



# Modelo entidade-relacionamento estendido

- Capítulo 8: modelo Entidade-Relacionamento Estendido (EER)

# Modelo Entidade Relacionamento Extendido - EER

- **Características**

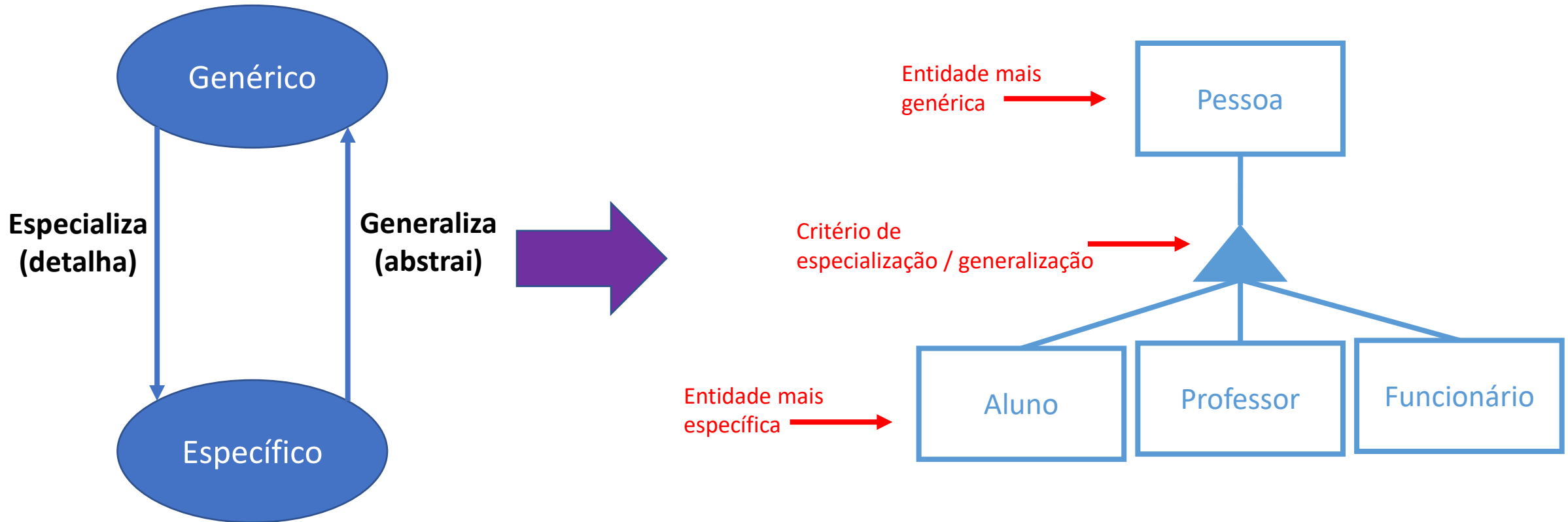
- Criado para projetar **esquemas de banco de dados mais precisos**
  - Reflete as propriedades de dados e restrições com mais precisão
- Requisitos mais complexos do que as aplicações mais tradicionais

- **Conceitos**

- Subtipo ou subclasse, supertipo ou superclasse
- Generalização, especialização



# Generalização / especialização



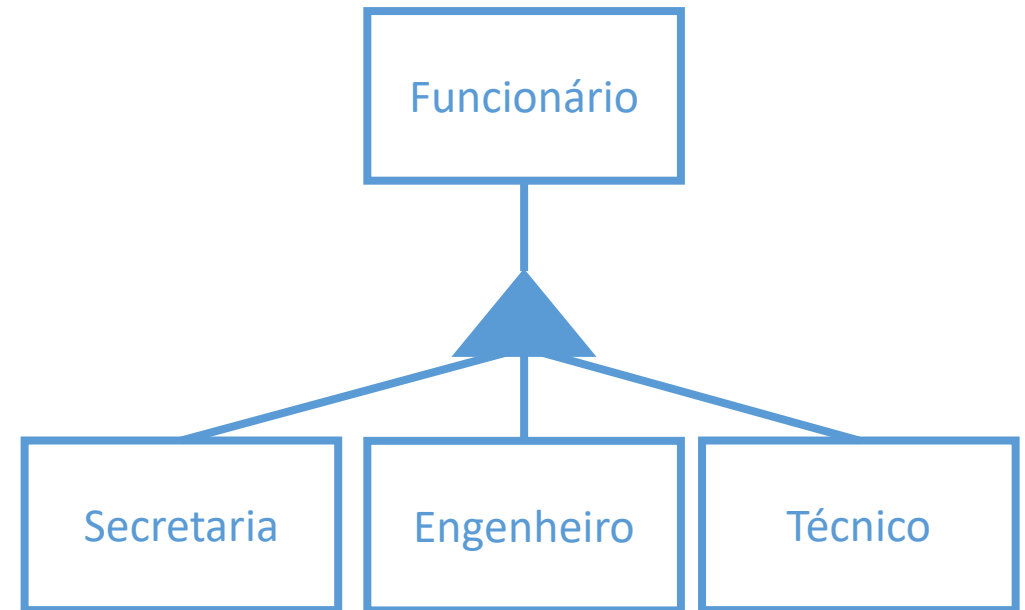
**Os conceitos superclasse (supertipo), subclasse (subtipo), herança, generalização, especialização estão intimamente relacionados**



# Subclasse / superclasse

- **Subclasse:**

- Subagrupamento das entidades de um tipo-entidade
- Exemplo:
  - **Superclasse (supertipo):** tipo entidade FUNCIONÁRIO
  - **Subclasse (subtipo):** SECRETARIO, ENGENHEIRO, TÉCNICO, ...



Cada entidade que é membro de qualquer uma das subclasses também **é um** FUNCIONÁRIO



