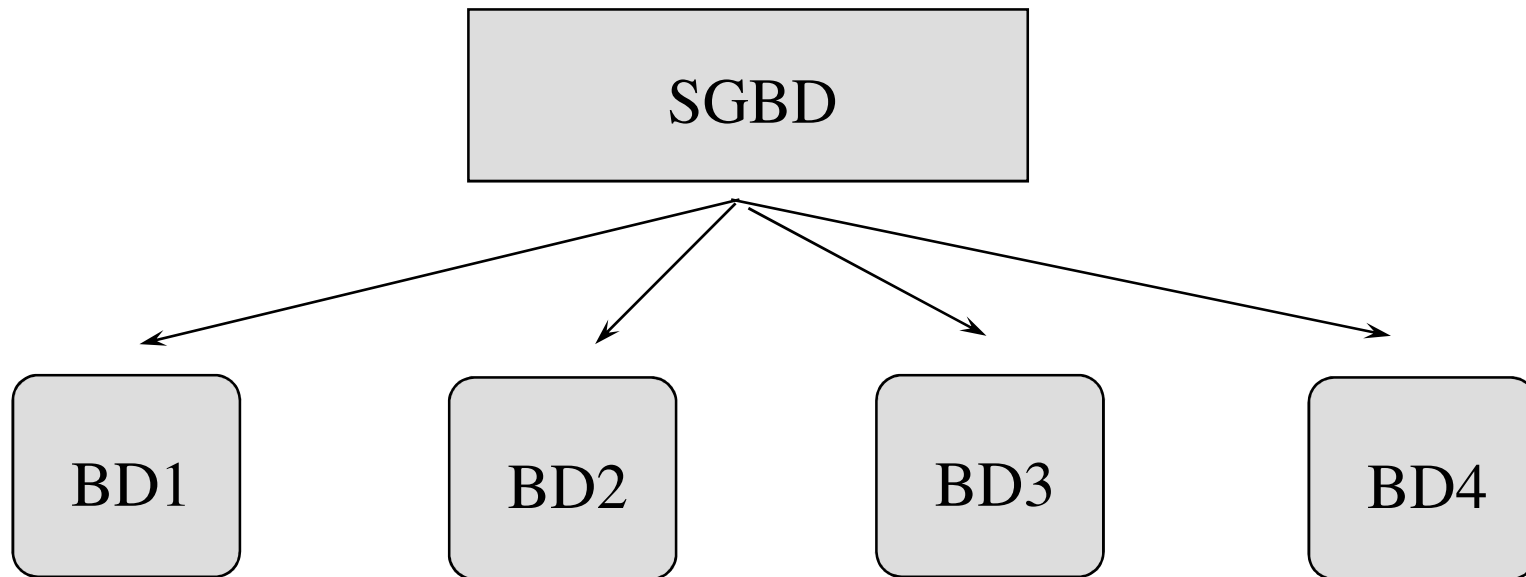
- banco de dados
- sistema gerenciador de banco de dados
- usuários
- hardware

Sistema de Banco de Dados (SBD)



Banco de Dados (BD)

- ✦ Depósito de dados armazenados
- ✦ Os dados devem ser logicamente coerentes
- ✦ Uma coleção randômica não é um BD



Sistema Gerenciador de Banco de Dados (SGBD)

- ♦ Coleção de programas para:
 - criar
 - mantero banco de dados
- ♦ Camada existente entre os dados e os usuários
- ♦ Isola os usuários dos detalhes de *hardware*
- ♦ Atende às solicitações dos usuários

Sistema Gerenciador de Banco de Dados (SGBD)

♦ Recursos:

- adição de novos arquivos
- inserção de dados
- recuperação de dados
- atualização dos dados
- eliminação dos dados
- criação de visões
- atribuição de privilégios
- ...

Usuários

♦ Administrador do BD

- coordena e monitora o uso do BD
- tem conhecimento total do BD

♦ Projetista do BD

- identifica os dados a serem armazenados no BD
- escolhe as estruturas apropriadas para representar e armazenar esses dados

♦ Programador de aplicações

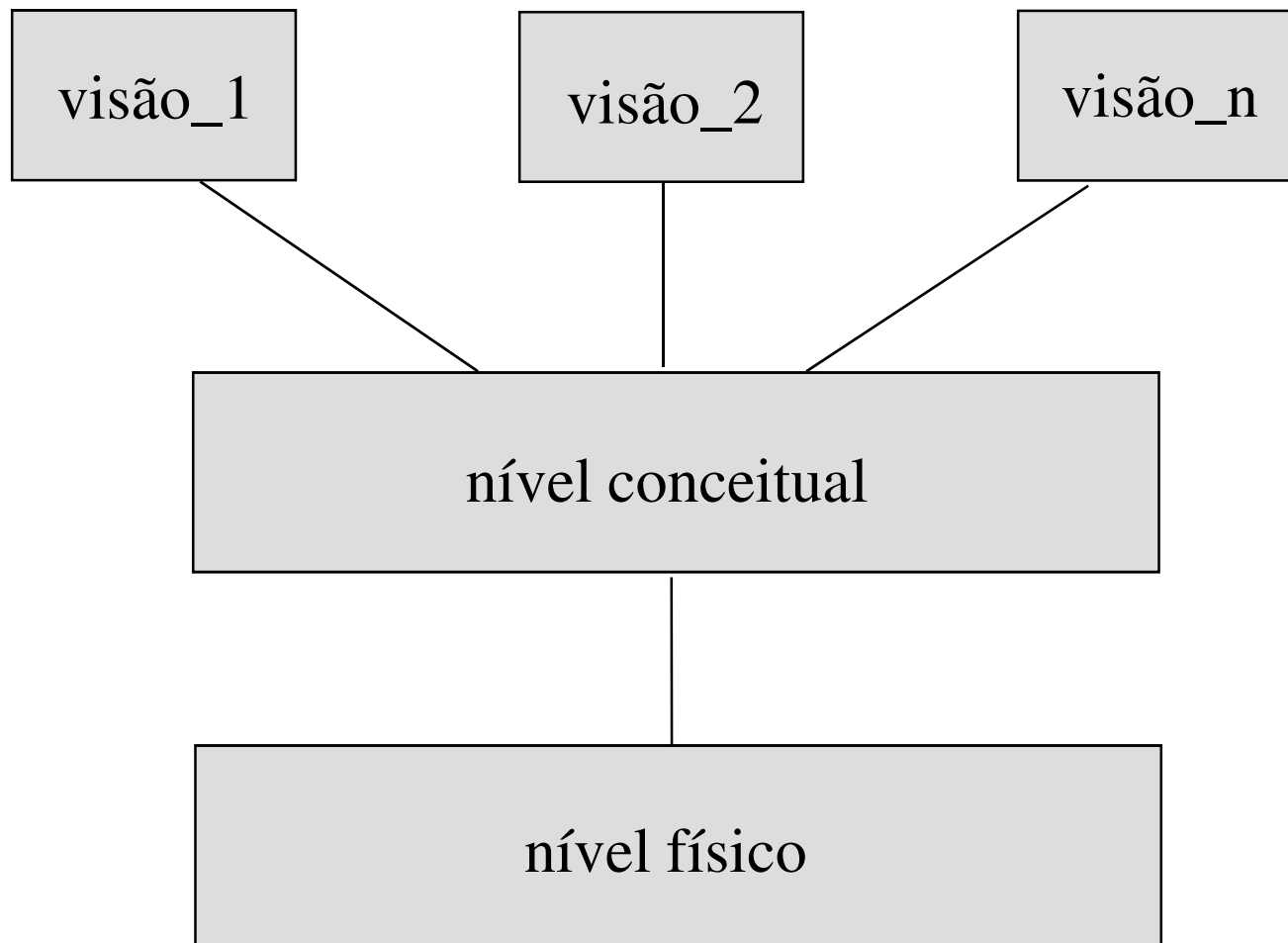
- escreve os programas aplicativos

♦ Usuário final

Hardware

- ♦ Volumes de armazenamento secundário
- ♦ Dispositivos de entrada e saída
- ♦ Canais de entrada e saída
- ♦ Controladores de dispositivos
- ♦ Processador + memórias associadas
 - ULA
 - registradores
 - unidade de controle
- ♦ ...

