



# Modelo Entidade Relacionamento Estendido

Prof. Bruno Travençolo

Slides cedidos pelo Prof. Humberto Razente



# Modelo Entidade Relacionamento Estendido

## ◆ Características

- introduz semântica adicional ao modelo ER
- utilizado na modelagem de aplicações mais complexas

## ◆ Conceitos

- subclasse, superclasse, hierarquia de herança
- generalização, especialização, e restrições
- agregação

# Subclasse/Superclasse

## ◆ Subclasse

- agrupamento das entidades de um subgrupo do tipo-entidade

## ◆ Exemplo

- superclasse: tipo-entidade empregado
- subclasses: secretário, engenheiro, técnico

cada entidade que é membro de qualquer uma das subclasses também é um empregado

# Herança

- ◆ de atributos
  - atributos da superclasse são herdados pelas subclasses
- ◆ de relacionamentos
  - instâncias de relacionamento da superclasse são herdados pelas entidades das subclasses
- ◆ Observação
  - qualquer entidade membro de uma subclasse deve ser também membro da superclasse
  - qualquer entidade membro da superclasse pode ser opcionalmente incluída como membro de qualquer número de subclasses

# Generalização/Especialização

## ◆ Especialização

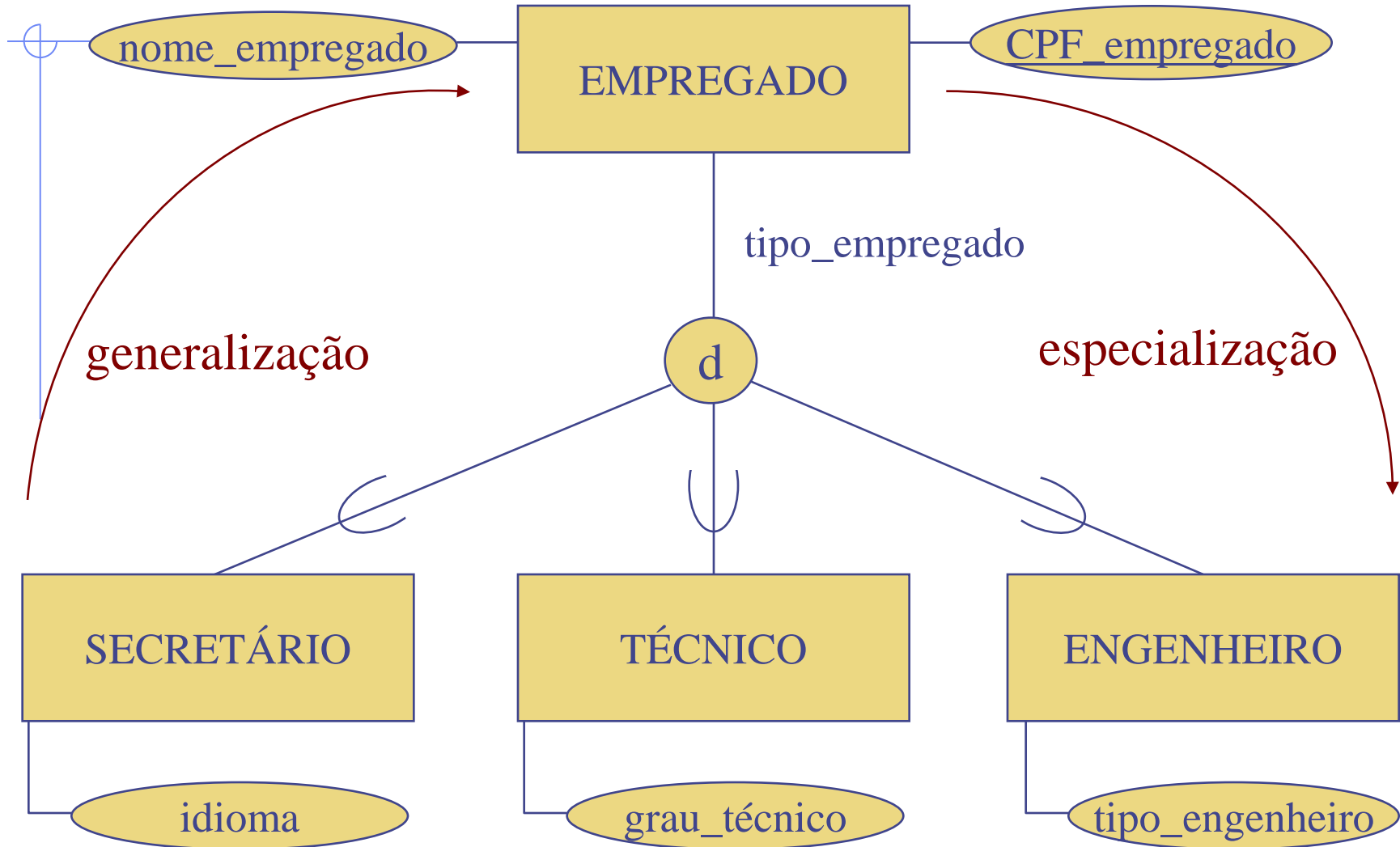
- resultado da separação de um tipo-entidade de nível mais alto (superclasse), formando vários tipos-entidade de nível mais baixo (subclasse)
- passos:
  - ◆ define-se um conjunto de subclasses de um tipo-entidade
  - ◆ associa-se atributos adicionais específicos às subclasses
  - ◆ estabelece-se tipos-relacionamento adicionais específicos às subclasses, caso necessário

# Generalização/Especialização

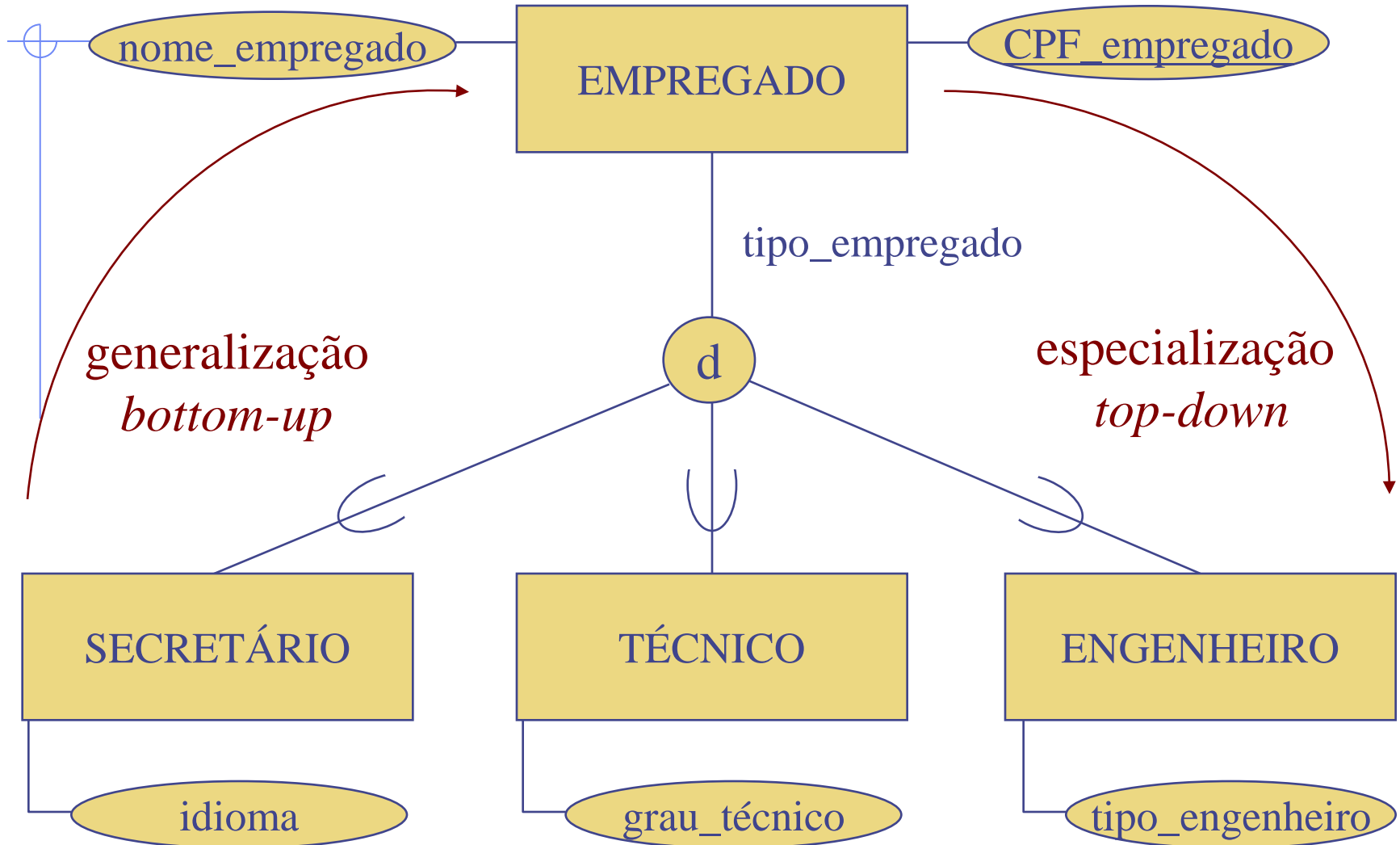
## ◆ Generalização

- resultado da união de dois ou mais tipos-entidade de nível mais baixo (subclasse), produzindo um tipo-entidade de nível mais alto (superclasse)
- é uma abstração de um conjunto de entidades
- passos:
  - ◆ suprime-se as diferenças entre os tipos-entidade
  - ◆ identifica-se os atributos em comum
  - ◆ generaliza-os em uma superclasse