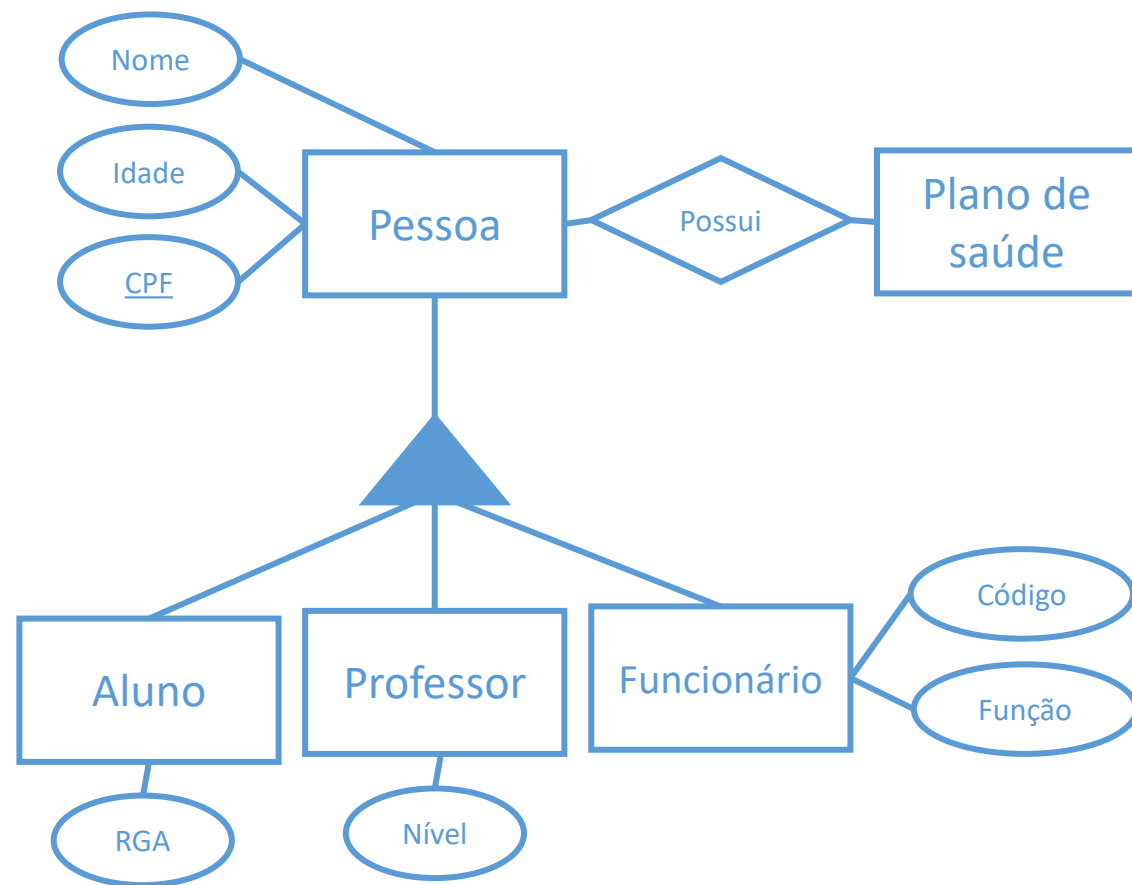
# Herança

- **De atributos:**

- Atributos da *superclasse* **são herdados** pelas *subclasses*

- **De relacionamentos:**

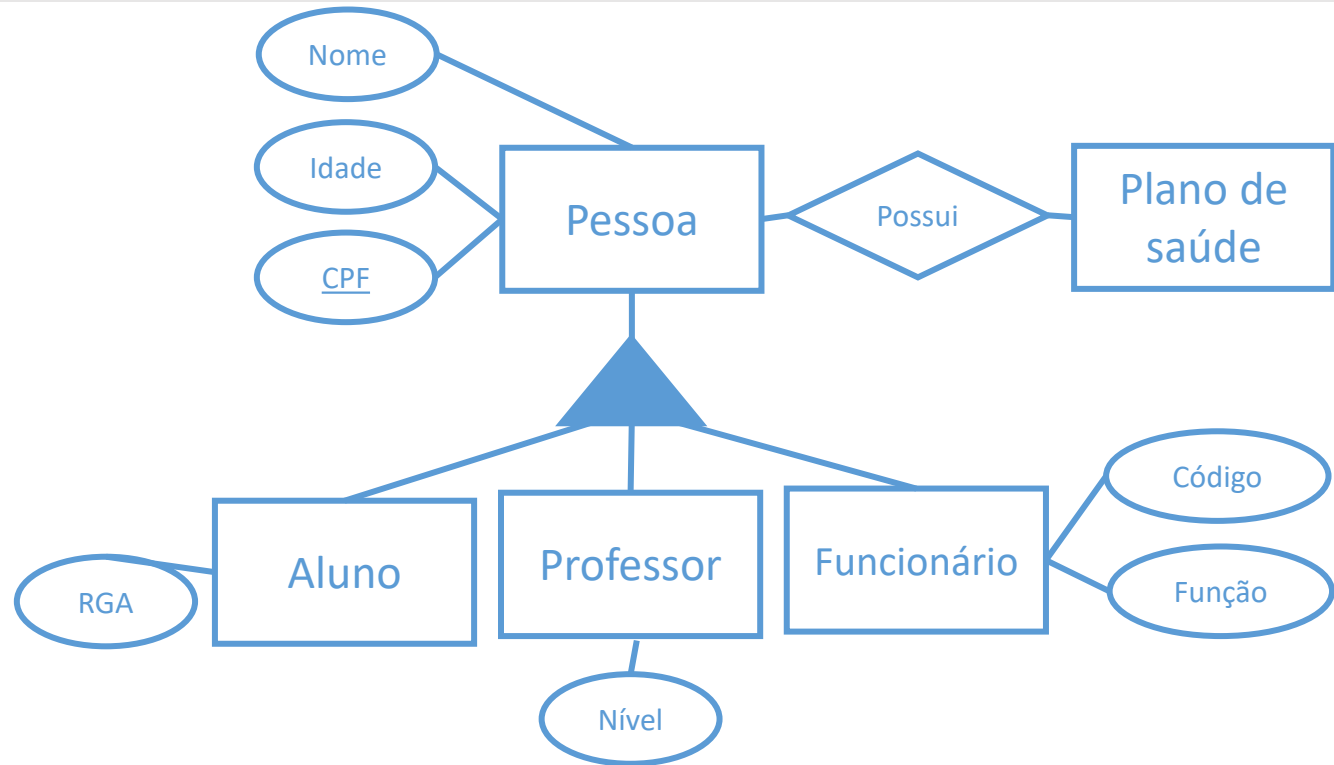
- Instâncias de relacionamentos da *superclasse* **são herdados** pelas entidades das *subclasses*