Arquitetura de Três Níveis



Arquitetura ANSI-X3-SPARC

Arquitetura de Três Níveis

♦ Objetivo

- separar as aplicações dos usuários do BD físico
- prover uma visão abstrata dos dados

♦ Três níveis de abstração

- organização física dos dados
 - ♦ esquema interno
- organização lógica global dos dados
 - ♦ esquema conceitual
- organização lógica particular dos dados
 - ♦ esquema externo (visão)

Arquitetura de Três Níveis

- ♦ Esquema interno
 - dados armazenados na memória secundária
 - contém definições de estruturas de dados e mecanismos de acesso
- ♦ Esquema conceitual
 - definição do conteúdo da informação
 - utiliza o conceito de modelo de dados
 - independe de estruturas de dados e mecanismos de acesso
- ♦ Esquema externo
 - usuário apenas vê parte dos dados
 - visões: também chamadas de subesquemas

Instâncias e Esquemas

♦ Instância

- coleção de informações armazenadas no BD em um determinado momento
- também chamado de **extensão** do BD
- sofre alterações constantemente

♦ Esquema

- projeto do BD, incluindo as entidades e os relacionamentos entre estas
- também chamado de **intenção** do BD
- não sofre alterações com frequência

Estado do Banco de Dados

- ♦ Os dados armazenados em um BD em um determinado momento

- ♦ Estado vazio
 - após a criação do BD
- ♦ Estado inicial
 - após o povoamento (ou carregamento) do BD com os dados iniciais

- ♦ Novo estado
 - após cada operação realizada nos dados do BD
- ♦ Estado atual
 - estado do BD em um determinado momento

Projeto de Banco de Dados

Laboratório de Bases de Dados

Profa. Dra. Cristina Dutra de Aguiar Ciferri

Projeto de Banco de Dados

♦ Objetivo da abordagem de BD

- oferecer abstração dos dados
- separar aplicações dos usuários dos detalhes de *hardware*
- ferramenta utilizada \Rightarrow **modelo de dados**

♦ Modelo de dados

- conjunto de ferramentas conceituais para a descrição dos dados, dos relacionamentos existentes entre os dados, da semântica e das restrições que atuam sobre estes

Modelos de Dados: Categorias

- ♦ Divisão baseada nos tipos de conceitos oferecidos pelo modelo para descrever a estrutura do banco de dados
- ♦ Modelo de dados conceitual
 - modelo de alto nível
 - oferece conceitos próximos aos usuários
 - **Modelo Entidade-Relacionamento (ER)**

Modelos de Dados: Categorias

- ♦ Modelo de dados de implementação
 - oferece conceitos que
 - ♦ podem ser facilmente utilizados por usuários finais
 - ♦ não estão distantes da forma na qual os dados estão organizados no computador
 - é implementado de maneira direta
- Modelo Relacional

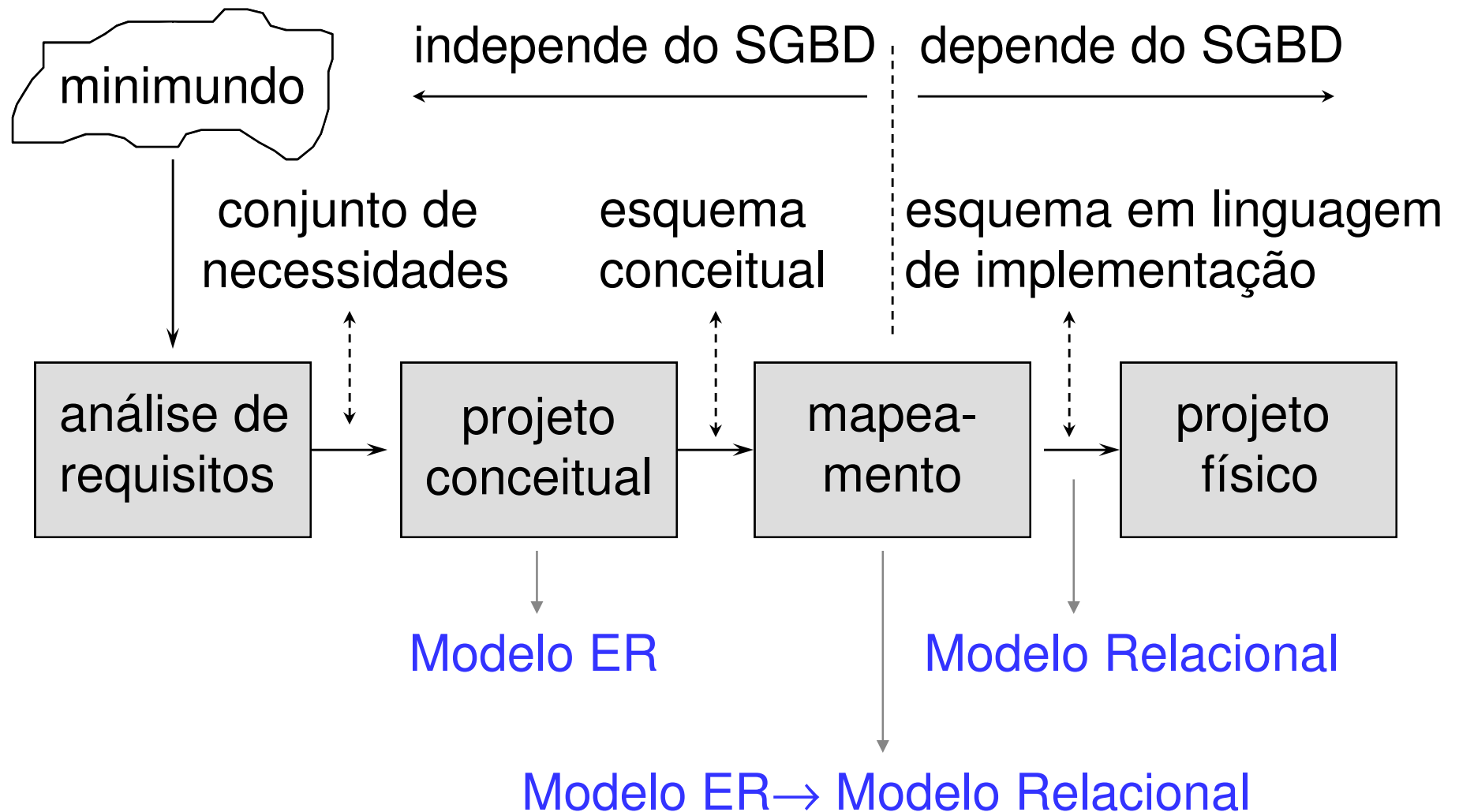
Modelos de Dados: Categorias

♦ Modelo de dados físico

- modelo de dados físico
- descreve como os dados estão armazenados fisicamente no computador