# Representação

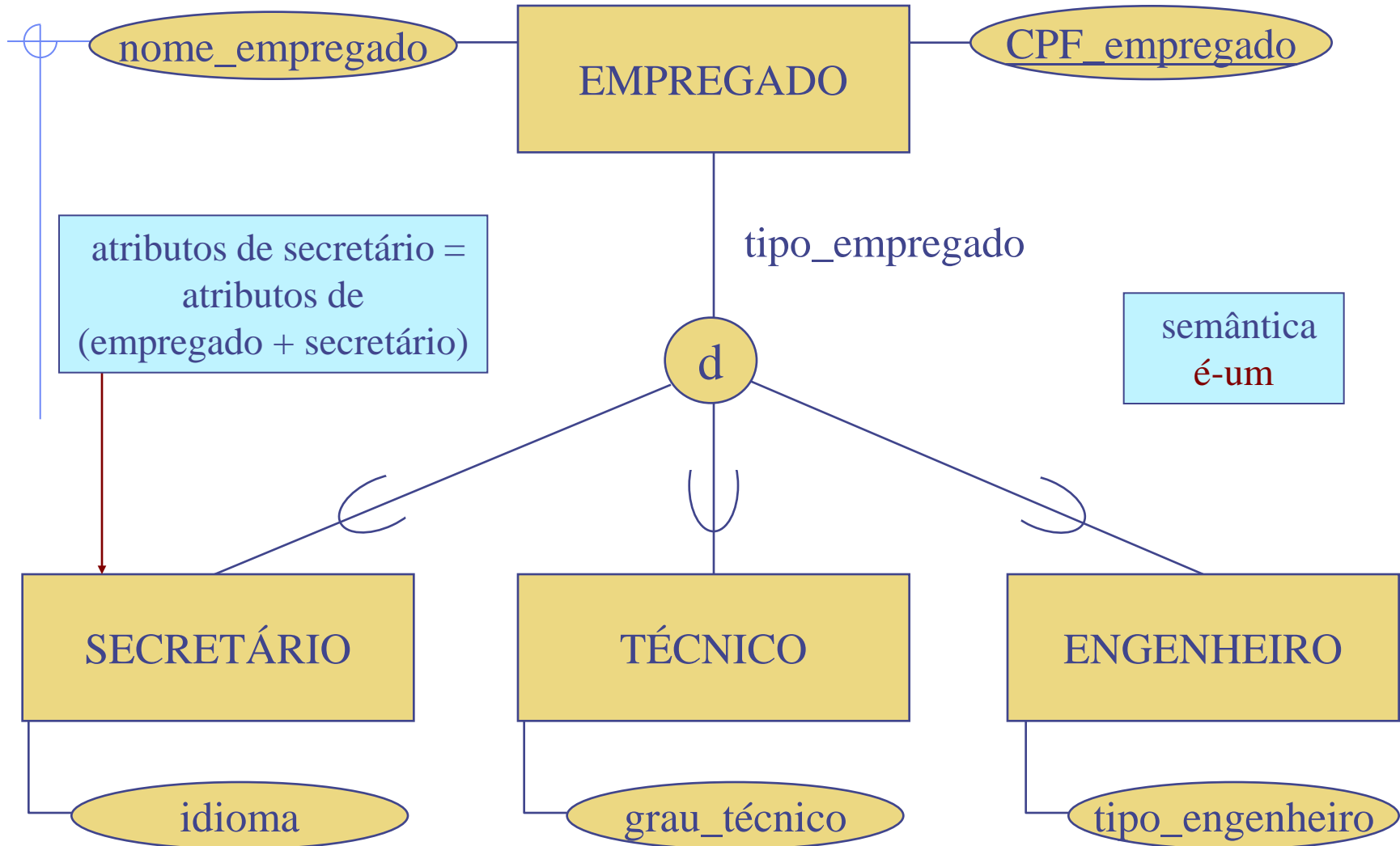


# Representação





# Representação




# Chaves dos Tipos-Entidade

- ◆ Restrição de chave do ME-R: todos os tipos-entidade devem ter uma chave única
  - Restrição relaxada para o MER-X
    - ◆ subclasses não precisam ter chave explicitamente definida

# Restrição de Disjunção

## ◆ Subclasses mutuamente exclusivas

- uma entidade de uma superclasse deve ser membro, quando muito, de apenas uma subclasse

■ representação:  ← "d" (disjoint)

## ◆ Subclasses que se sobrepõem

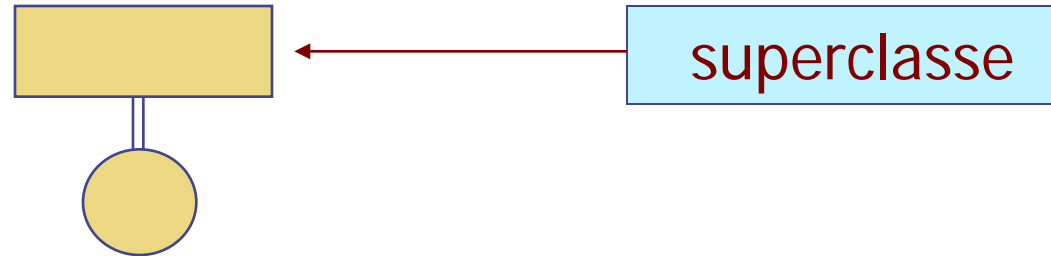
- uma entidade de uma superclasse pode ser membro de mais do que uma subclasse

■ representação:  ← "o" (overlap)

# Restrição de Completude

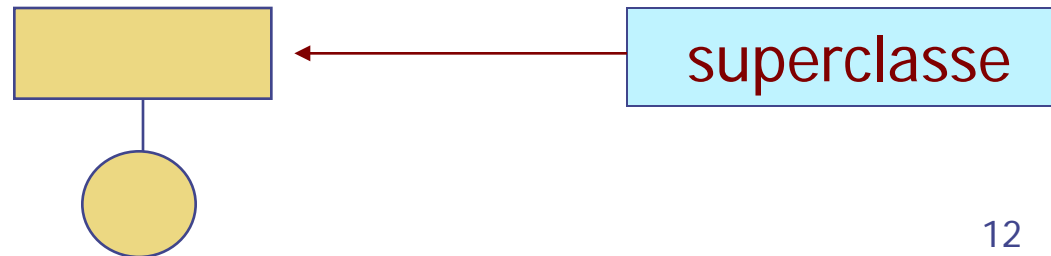
## ◆ Total

- cada entidade de uma superclasse deve ser membro de alguma subclasse na especialização
- representação:



## ◆ Parcial

- uma entidade de uma superclasse pode não pertencer a qualquer uma das subclasses
- representação:

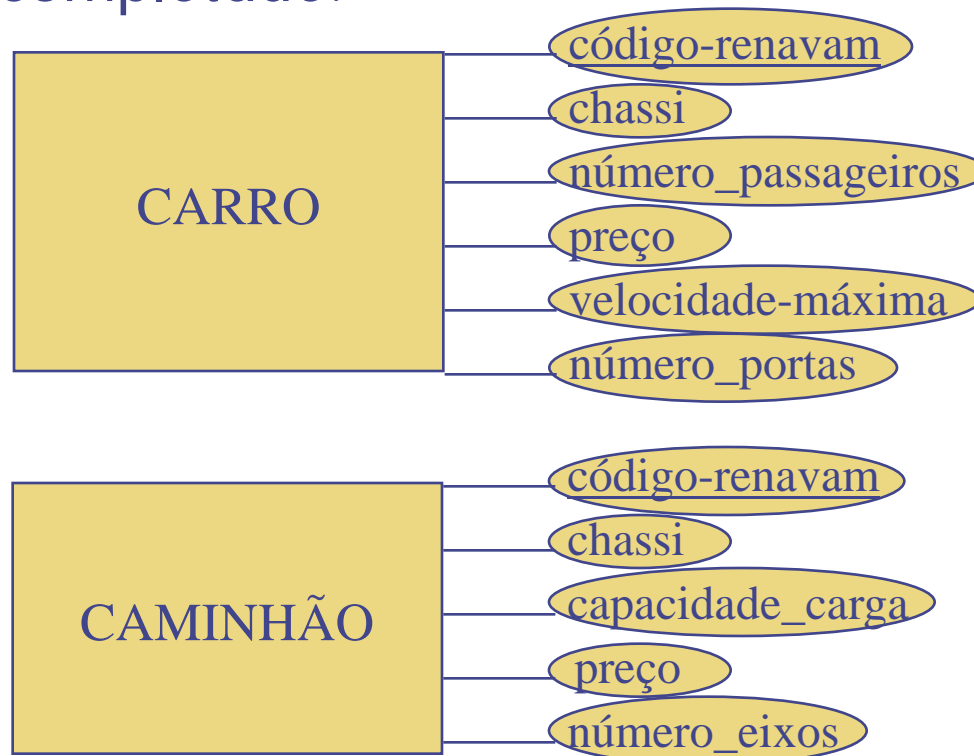


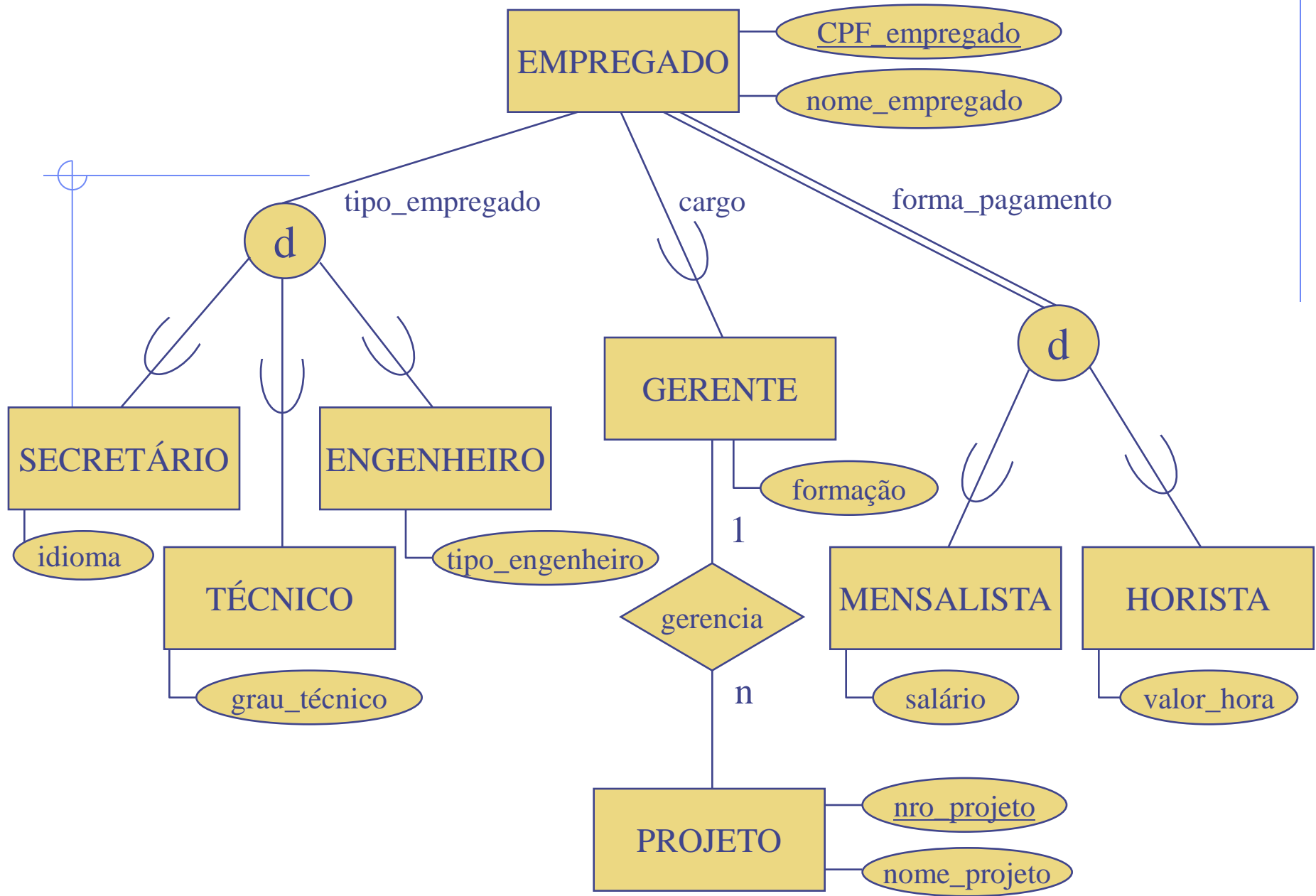
# Observações

- ◆ Restrições de disjunção e de completude são independentes
  - possibilidades de hierarquias
    - ◆ total disjunta
    - ◆ parcial disjunta
    - ◆ total com sobreposição
    - ◆ parcial com sobreposição

# Exercício

- ◆ Modele uma hierarquia de generalização/especialização para os tipos-entidade *carro* e *caminhão*. Defina as restrições de disjunção e de completude.