# Herança (hierarquia)

## Hierarquia

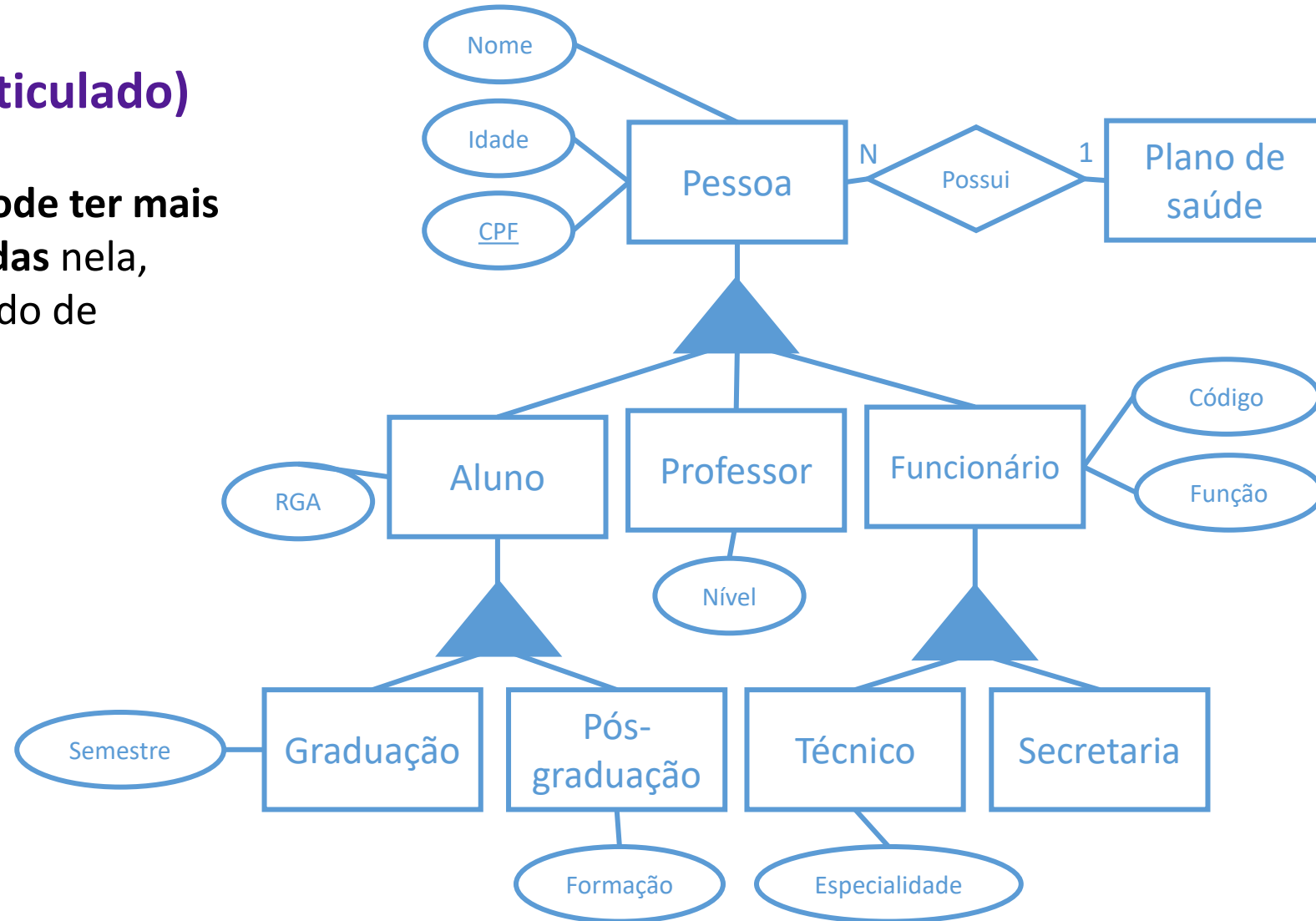
Cada **subclasse** tem apenas um **pai** que resulta em uma estrutura de árvore ou hierarquia estrita



# Herança múltipla (reticulado)

## Herança múltipla (reticulado)

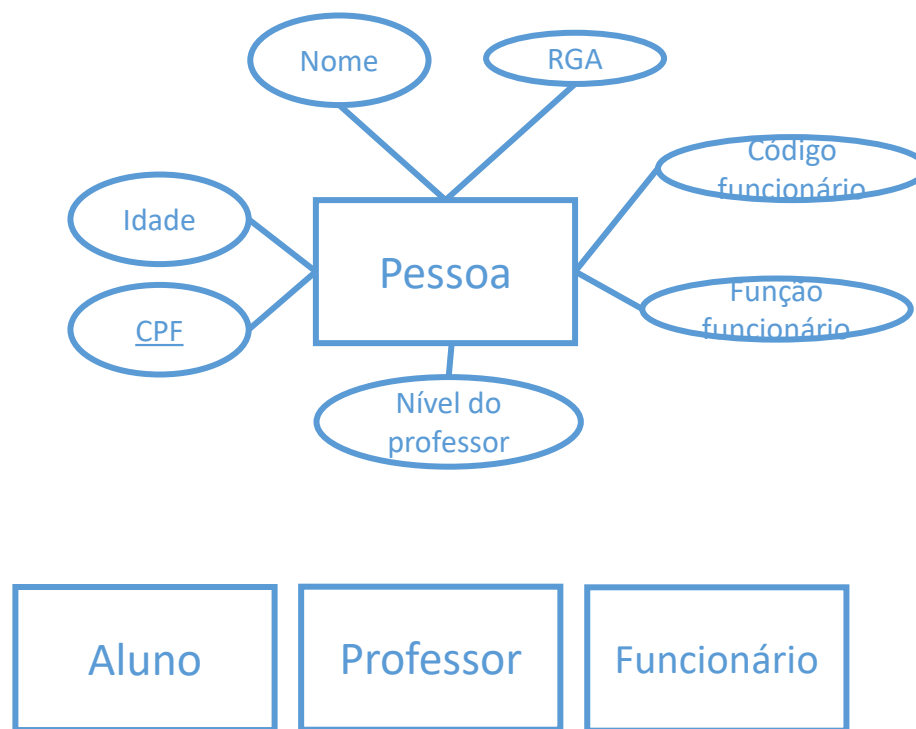
- A própria **subclasse pode ter mais subclasses especificadas** nela, formando um reticulado de especializações.



# Generalização / especialização

- **Especialização**

- É o processo de definir um **conjunto de subclasses** de um tipo entidade
- Passos:
  1. define-se um conjunto de **subclasses** de um tipo-entidade
  2. associa-se **atributos** adicionais **específicos** às **subclasses**
  3. estabelece-se **tipos-relacionamento** adicionais **específicos** às subclasses, caso necessário