➤ SGBDR (Sistema Gerenciador de Banco de Dados Relacional) Oracle

Projeto de Banco de Dados



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

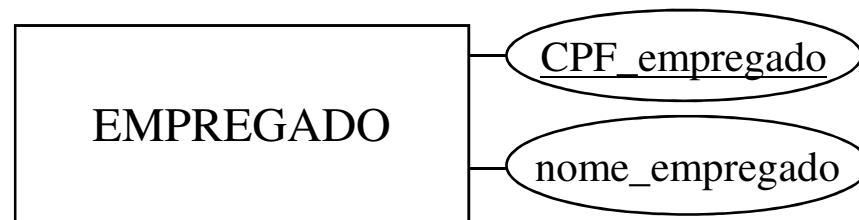
Aspectos Adicionais

- ♦ Opções para remoção/atualização
 - CASCADE
 - SET NULL
 - SET DEFAULT

Mapeamento do Modelo Entidade-Relacionamento para o Modelo Relacional

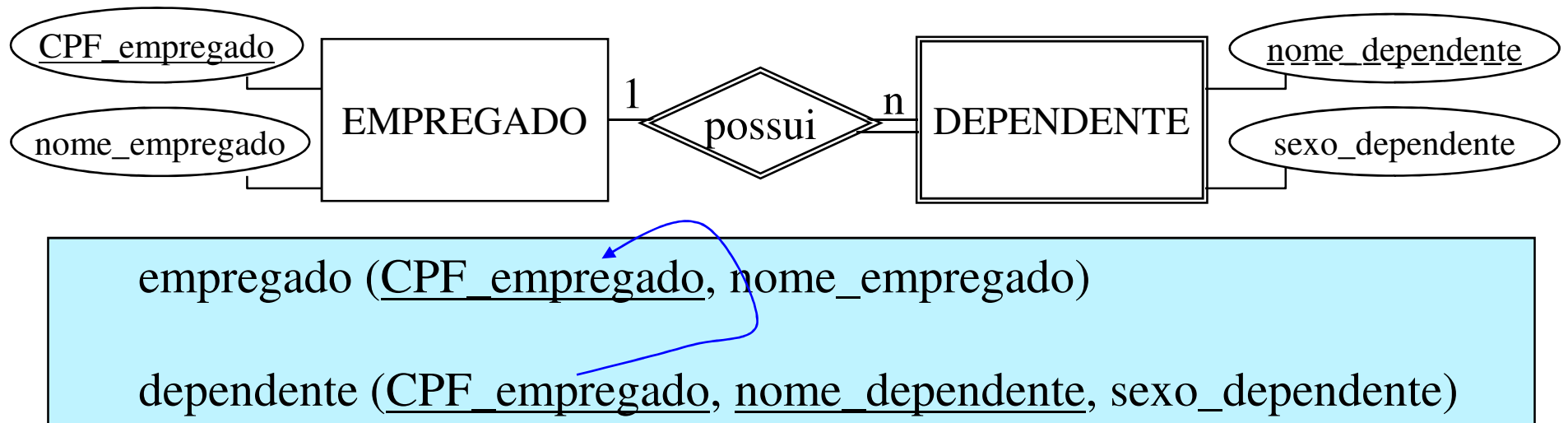
Laboratório de Bases de Dados
Profa. Dra. Cristina Dutra de Aguiar Ciferri

Tipo-Entidade Forte

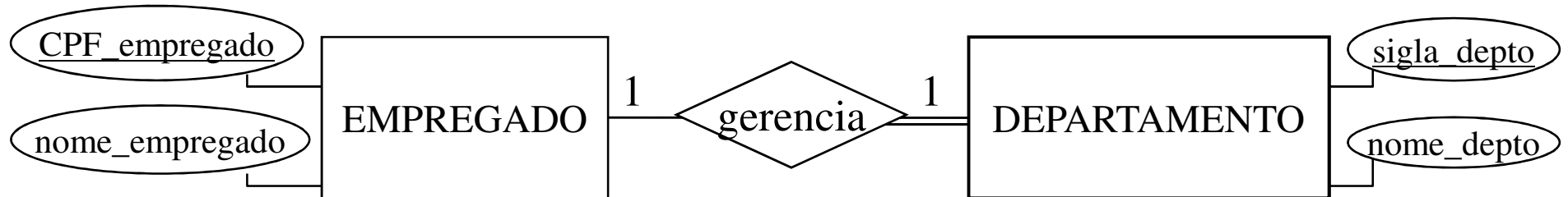


empregado (CPF_empregado, nome_empregado)

Tipo-Entidade Fraca



Tipo-Relacionamento (1:1)

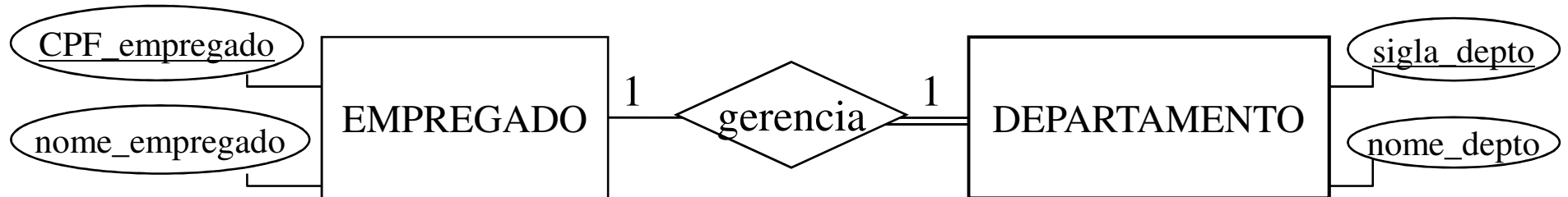


empregado (CPF_empregado, nome_empregado, sigla_depto)
departamento (sigla_depto, nome_depto, CPF_empregado)

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)



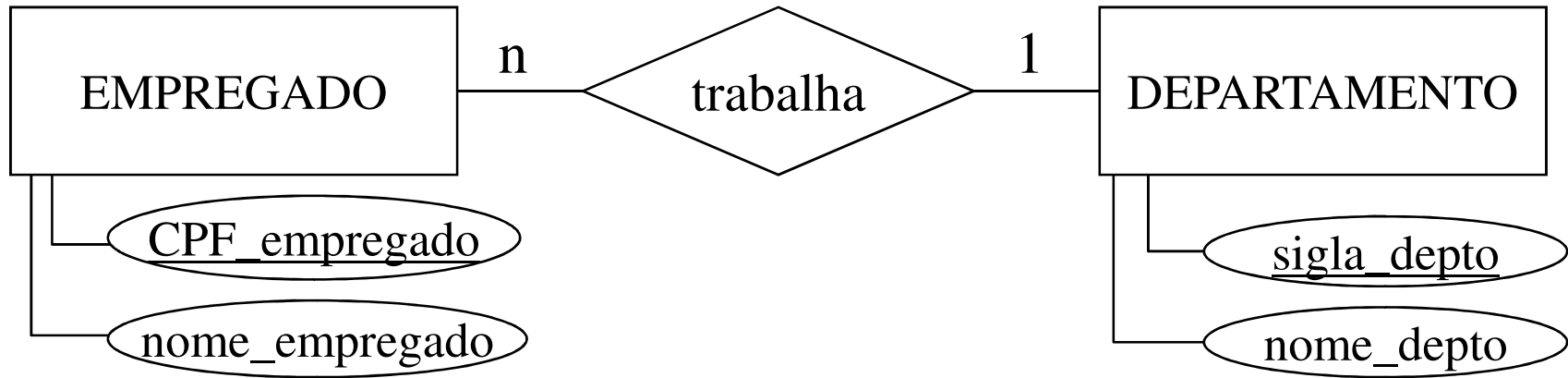
- não pode existir departamento sem gerente
- pode existir empregado que não gerencia o departamento

empregado (CPF_empregado, nome_empregado)

departamento (sigla_depto, nome_depto, CPF_empregado)

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

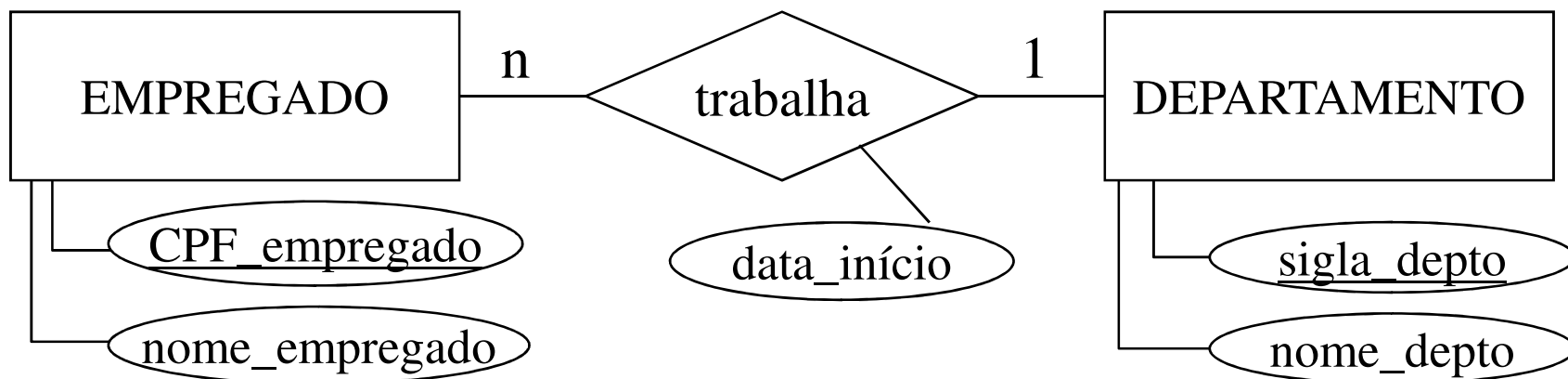
Tipo-Relacionamento (1:n)



empregado (CPF_empregado, nome_empregado, sigla_depto)

departamento (sigla_depto, nome_depto)