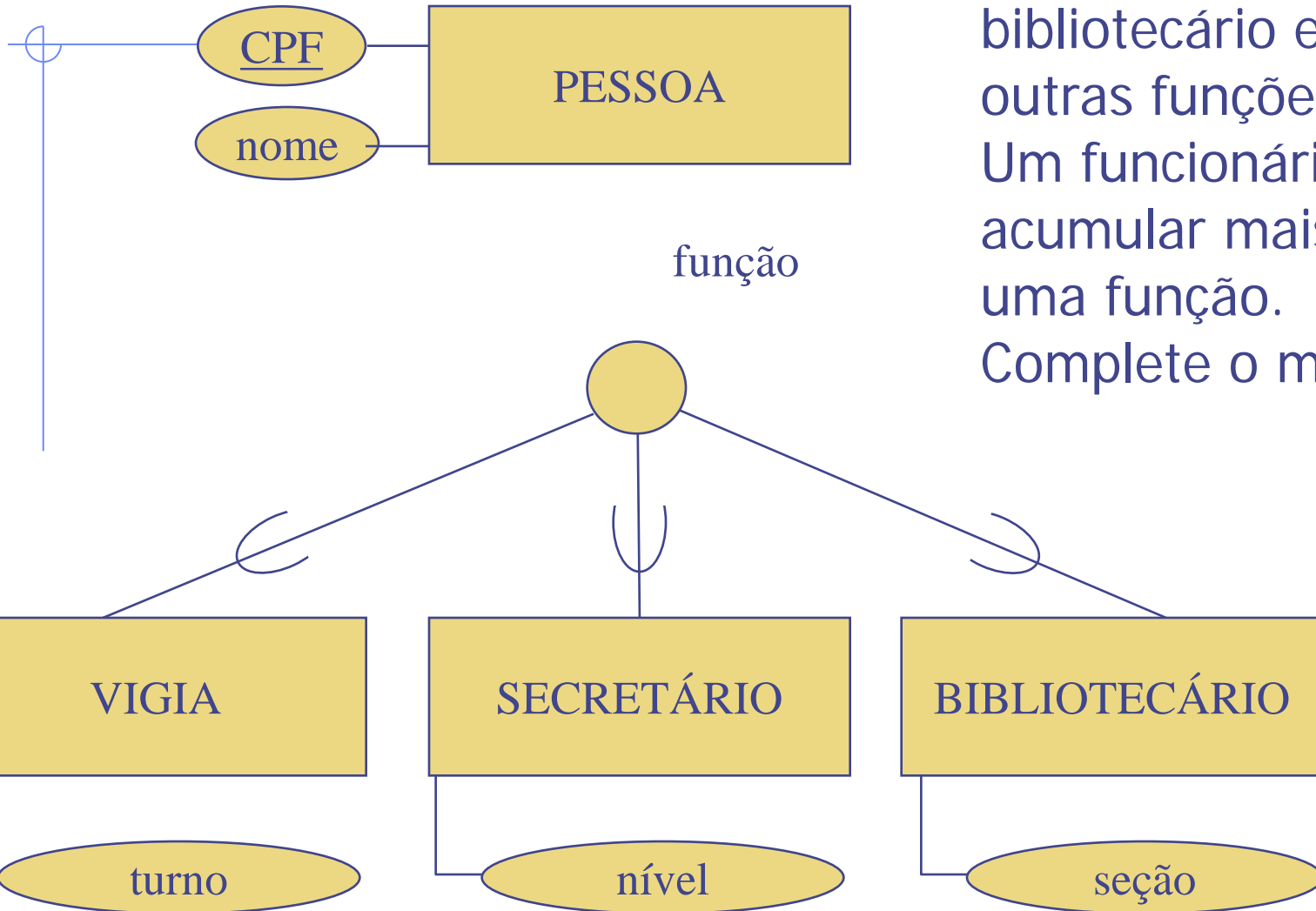
# Generalização/Especialização

- ◆ Uma subclasse pode possuir outras subclasses especificadas a partir dela
- ◆ Herança simples
  - cada subclasse participa como subclasse em **apenas um** relacionamento superclasse/subclasse
- ◆ Herança múltipla
  - cada subclasse pode participar como uma subclasse em **mais do que um** relacionamento superclasse/subclasse



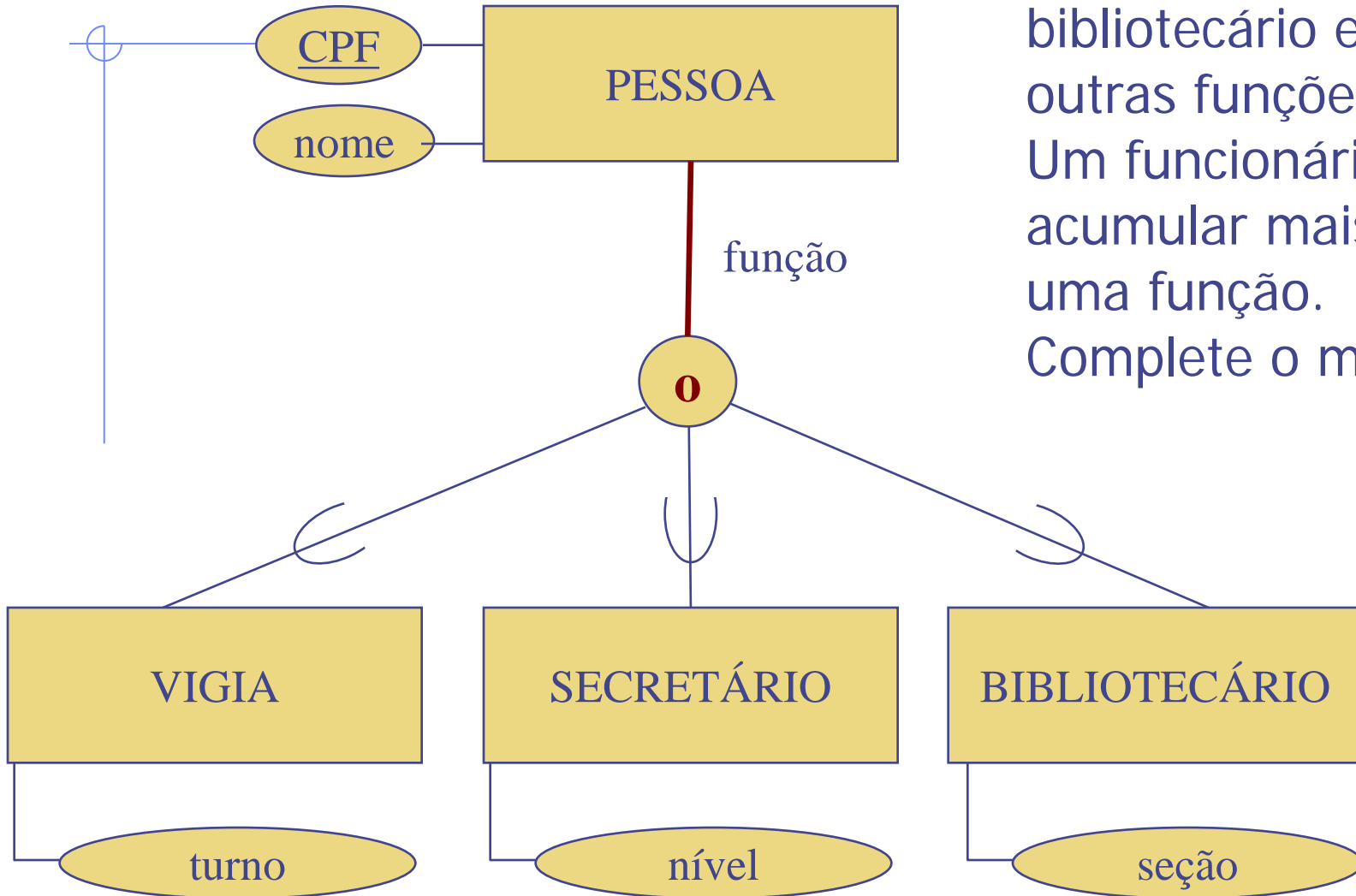
# Exercício

Além de vigia, secretário e bibliotecário existem outras funções. Um funcionário pode acumular mais de uma função. Complete o modelo!

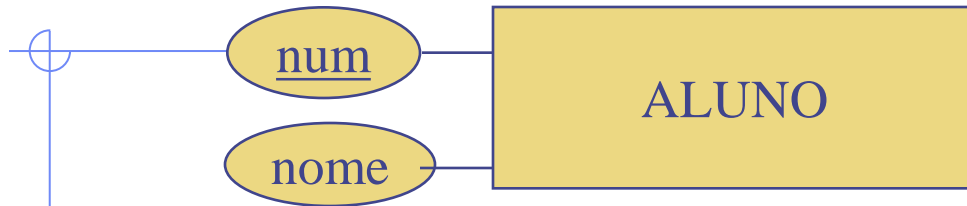


# Exercício

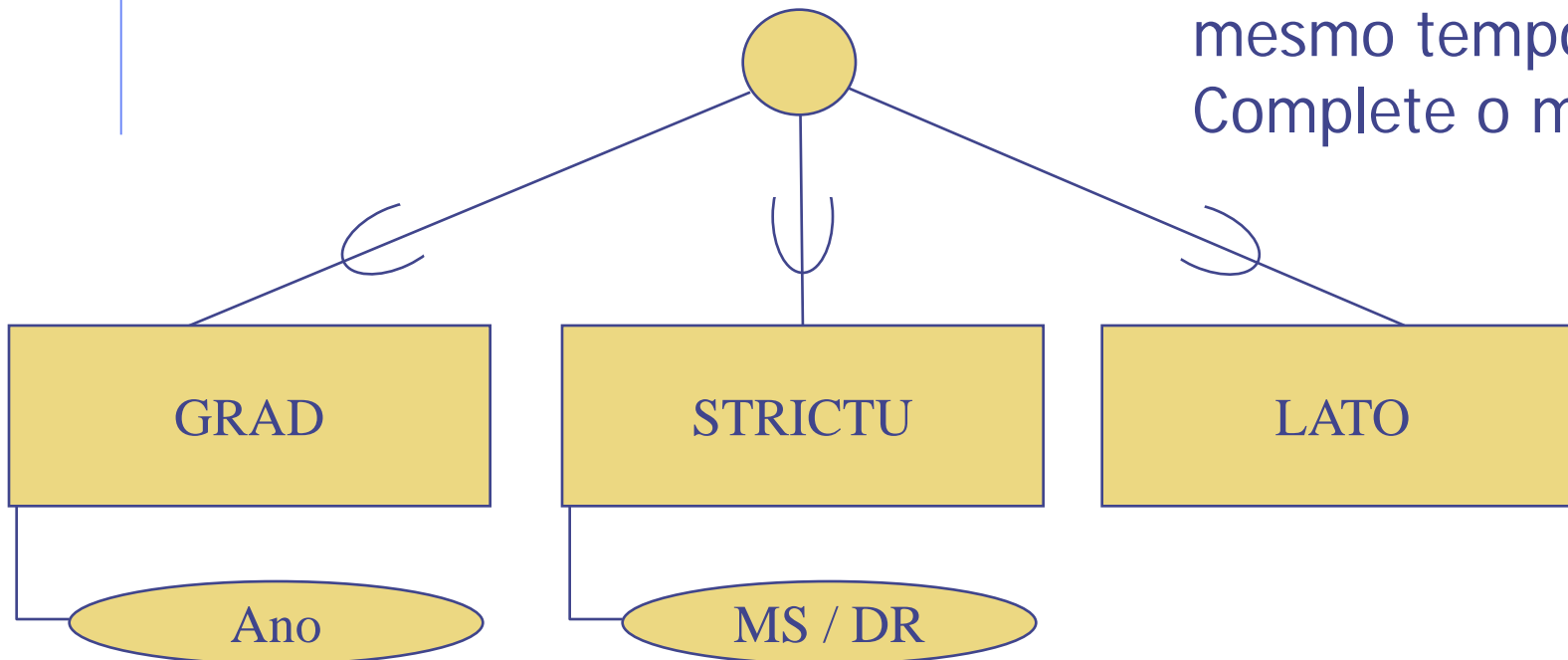
Além de vigia, secretário e bibliotecário existem outras funções. Um funcionário pode acumular mais de uma função. Complete o modelo!