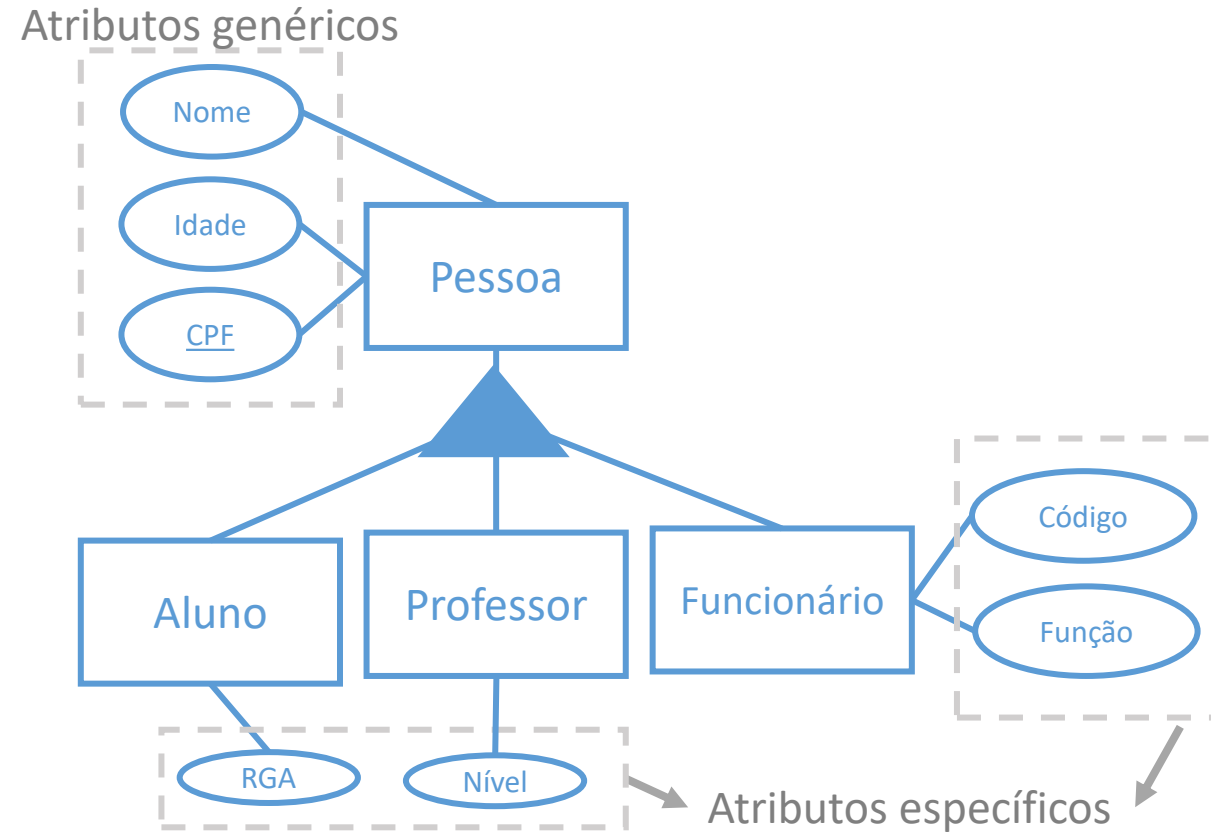




# Generalização / especialização

- **Especialização**

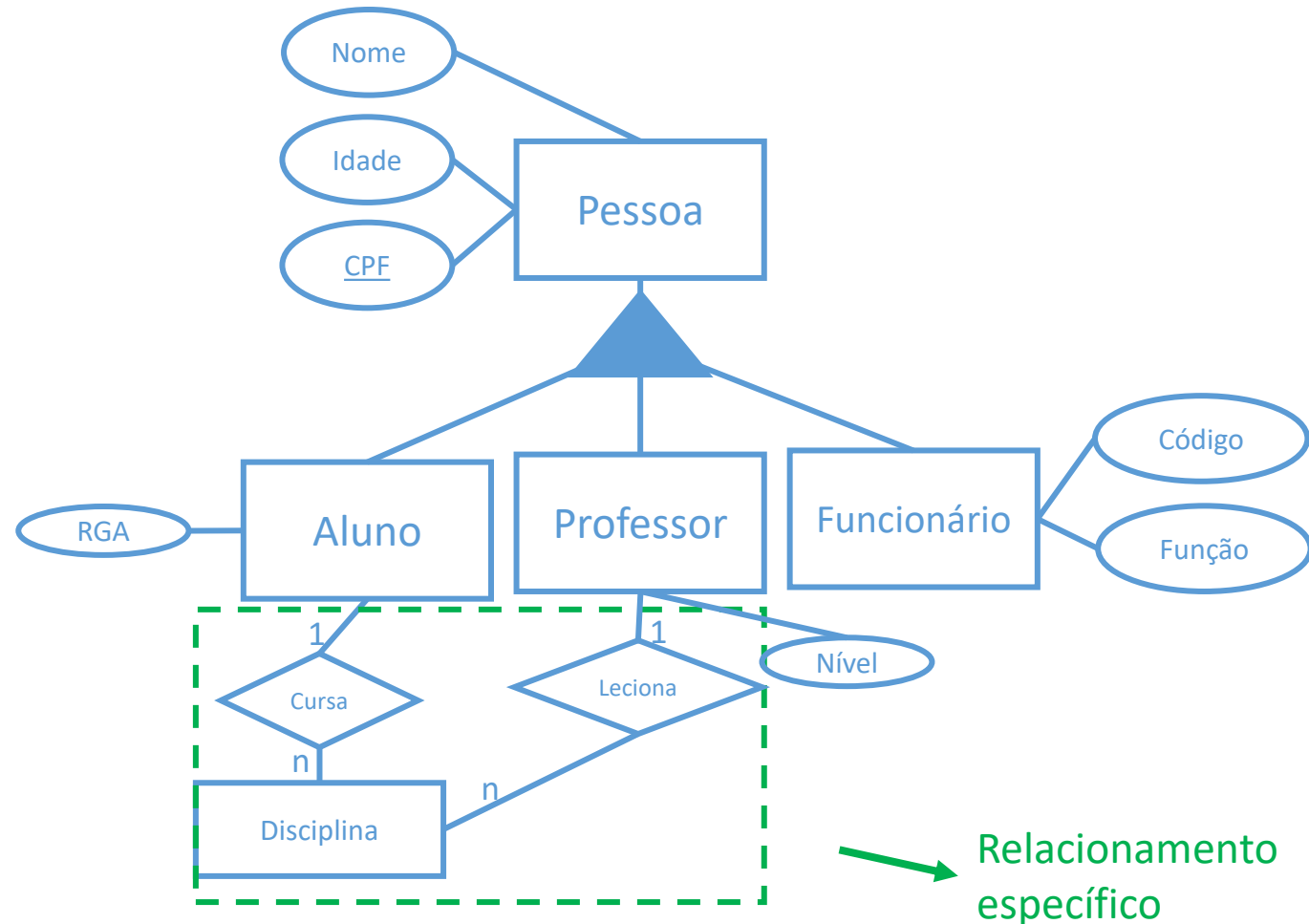
- É o processo de definir um **conjunto de subclasses** de um tipo entidade
- Passos:
  1. define-se um conjunto de **subclasses** de um tipo-entidade
  2. associa-se **atributos** adicionais **específicos** às **subclasses**
  3. estabelece-se **tipos-relacionamento** adicionais **específicos** às subclasses, caso necessário



# Generalização / especialização

- **Especialização**

- É o processo de definir um **conjunto de subclasses** de um tipo entidade
- Passos:
  1. define-se um conjunto de **subclasses** de um tipo-entidade
  2. associa-se **atributos** adicionais **específicos** às **subclasses**
  3. estabelece-se **tipos-relacionamento** adicionais **específicos** às subclasses, caso necessário