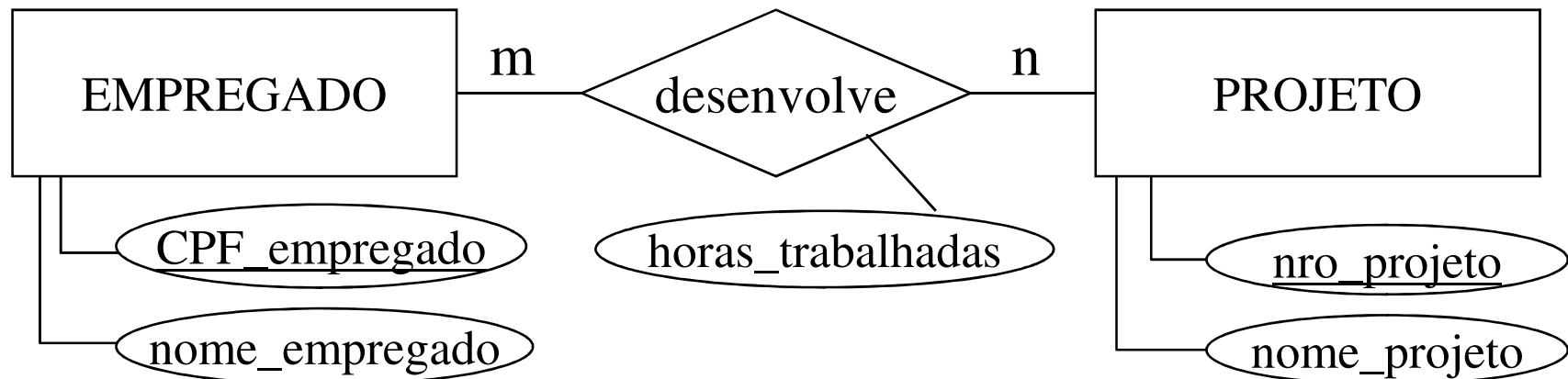
Atributo de Tipo-Relacionamento (1:1 e 1:n)



empregado (CPF_empregado, nome_empregado, sigla_depto, data_início)

departamento (sigla_depto, nome_depto)

Tipo-Relacionamento (m:n)

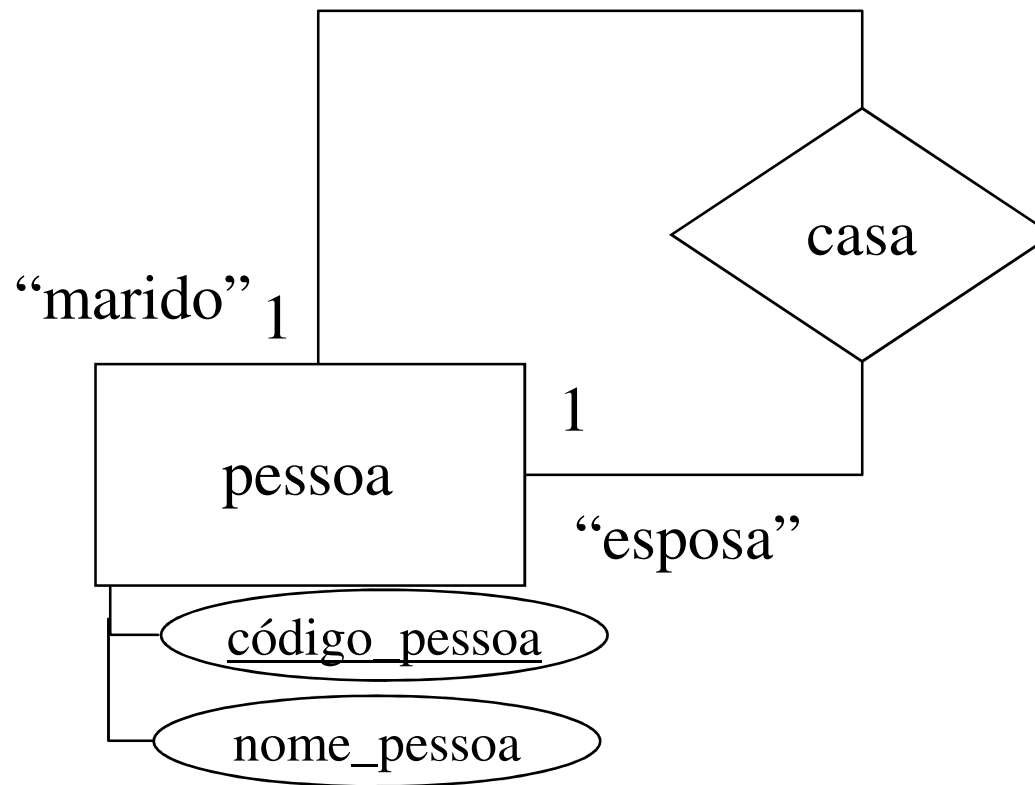


empregado (CPF_empregado, nome_empregado)

projeto (nro_projeto, nome_projeto)

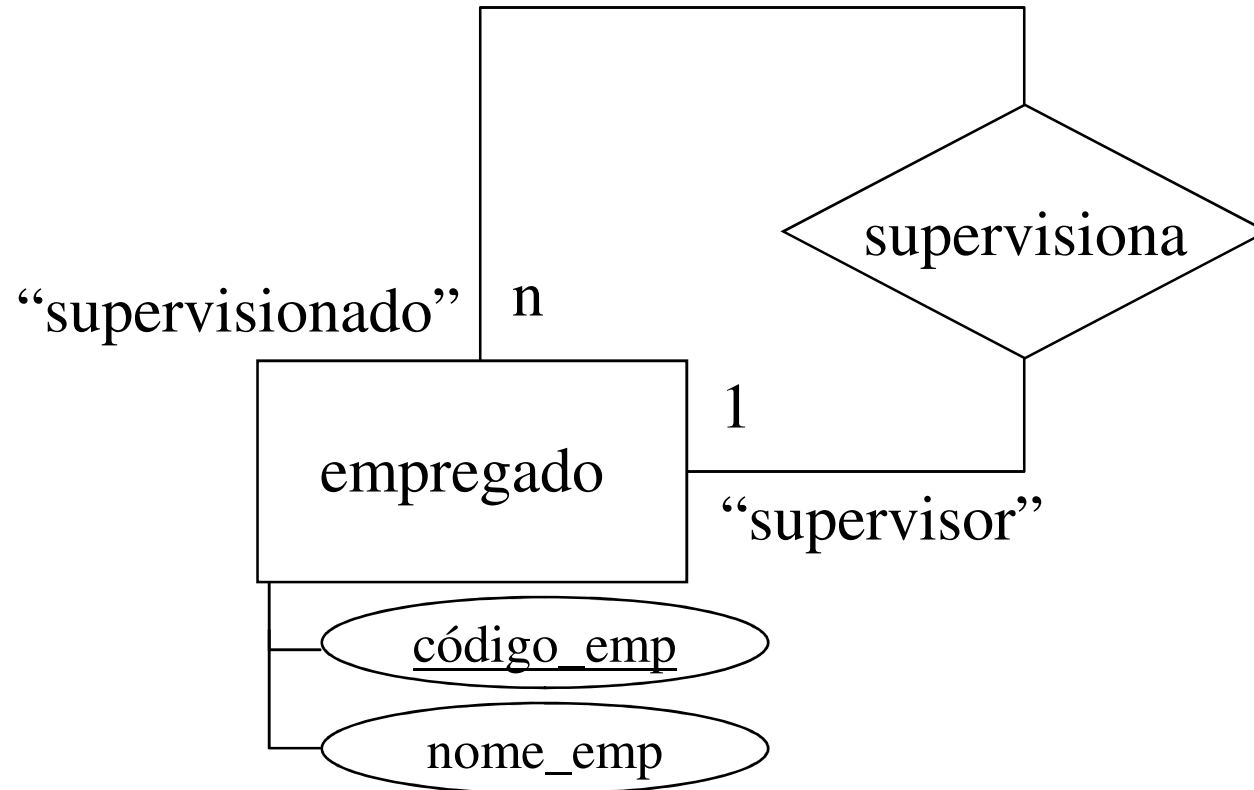
desenvolve (CPF_empregado, nro_projeto, horas_trabalhadas)