# Exercício

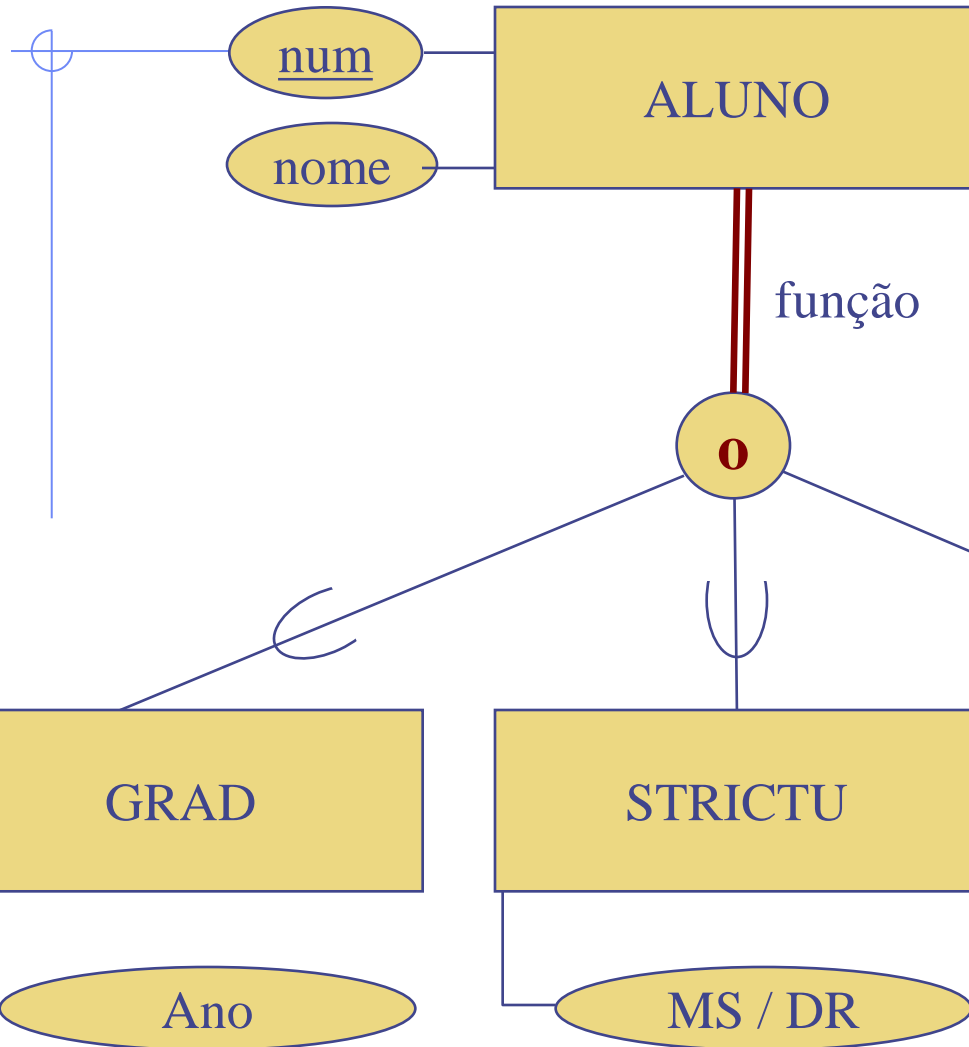


função



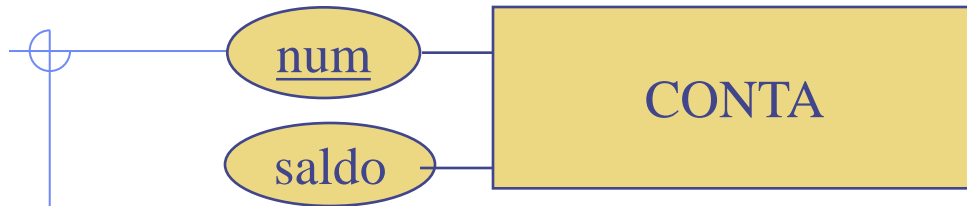
Somente existem alunos de graduação, strictu (mestrado ou doutorado) e lato (especialização). Um aluno pode cursar mais de um curso ao mesmo tempo. Complete o modelo!

# Exercício



Somente existem alunos de graduação, strictu (mestrado ou doutorado) e lato (especialização). Um aluno pode cursar mais de um curso ao mesmo tempo. Complete o modelo!

# Exercício

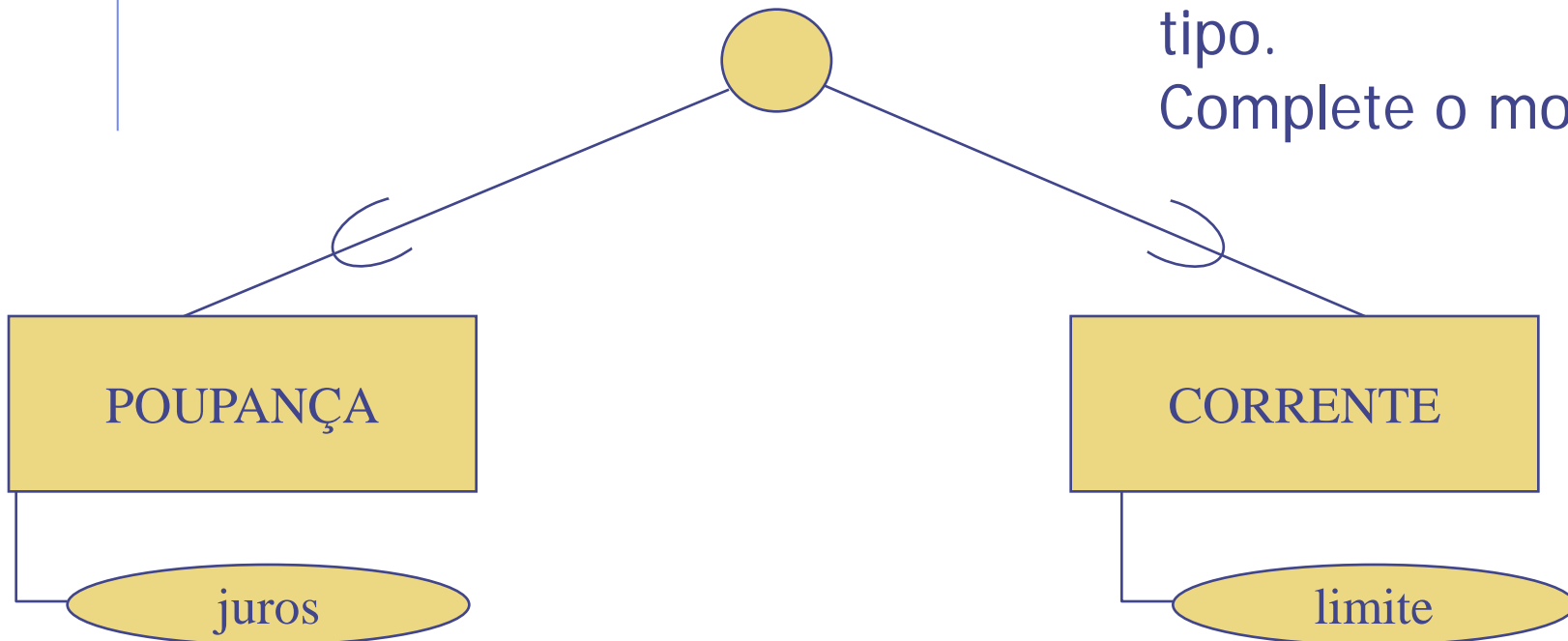


Existem contas que não são nem de poupança nem corrente.

Um conta não pode ser classificada como sendo mais de um tipo.

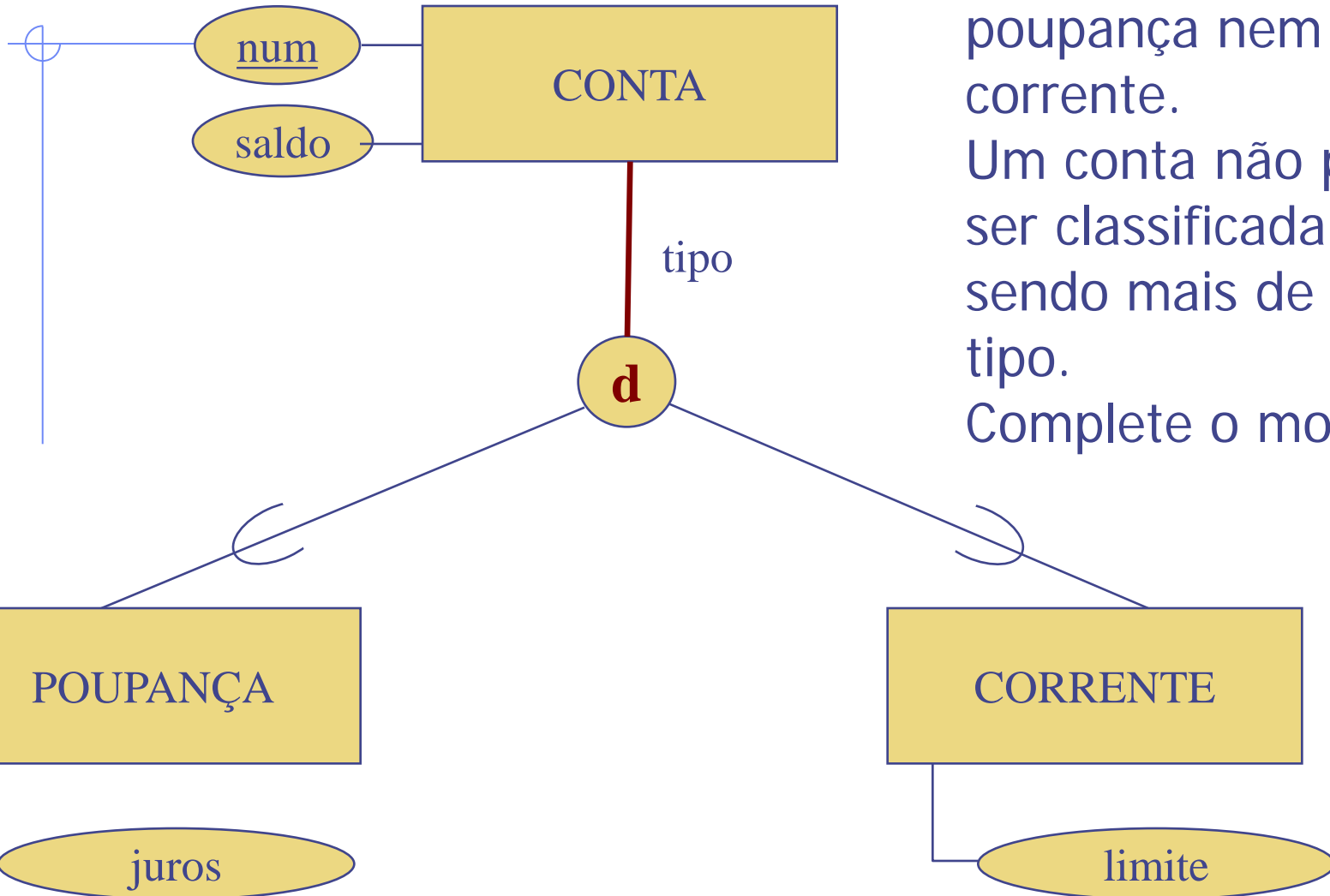
Complete o modelo!