# Generalização / especialização

## • Generalização

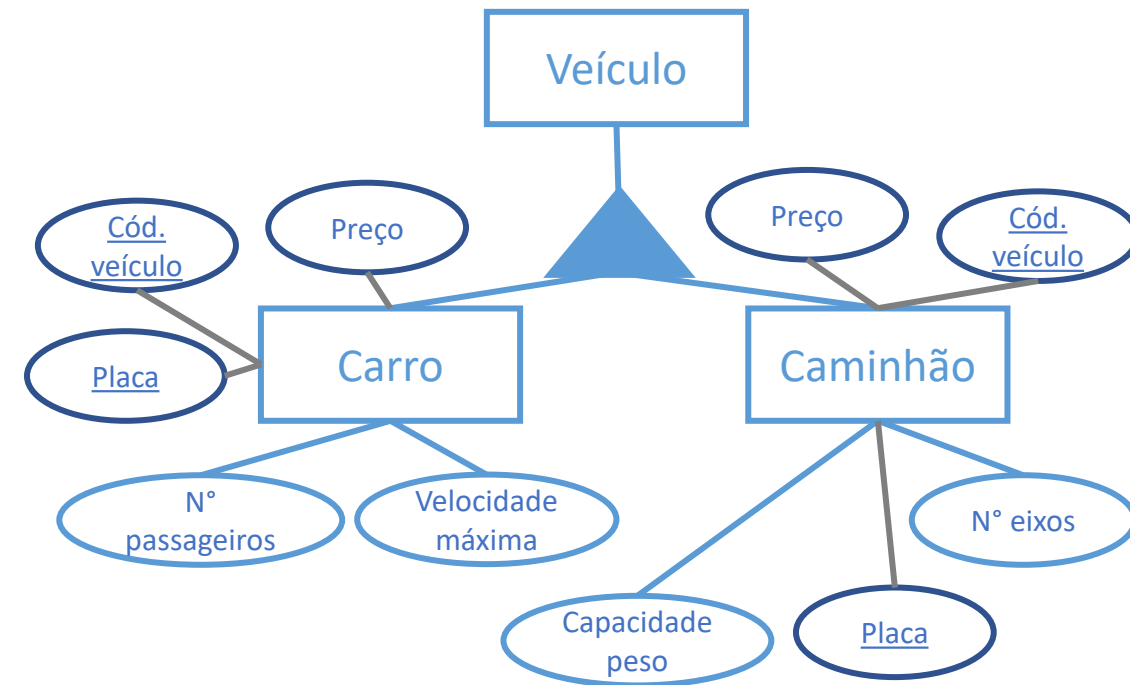
- Resultado da união de dois ou mais tipos-entidade de **nível mais baixo (subclasse)**, produzindo um tipo-entidade de **nível mais alto (superclasse)**
- É uma abstração de um conjunto de entidades
- Passos:
  1. Suprime-se as diferenças entre os tipos-entidades
  2. Identifica-se os atributos em comum
  3. Generaliza-se em uma superclasse



# Generalização / especialização

- **Generalização**

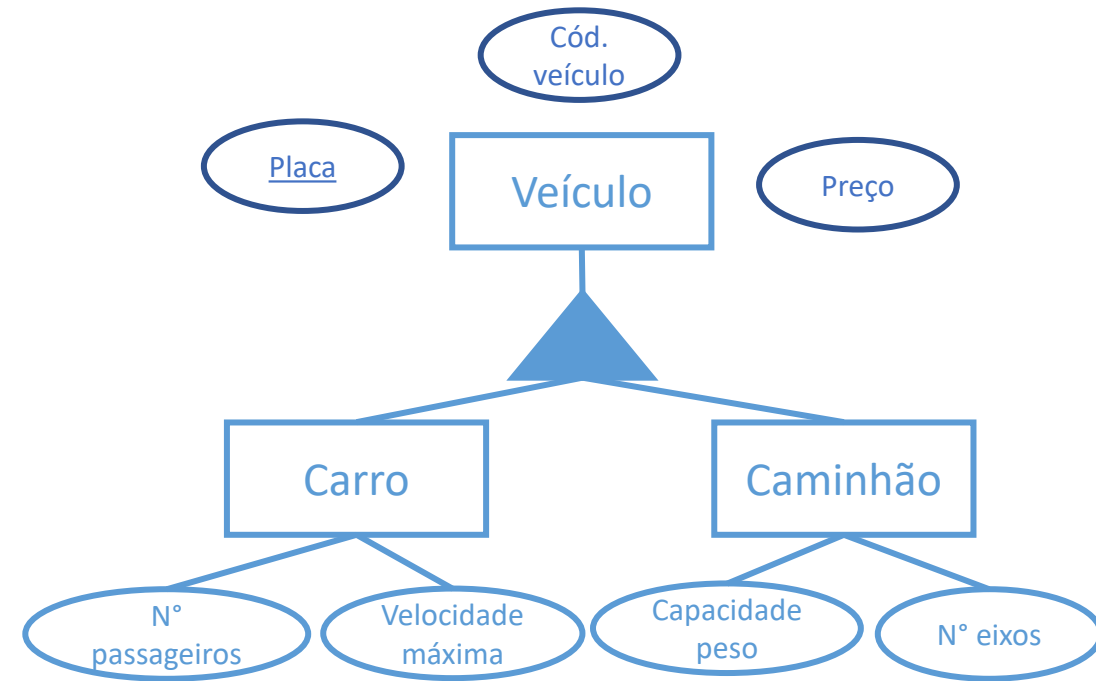
- Resultado da união de dois ou mais tipos-entidade de **nível mais baixo (subclasse)**, produzindo um tipo-entidade de **nível mais alto (superclasse)**
- É uma abstração de um conjunto de entidades
- Passos:
  1. **Suprime-se as diferenças entre os tipos-entidades**
  2. Identifica-se os atributos em comum
  3. Generaliza-se em uma superclasse



# Generalização / especialização

- **Generalização**

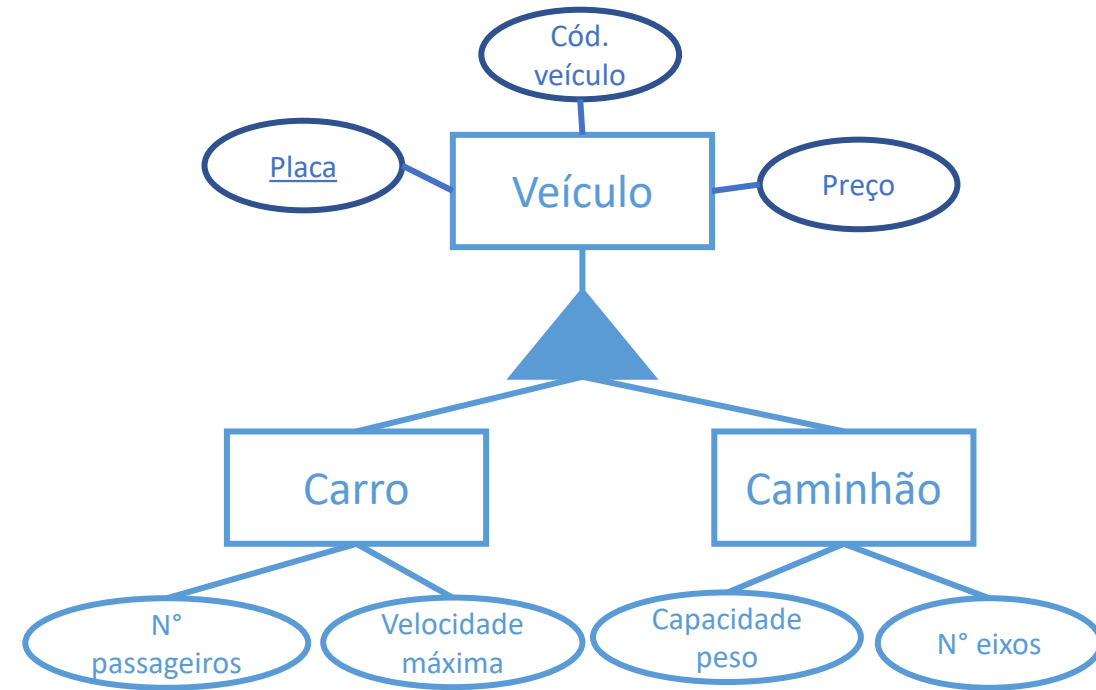
- Resultado da união de dois ou mais tipos-entidade de **nível mais baixo (subclasse)**, produzindo um tipo-entidade de **nível mais alto (superclasse)**
- É uma abstração de um conjunto de entidades
- Passos:
  1. Suprime-se as diferenças entre os tipos-entidades
  2. **Identifica-se os atributos em comum**
  3. Generaliza-se em uma superclasse