Tipo-relacionamento Unário (1:1)



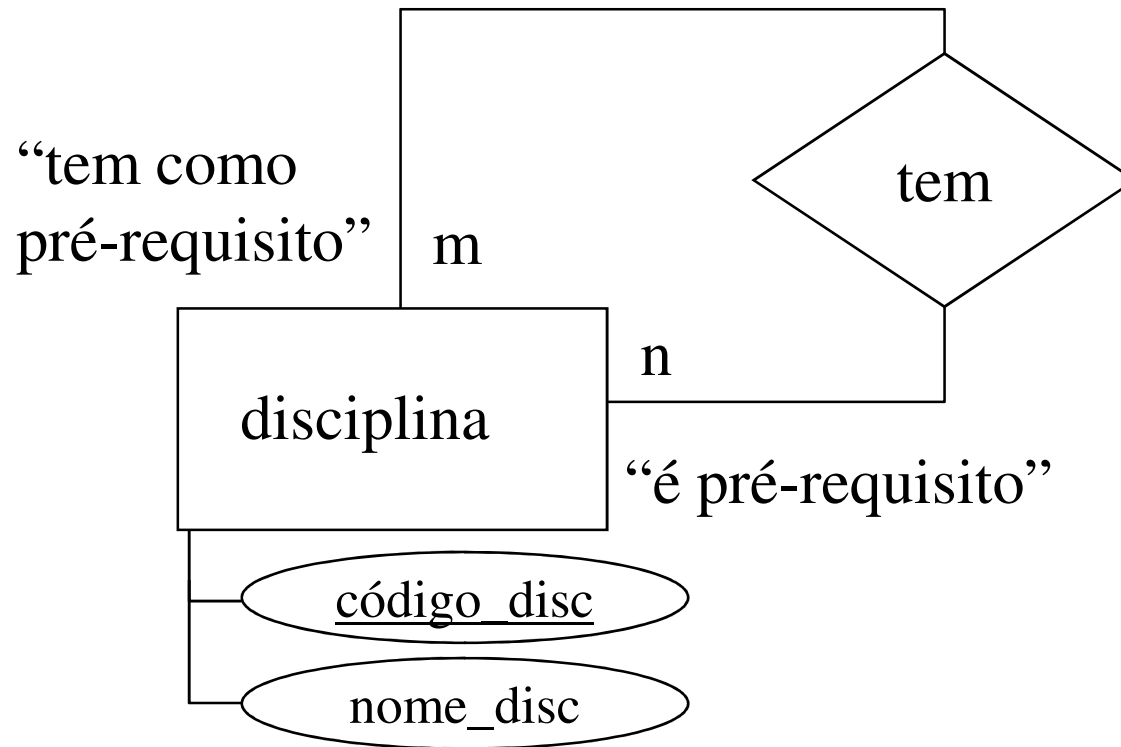
pessoa (código_pessoa, nome_pessoa, código_cônjuge)

Tipo-relacionamento Unário (1:n)



empregado (código_emp, nome_emp, código_supervisor)

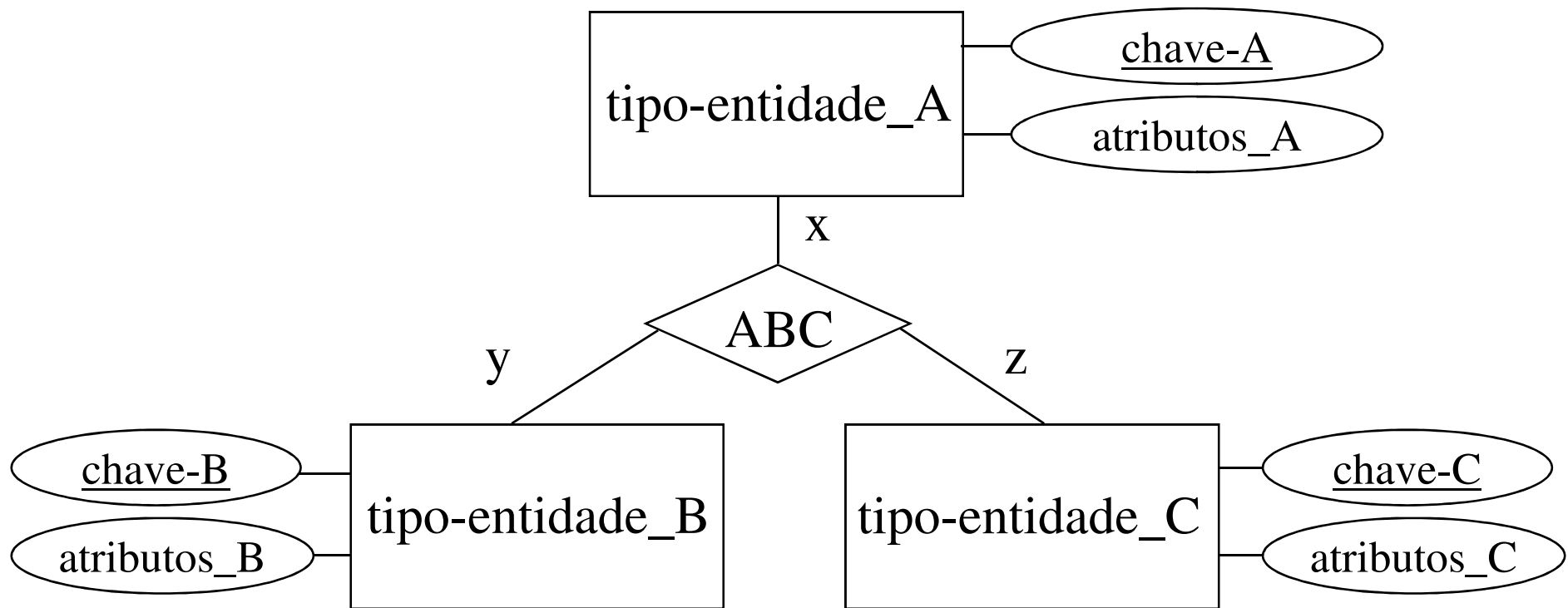
Tipo-relacionamento Unário (m:n)



disciplina (código_disc, nome_disc)

pré_requisito (código_disc, código_pré_requisito)