tipo



# Exercício

Existem contas que não são nem de poupança nem corrente.  
Um conta não pode ser classificada como sendo mais de um tipo.  
Complete o modelo!



# Agregação

- ◆ É um conceito para a construção de objetos compostos a partir de seus objetos componentes
  - Idéia: elementos de modelagem podem associar-se, formando outros elementos que representam essa associação
- ◆ Pode assumir diversas formas:
  - Agregando atributos em Tipos-Entidade e Tipos-Relacionamento
    - ◆ os valores dos atributos compõem a entidade
  - Agregando Tipos-Entidade e Tipos-Relacionamentos
    - ◆ combinar entidades que estão relacionadas por uma instância de relacionamento em uma entidade agregada de alto nível

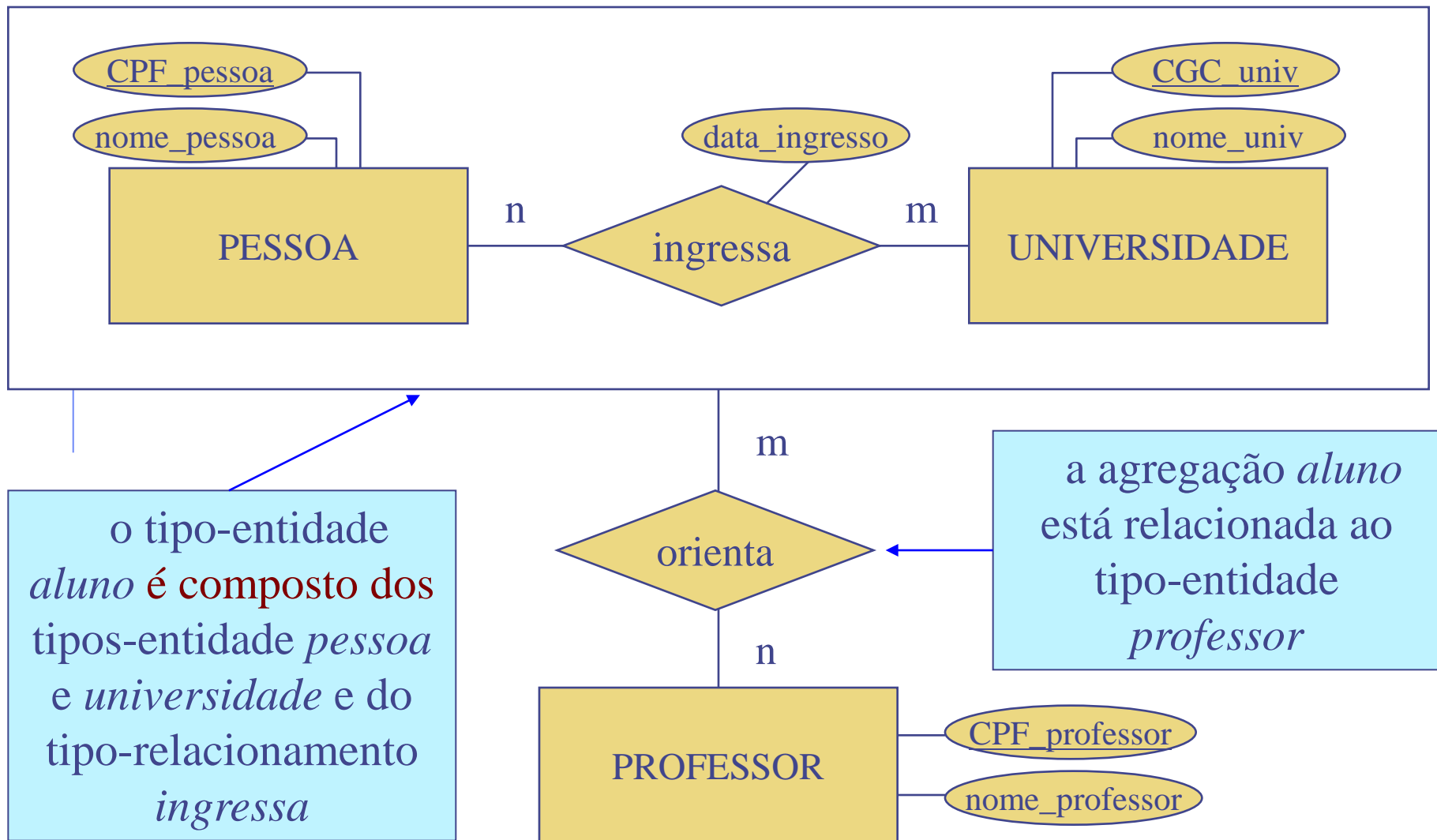
# Agregação

- ◆ Tipos-entidades agregados são representados como tipos-entidades comuns
- ◆ Engloba
  - dois tipos-entidades e um tipo-relacionamento



# Representação

ALUNO

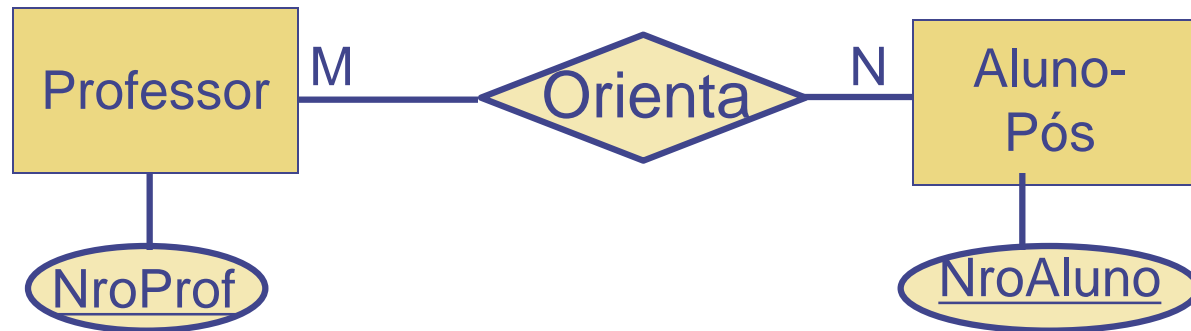


# Agregação

- ◆ Situações que indicam a necessidade de agregação:
  - 1) Quando é necessário identificar cada relacionamento (o relacionamento tem chave)
  - 2) Quando é necessário mais de um relacionamento envolvendo as mesmas entidades
  - 3) Quando existe a necessidade de associar dois relacionamentos

# Agregação

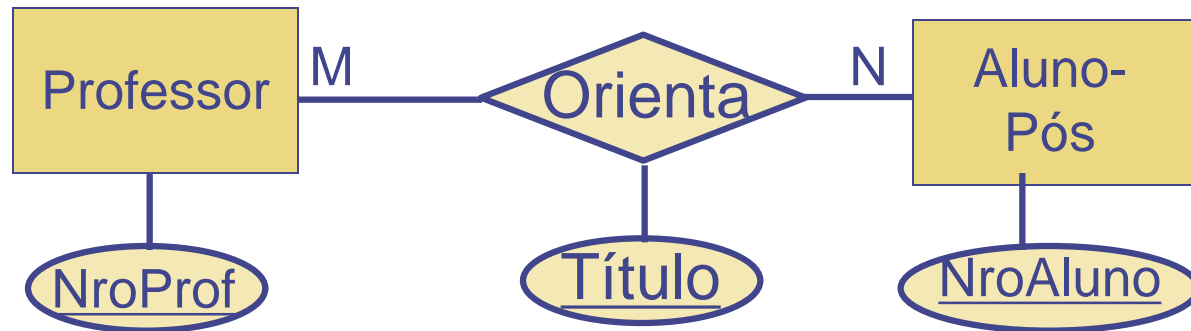
- ◆ **1o. Caso:** O tipo relacionamento tem um identificador próprio:
  - Nesse caso, embora seja possível identificar a entidade agregação por um identificador próprio, ela também pode ser identificada pelo relacionamento entre as entidades que participam do relacionamento:



Título único em todo o sistema

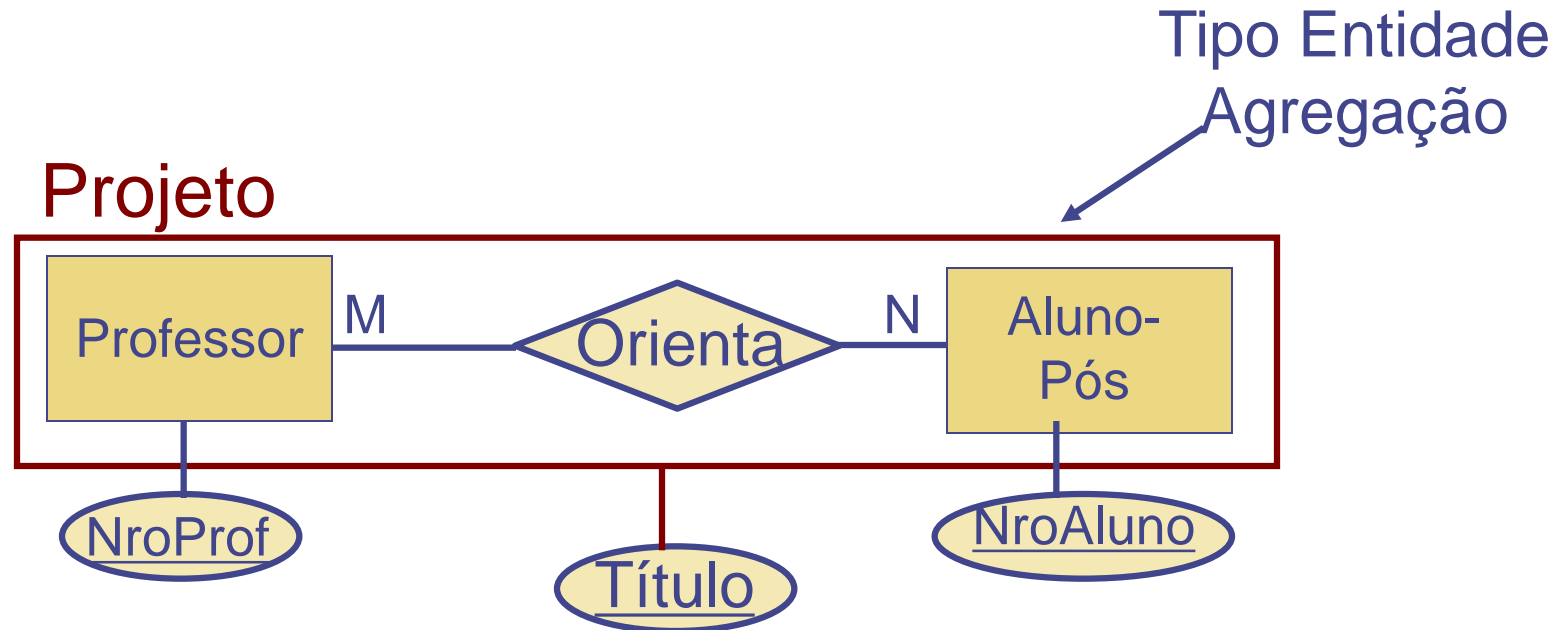
# Agregação

- ◆ **1o. Caso:** O tipo relacionamento tem um identificador próprio:
  - Nesse caso, embora seja possível identificar a entidade agregação por um identificador próprio, ela também pode ser identificada pelo relacionamento entre as entidades que participam do relacionamento:



***Um Tipo-Relacionamento não pode ter atributo identificador!***

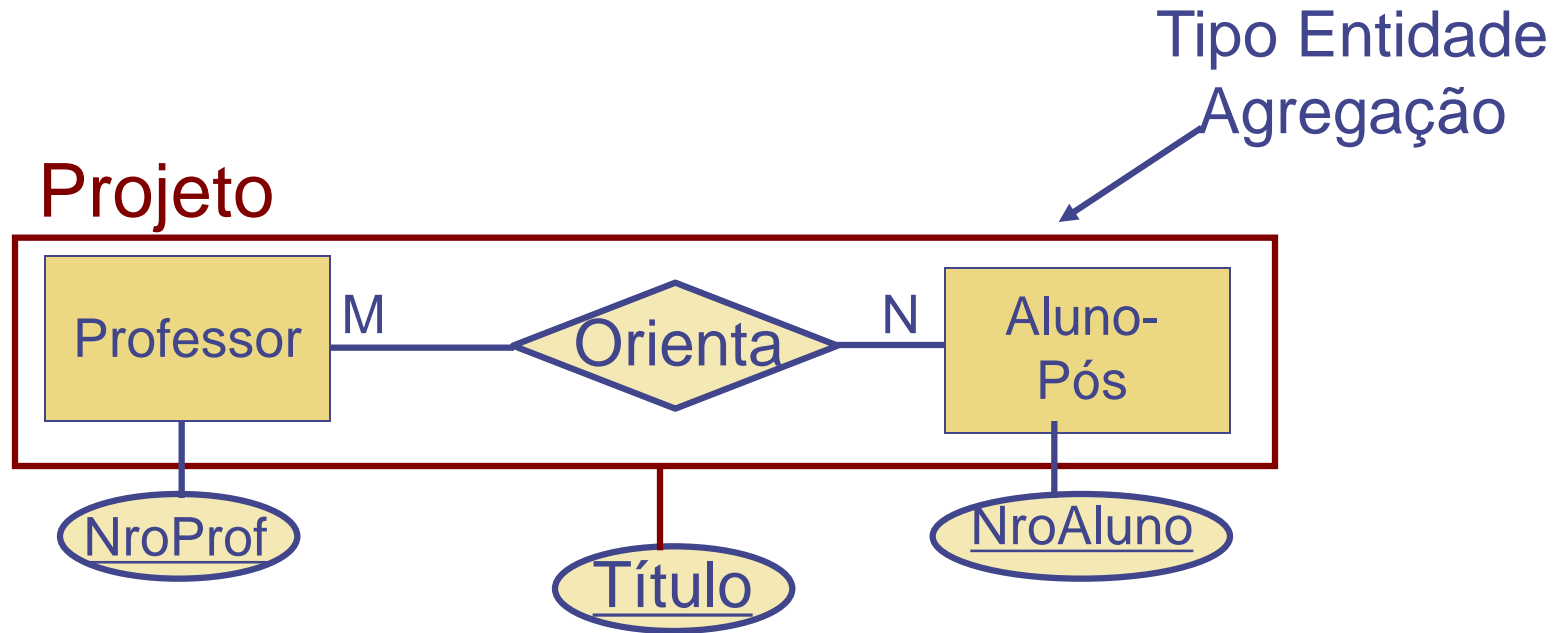
# Agregação



Chave:

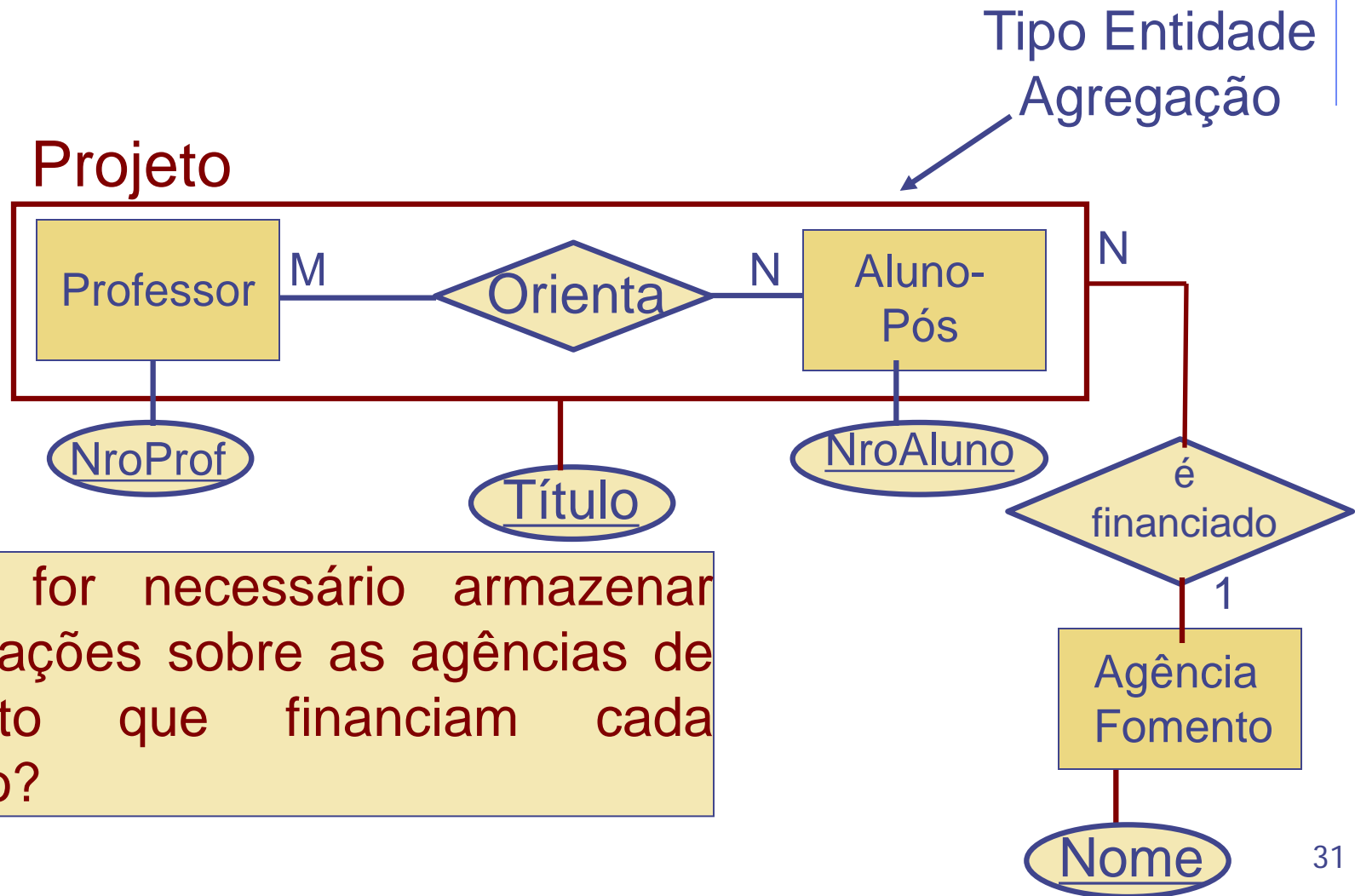
- NroProfessor + NroAluno
- ou
- Título

# Agregação



E se for necessário armazenar informações sobre as agências de fomento que financiam cada projeto?

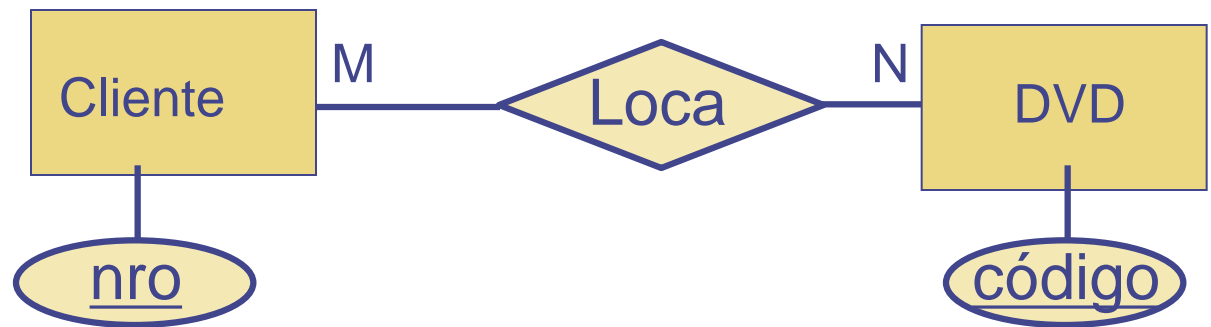
# Agregação



E se for necessário armazenar informações sobre as agências de fomento que financiam cada projeto?

# Agregação

- ◆ **2o. Caso:** Pode haver mais de um relacionamento envolvendo as mesmas entidades
  - Como identificar cada locação?