# Generalização / especialização

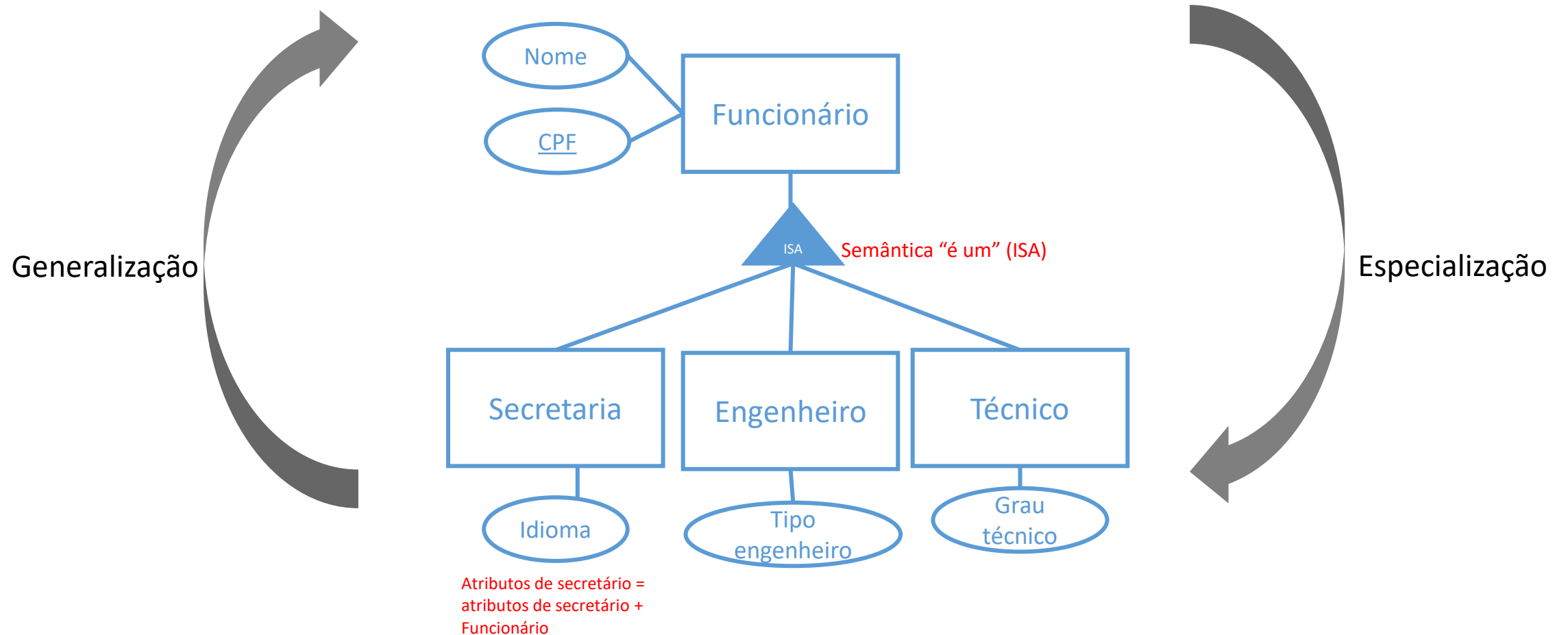
- **Generalização**

- 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:
  1. Suprime-se as diferenças entre os tipos-entidades
  2. Identifica-se os atributos em comum
  3. **Generaliza-se em uma superclasse**



# Generalização / especialização

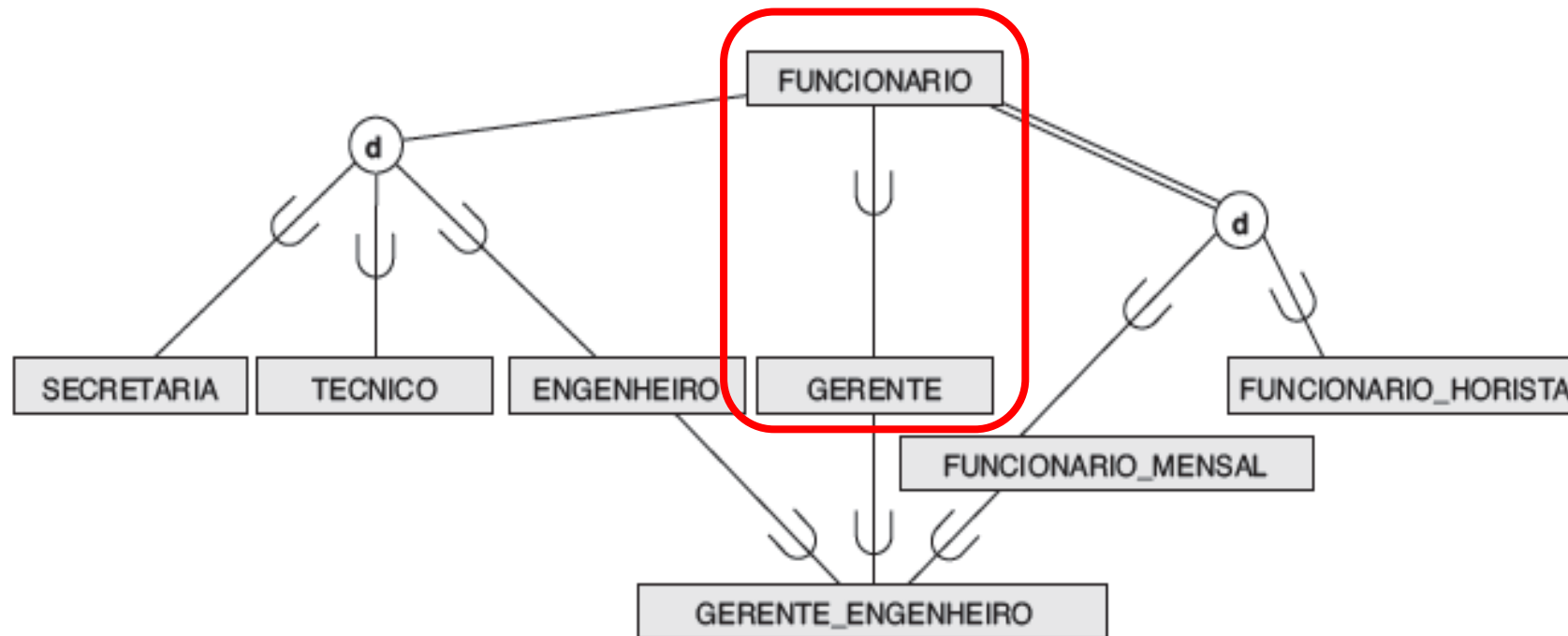
## Representação



# Restrições – única subclasse

- **Única subclasse:**

- Uma especialização pode constituir em uma única subclasse apenas.
  - Nesse caso não utilizamos notação de círculo (GERENTE)