Tipo-relacionamento Ternário

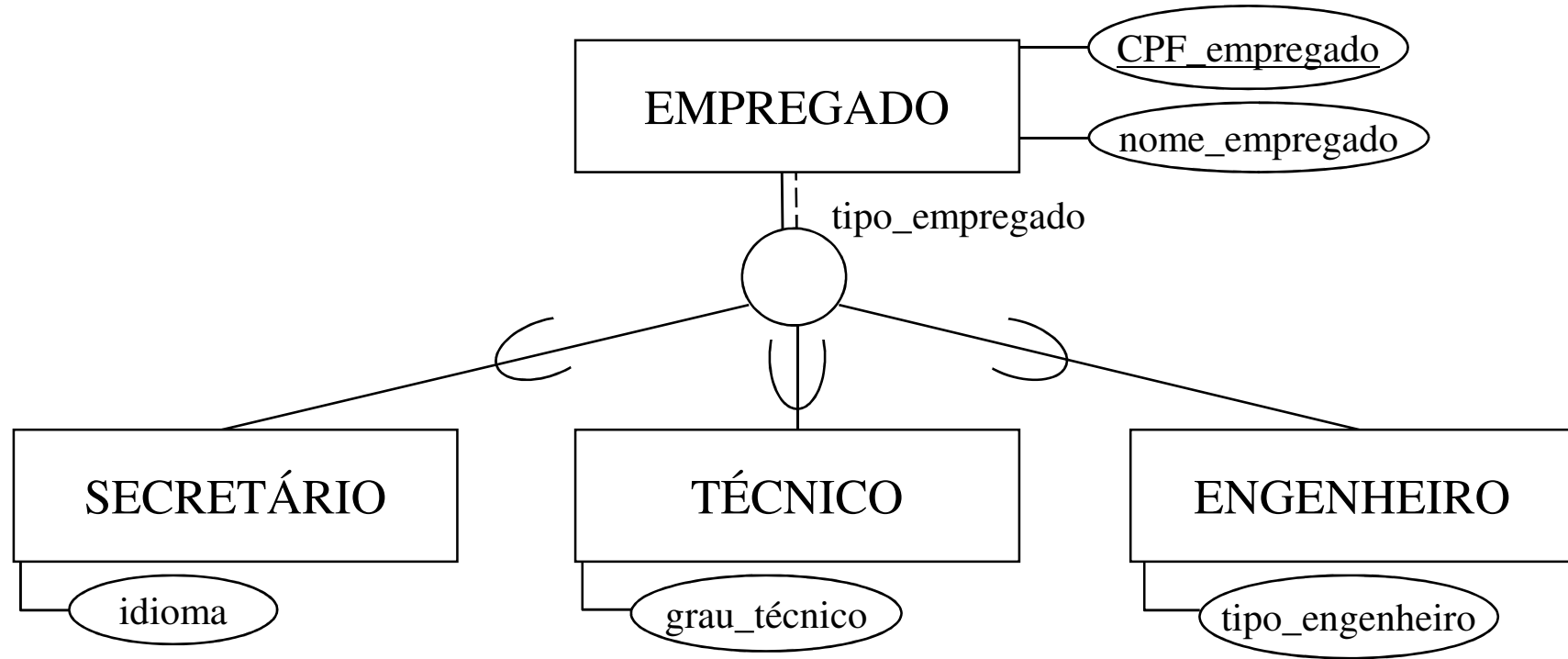


tabelas relativas
aos tipos-entidade

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



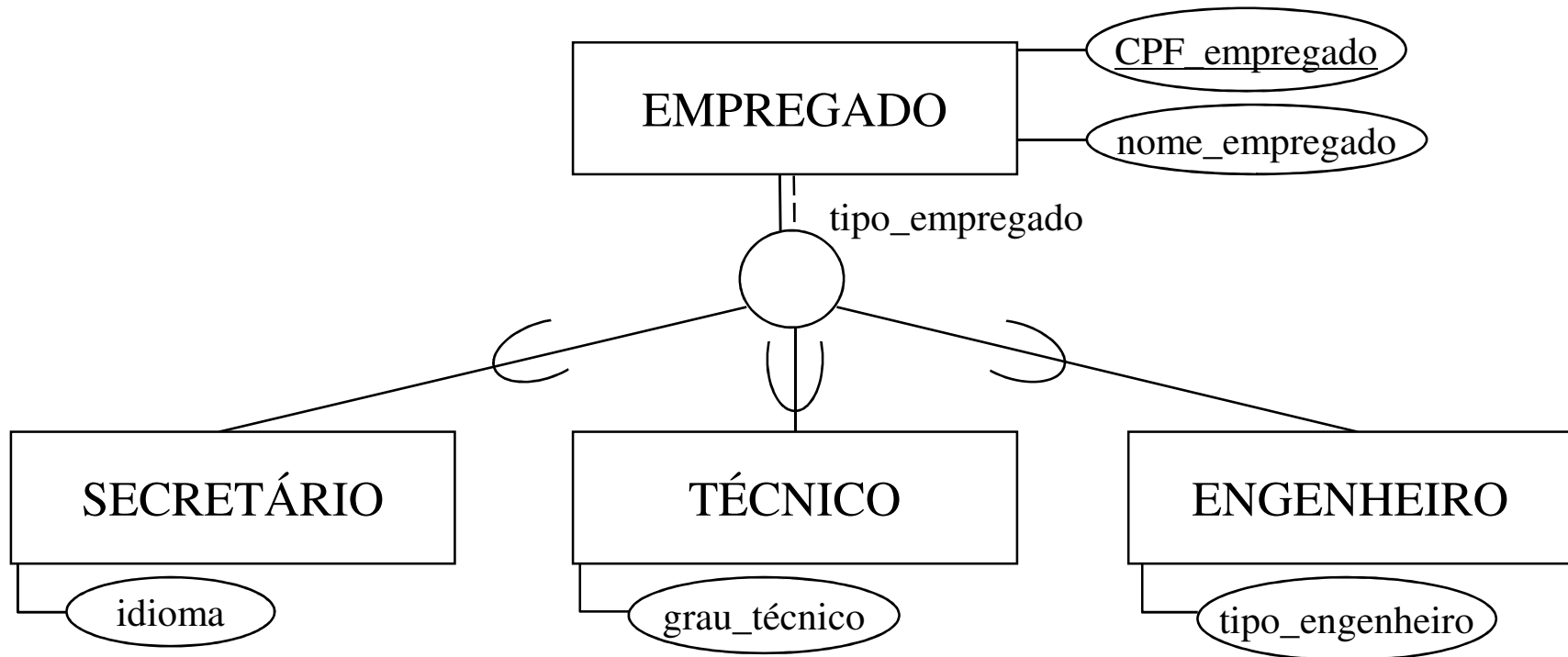
empregado (CPF_empregado, nome_empregado, tipo_empregado)

secretário (CPF_empregado, idioma)

técnico (CPF_empregado, grau_técnico)

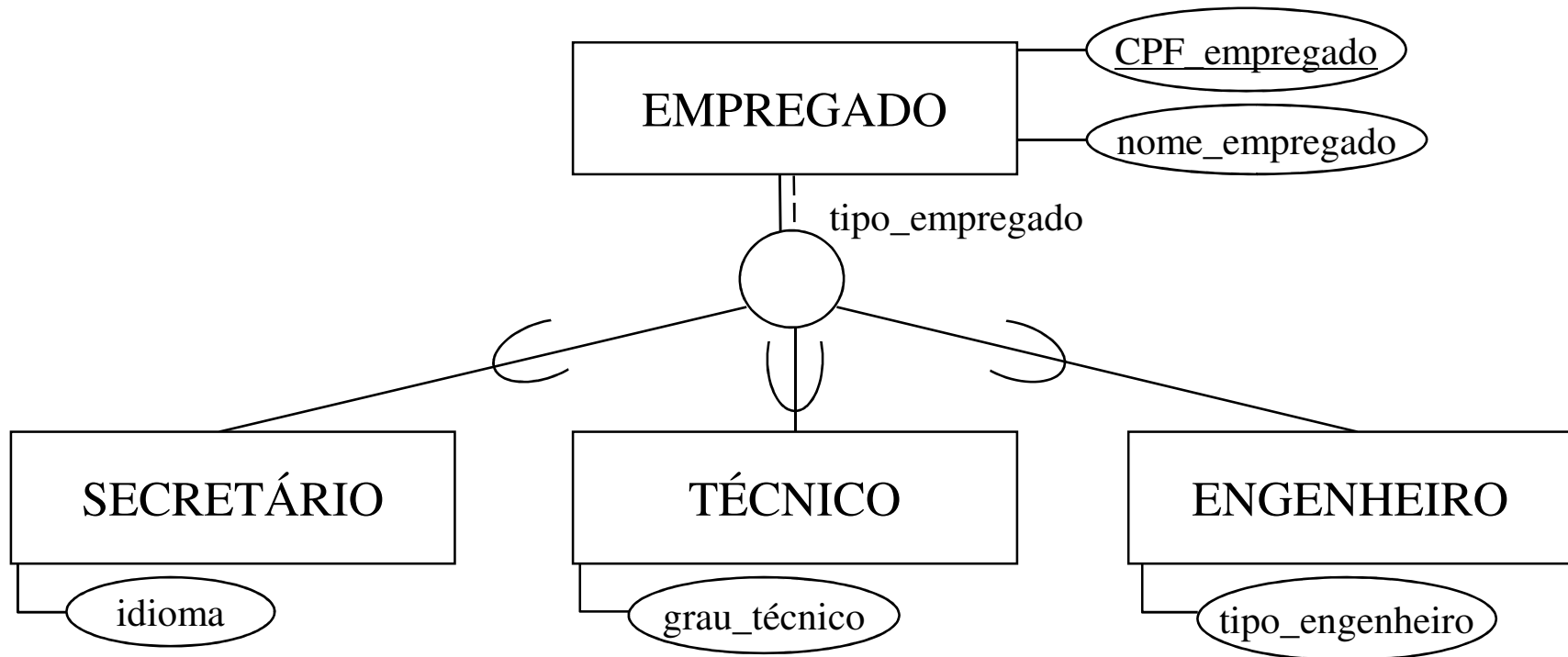
engenheiro (CPF_empregado, tipo_engenheiro)