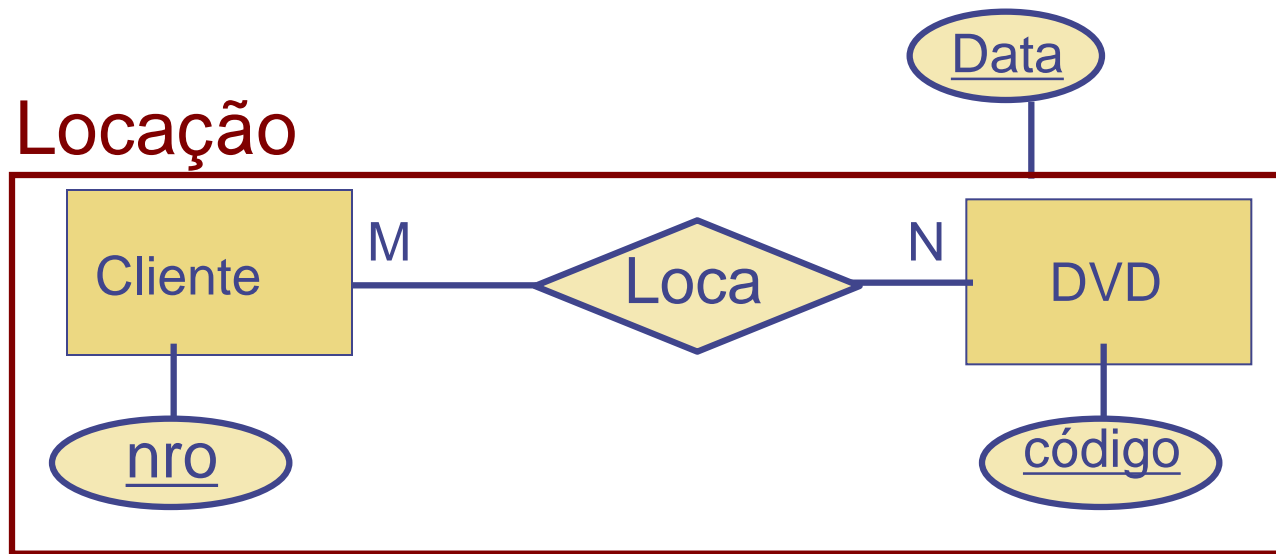


# Agregação

- ◆ **2o. Caso:** Pode haver mais de uma instância de relacionamento envolvendo as mesmas entidades
  - Como identificar cada locação?

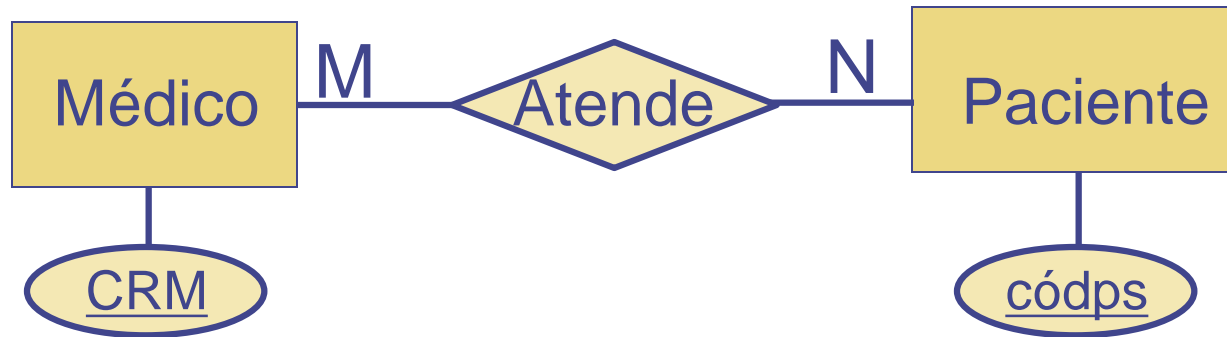
Chave:

- nro +
- código +
- data



# Agregação

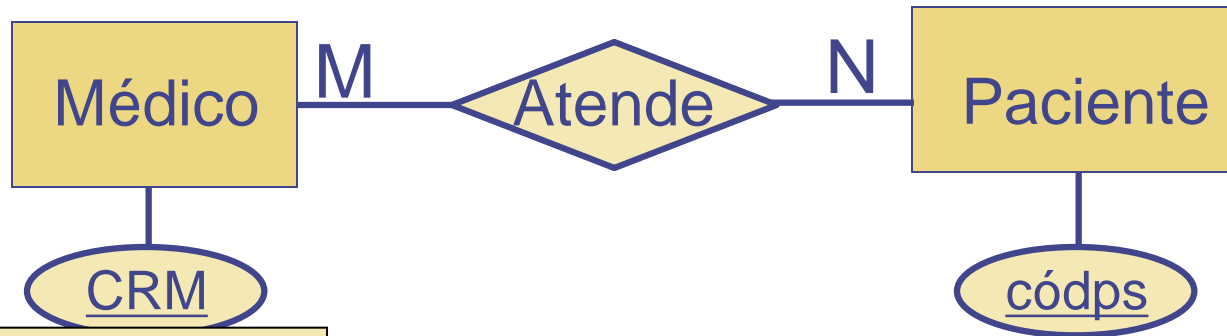
- ◆ **3o. Caso:** Pode haver a necessidade de associar dois tipos relacionamentos



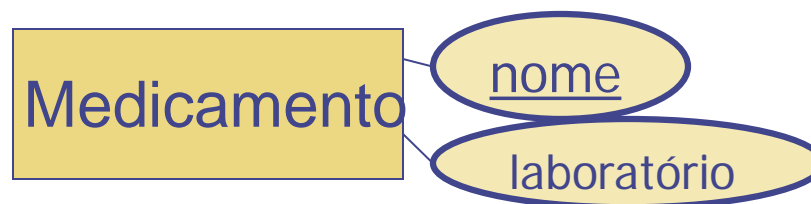
- Suponha que seja necessário alterar esse modelo do seguinte modo: é necessário saber que medicamentos existem e que medicamentos foram prescritos em cada consulta.

# Agregação

- ◆ **3o. Caso:** Pode haver a necessidade de associar dois tipos relacionamentos



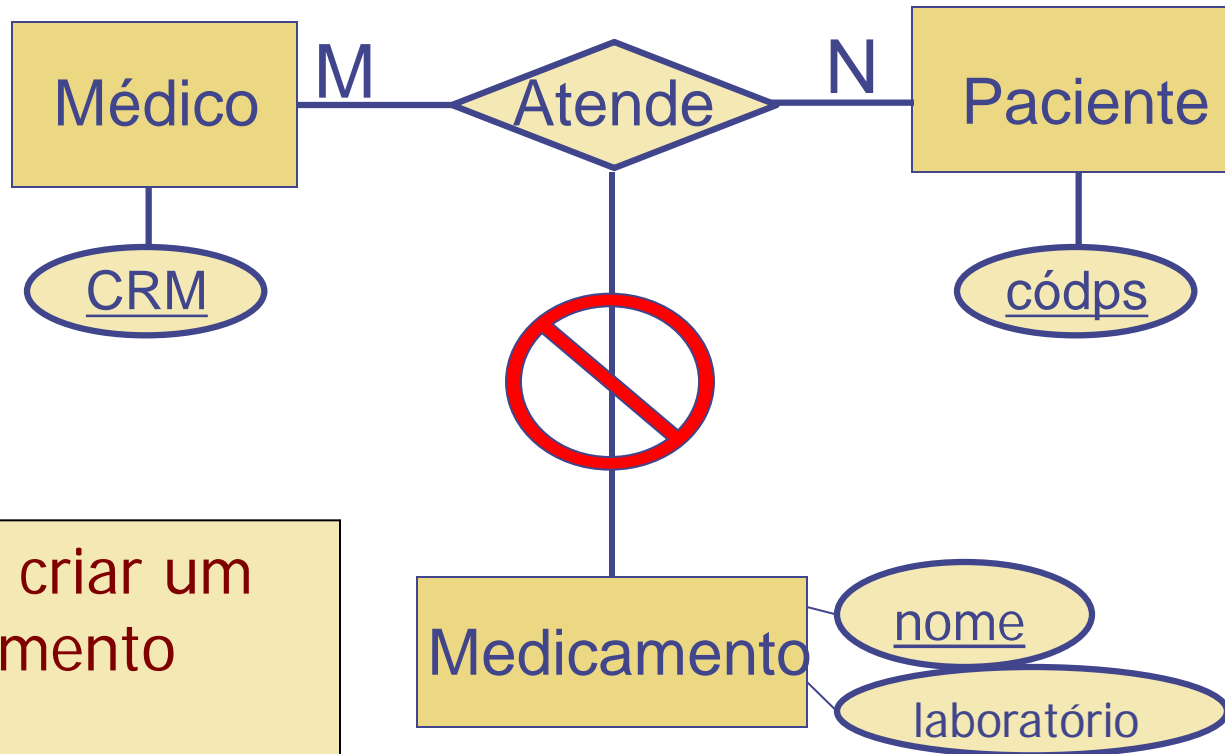
- A questão agora é:  
com que tipo  
entidade existente  
o novo tipo  
entidade deve ser  
associado?



# Agregação

- Relacionamento Ternário?
- uma instância do relacionamento teria 1 médico atendendo 1 paciente e receitando 1 medicamento

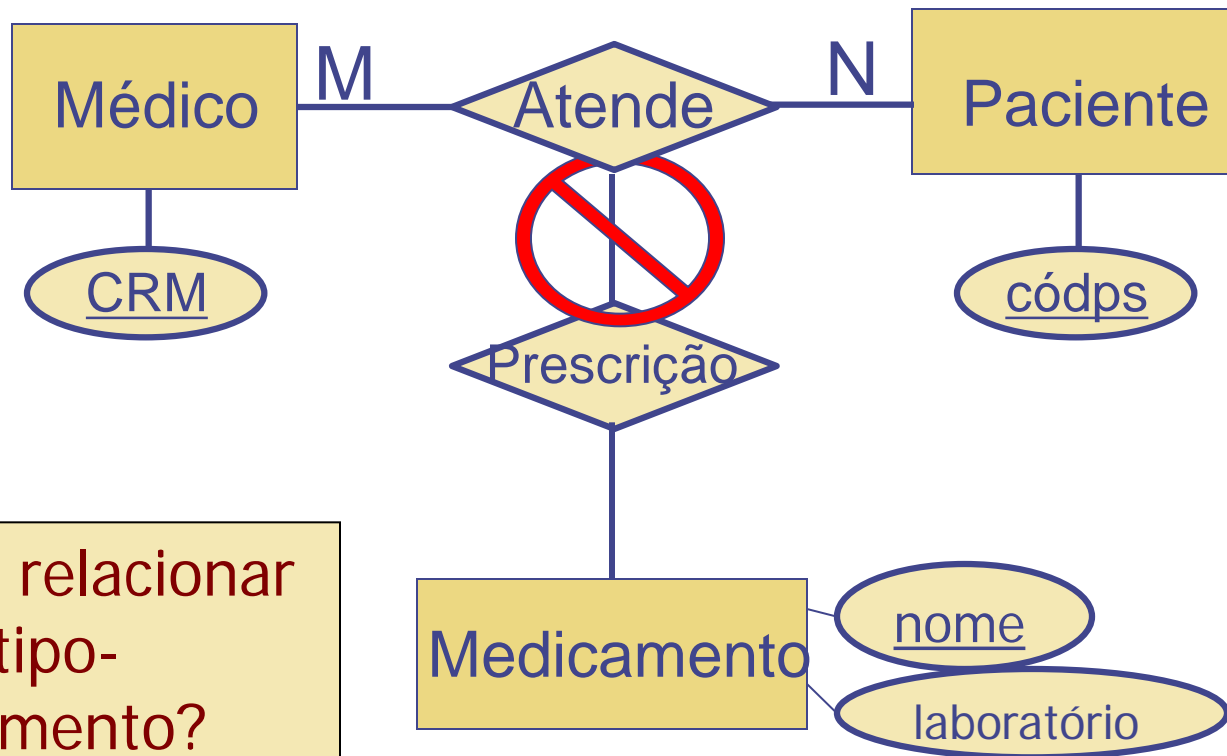
◆ **3o. Caso:** Pode haver a necessidade de associar dois tipos relacionamentos



- Podemos criar um relacionamento ternário?