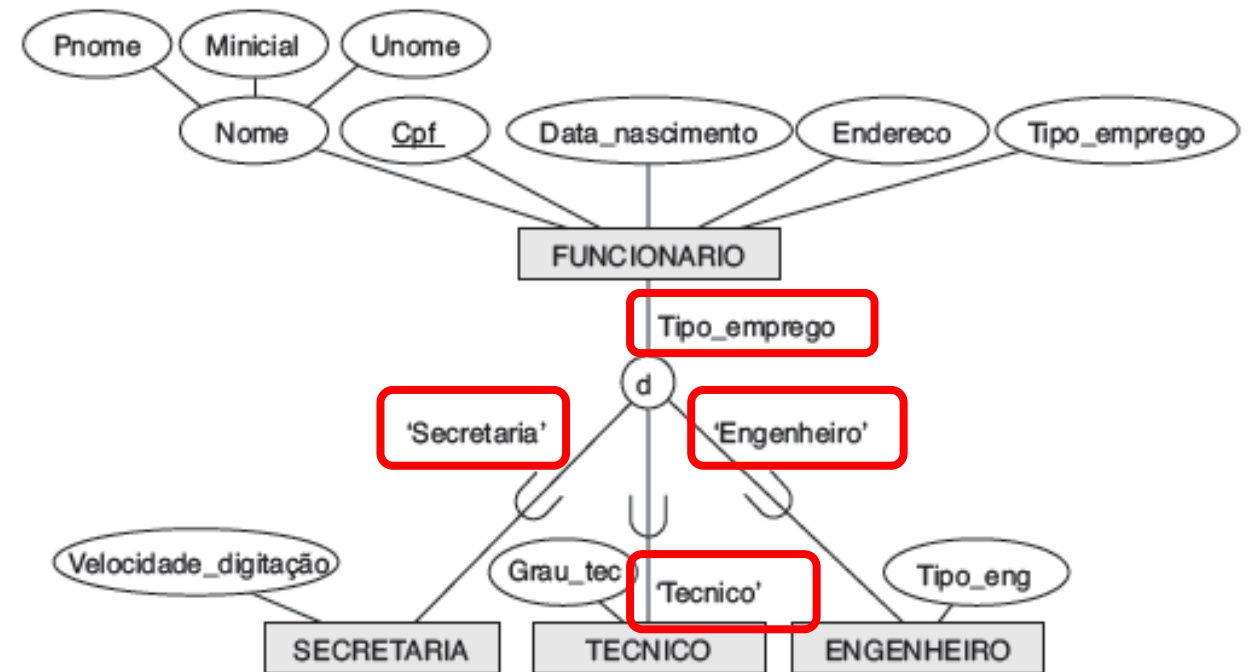




# Restrições - predicado

- Subclasse definida por predicado (ou por condição)
- É possível determinar exatamente as entidades que se tornarão membro de cada subclasse ao **colocar uma condição sobre o valor de algum atributo em uma superclasse.**

Ex: Tipo\_emprego = 'Secretaria'

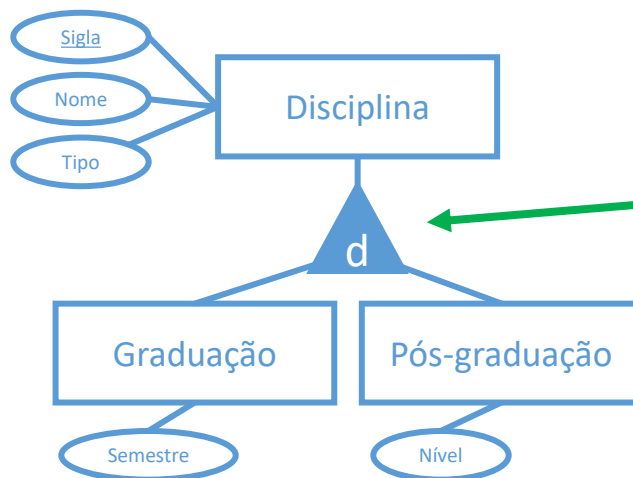


# Restrição de disjunção / sobreposta

- Subclasses mutuamente exclusivas

- Restrição Disjunta (exclusiva):

- Especifica que as subclasses da especialização devem ser disjuntas:
      - Uma entidade pode ser um membro de **no máximo uma subclasse da especialização**
      - Especificada por d no diagrama EER



Um disciplina não pode ser de graduação e pós-graduação ao mesmo tempo

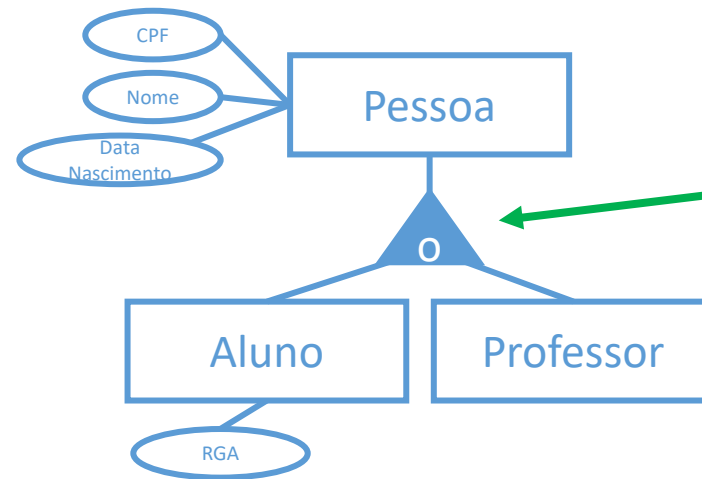


# Restrição de disjunção / sobreposta

- Subclasses que se sobrepõem

- Se não for disjunta/compartilhada é sobreposta (*overlapping*):

- Isto é, a mesma entidade **pode ser membro de mais de uma subclasse da especialização**
    - Especificado por o no diagrama EER



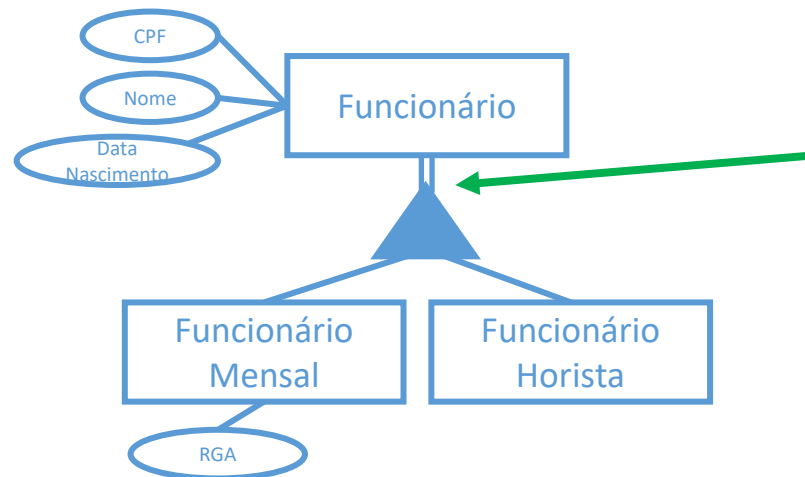
Uma pessoa pode ser aluno e professor ao mesmo tempo