Outras Formas de Mapeamento



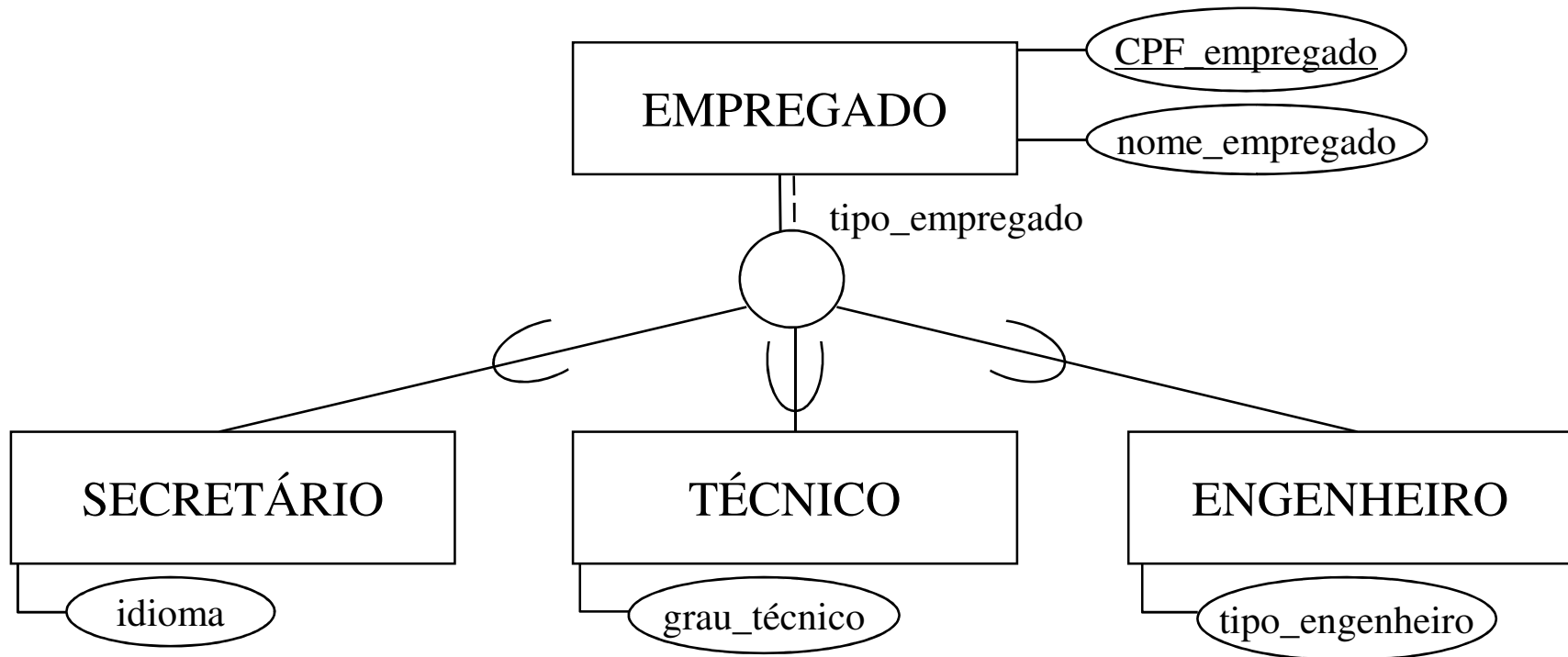
secretário (CPF empregado, nome empregado, idioma)
técnico (CPF empregado, nome empregado, grau técnico)
engenheiro (CPF empregado, nome empregado, tipo engenheiro)

Outras Formas de Mapeamento



empregado (CPF empregado, nome empregado, tipo empregado, idioma, grau técnico, tipo engenheiro)

Outras Formas de Mapeamento



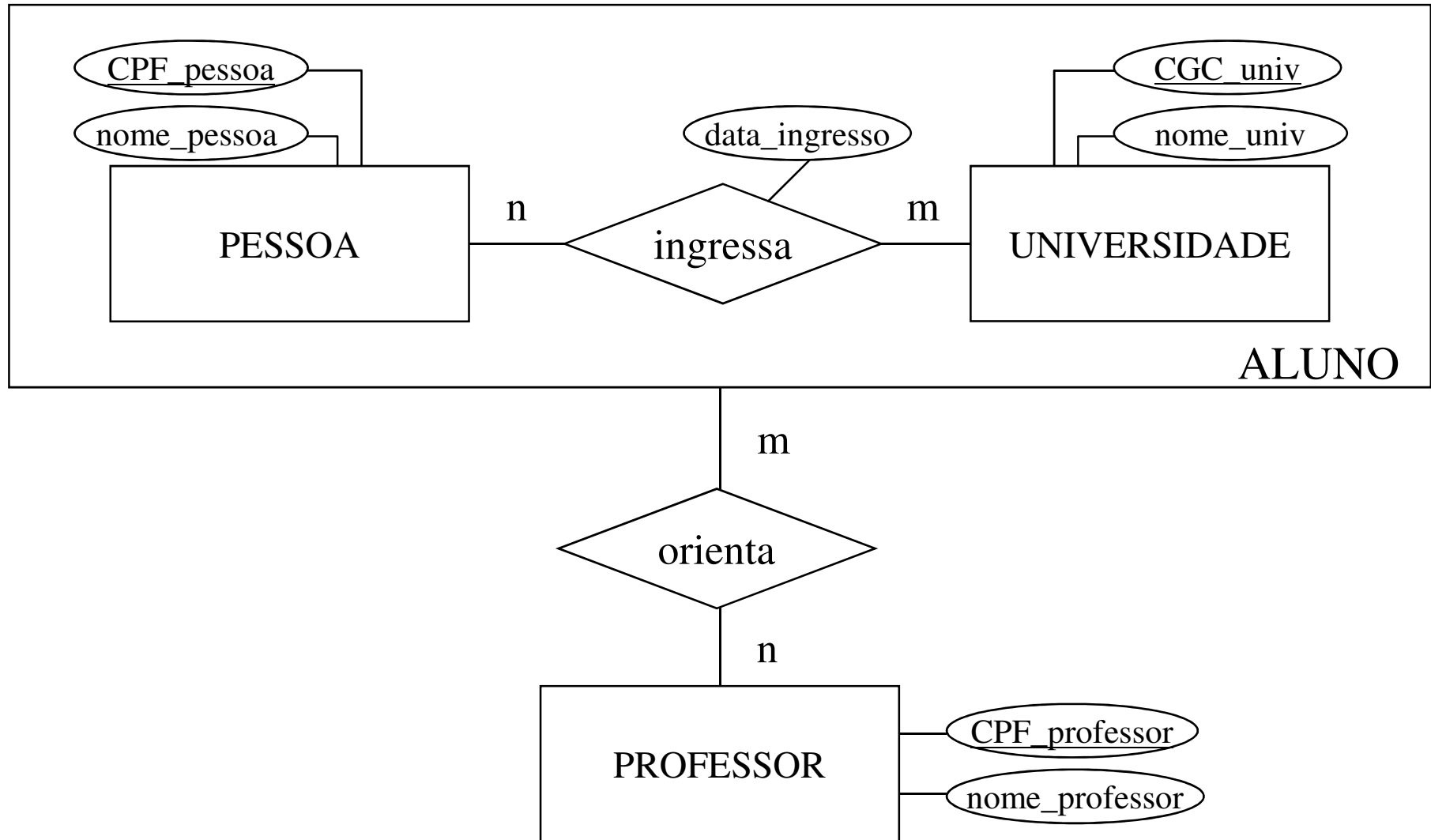
empregado (CPF_empregado, nome_empregado)