# Agregação

- ◆ **3o. Caso:** Pode haver a necessidade de associar dois tipos relacionamentos

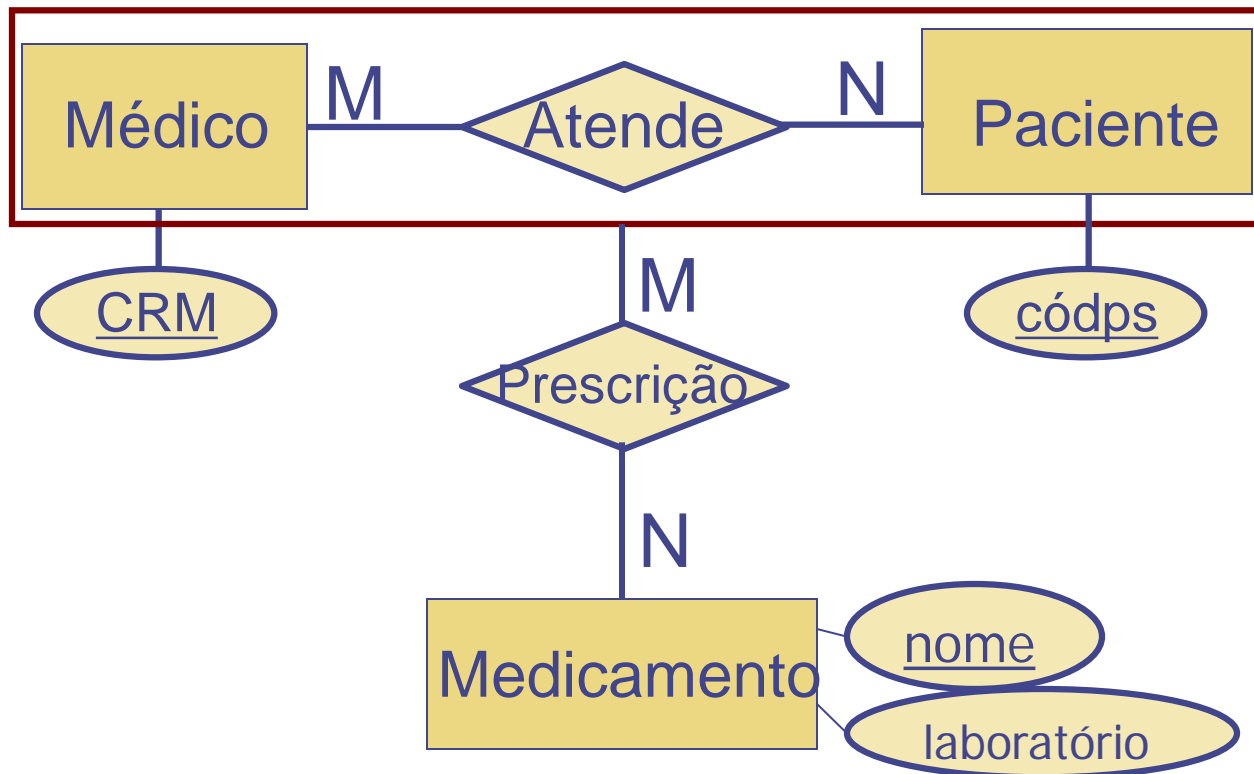


- Podemos relacionar com um tipo-relacionamento?

# Agregação

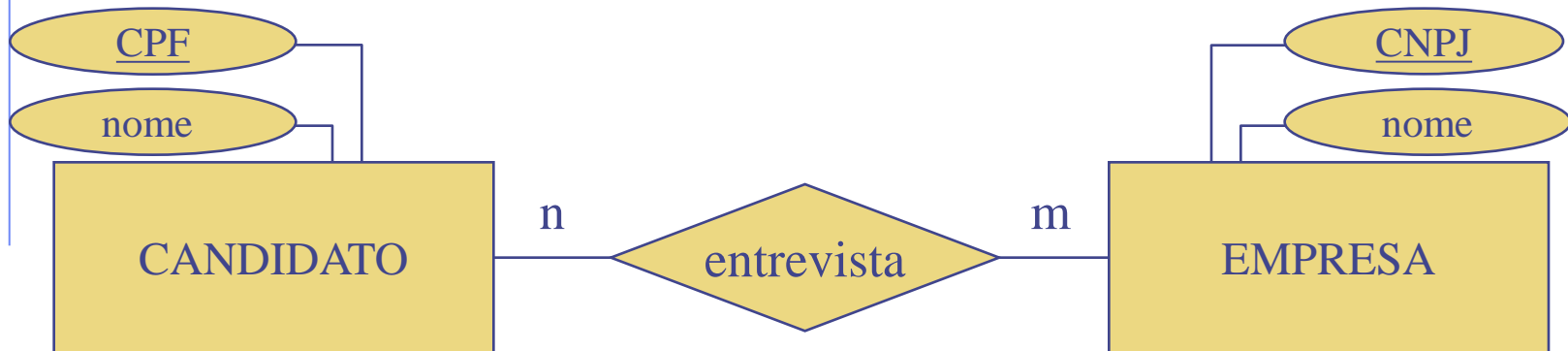
- ◆ **3o. Caso:** Pode haver a necessidade de associar dois tipos relacionamentos

## Consulta



# Exercício

◆ Considere o esquema ER abaixo:

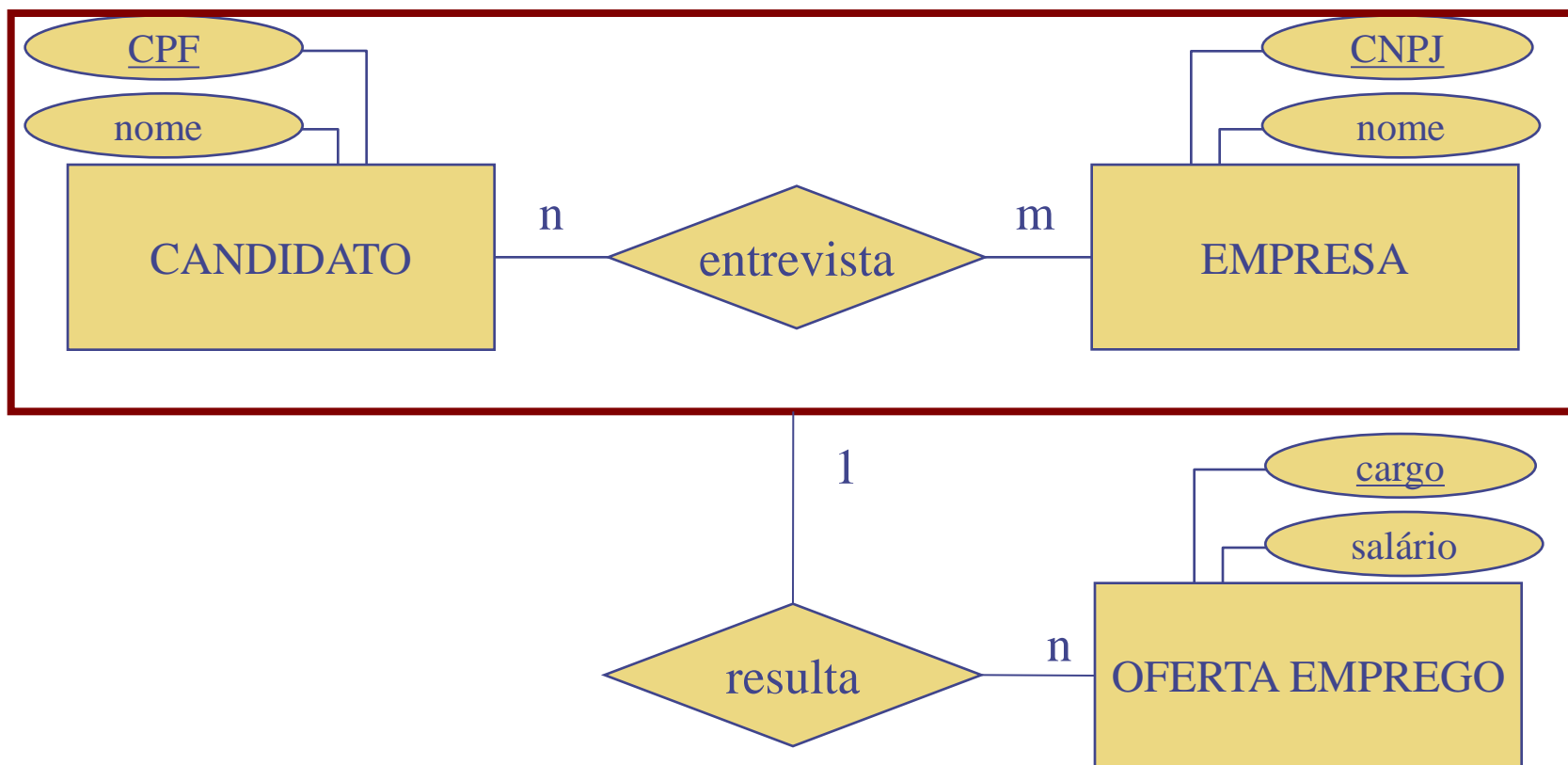


- Como modelar a situação em que algumas entrevistas resultam em uma oferta de emprego (com cargo e salário inicial) e outras não?

# Exercício

◆ Resposta:

SELEÇÃO





# Passos para o Projeto Lógico de BD

1. Classificar tipos-entidades e atributos
  - tipos-entidade possuem informações descritivas, atributos não
  - atributos devem ser mantidos de forma atômica
  - atributos devem ser relacionados às entidades que eles descrevem
2. Identificar chaves primárias

# Projeto Lógico de BD

## 3. Identificar tipos-relacionamentos e seus atributos

- determinar o grau dos tipos-relacionamentos
  - ◆ definir tipos-relacionamento ternários cuidadosamente
- identificar as restrições que se aplicam sobre cada tipo-relacionamento
  - ◆ cardinalidade
  - ◆ participação
- Caso necessário, definir os papéis

## 4. Identificar tipo-entidade forte e tipo-entidade fraca

# Projeto Lógico de BD

5. Verificar os requisitos de operações
  - se eles se referirem a dados que não estão modelados, repetir os passos anteriores.
6. Modelar hierarquias de generalização
  - identificar atributos e relacionamentos comuns
  - determinar as restrições de disjunção e de completude
7. Modelar agregações

*“Podemos notar que um projetista de BD necessita de um bom conhecimento do minimundo que está sendo modelado para que possa tomar essas decisões!”*

# Leitura complementar para casa

- ◆ Capítulo “O modelo entidade-relacionamento estendido” do livro:  
Elmasri, Ramez; Navathe, Shamkant B.  
Sistemas de banco de dados
  - 6ª edição: capítulo 8