↑
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)

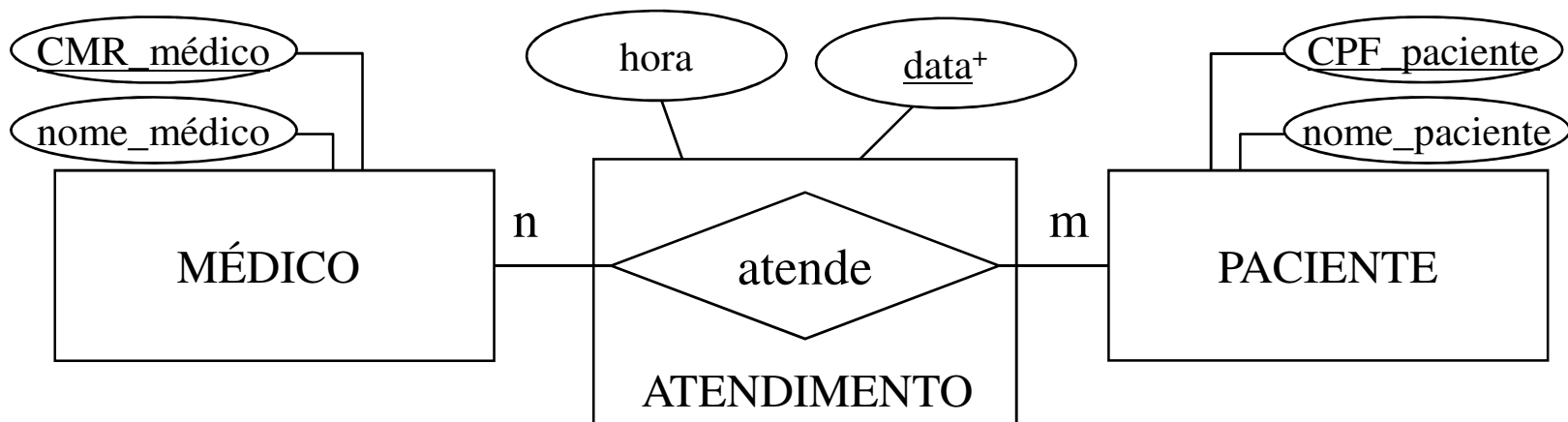
universidade (CGC_univ, nome_univ)

ingressa/aluno (CPF_pessoa, CGC_univ, data_ingresso)

professor (CPF_professor, nome_professor)

orienta (CPF_pessoa, CGC_univ, CPF_professor)

Agregação

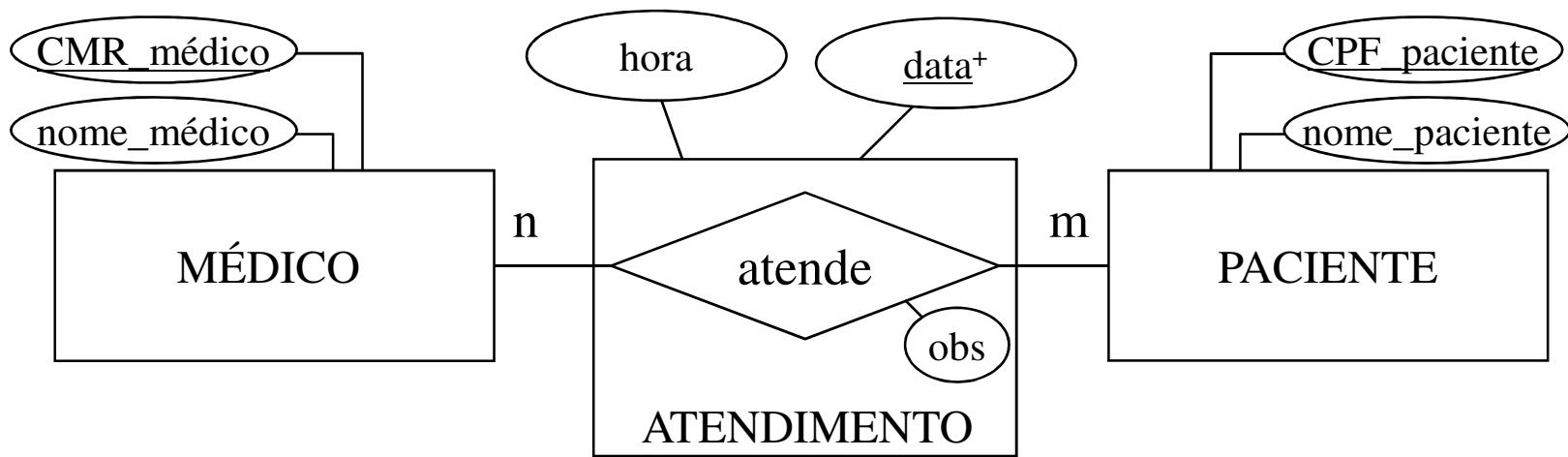


médico (CRM_médico, nome_médico)

paciente (CPF_paciente, nome_paciente)

atendimento (CRM_médico, CPF_paciente, data, hora)

Agregação



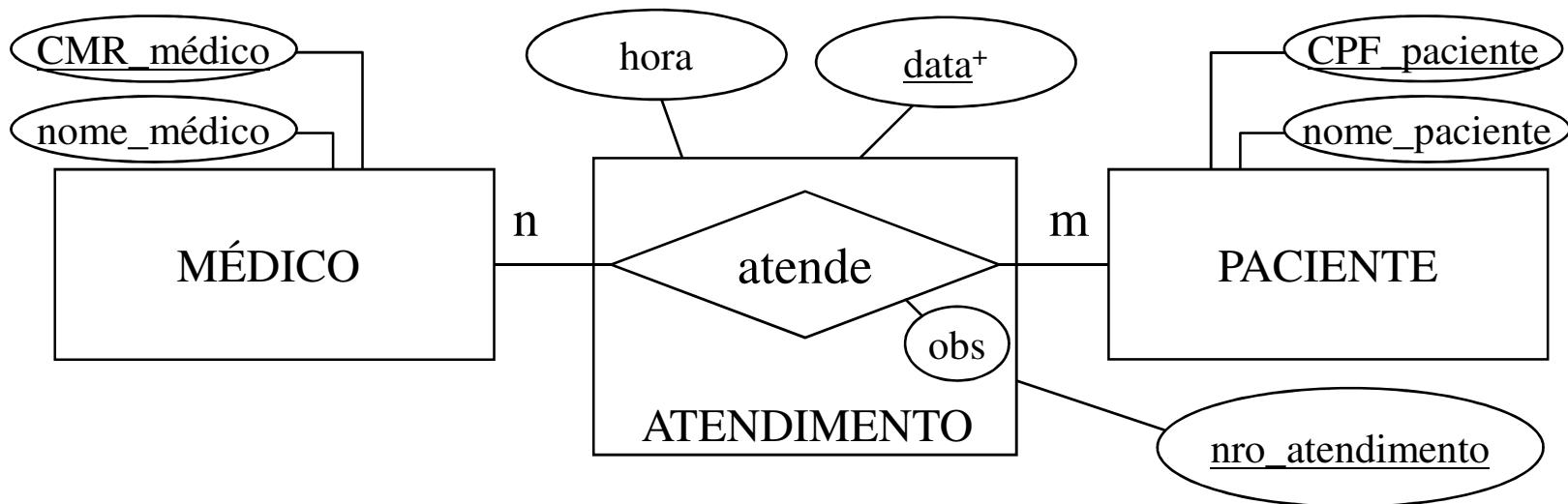
médico (CRM_médico, nome_médico)

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)