# Restrição de completude

- **Total**

- Especifica que **toda entidade em uma superclasse** deve ser membro de **pelo menos** uma subclasse na especialização
  - Representada no diagrama EER por uma **linha dupla**



Todo FUNCIONÁRIO precisa ser **necessariamente** FUNCIONÁRIO\_HORISTA ou FUNCIONÁRIO\_MENSAL

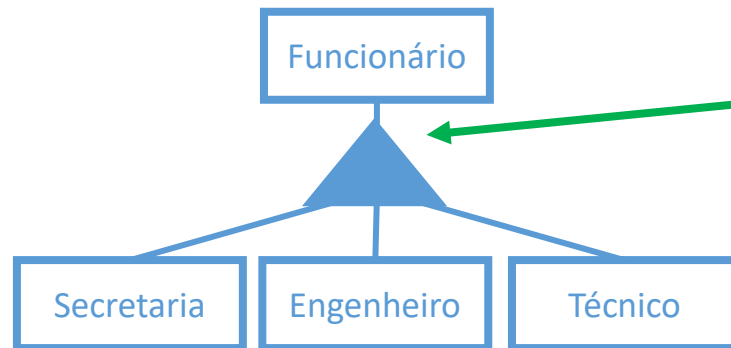
Nesse caso a especialização {FUNCIONARIO\_HORISTA, FUNCIONARIO\_MENSA} é uma **especialização TOTAL** de FUNCIONÁRIO



# Restrição de completude

- **Parcial**

- Permite uma entidade **não pertencer a qualquer uma das subclasse**
  - Representada no diagrama EER por uma **linha simples**



Se a superclasse **FUNCIONÁRIO** não pertencer a nenhuma das subclasses {SECRETARIA, ENGENHEIRO, TECNICO}. Então a especialização será parcial de **FUNCIONÁRIO**.

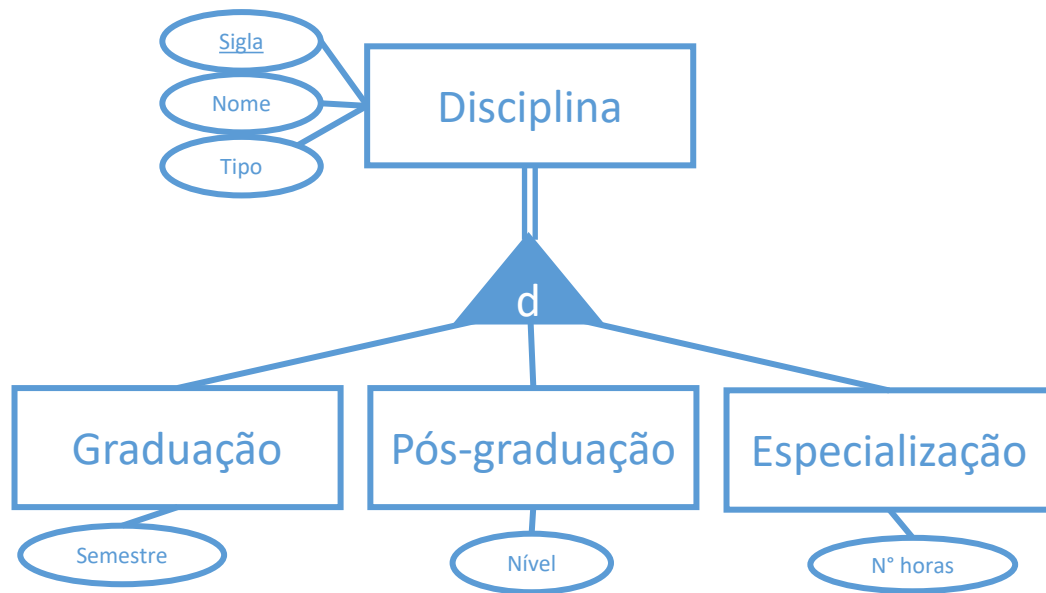


# Restrições sobre especialização e generalização

- Restrições de **disjunção** e **completude** são **independentes**, logo:
  - Disjunta, total
  - Disjunta, parcial
  - Sobreposição, total
  - Sobreposição, parcial



# Restrições – Disjunta Total



## DISJUNÇÃO:

Um disciplina só pode ser membro de no máximo uma subclasse.

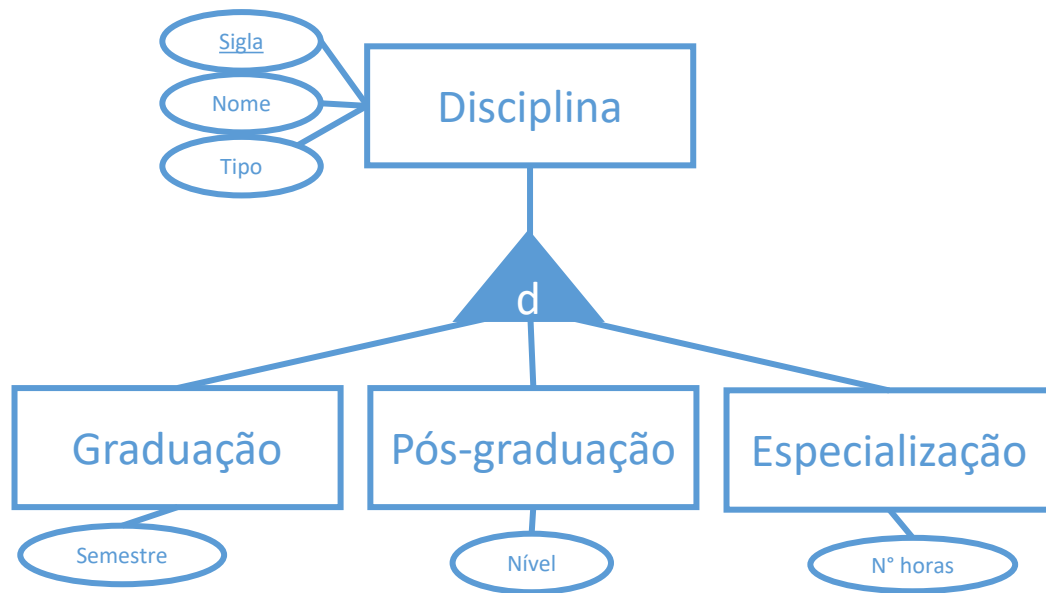
## TOTAL:

Um disciplina obrigatoriamente precisa ser membro de uma subclasse.

Logo, disciplina precisa ser de graduação ou pós-graduação, ou especialização.



# Restrições – Disjunta Parcial



## DISJUNÇÃO:

Um disciplina só pode ser membro de no máximo uma subclasse.

## PARCIAL:

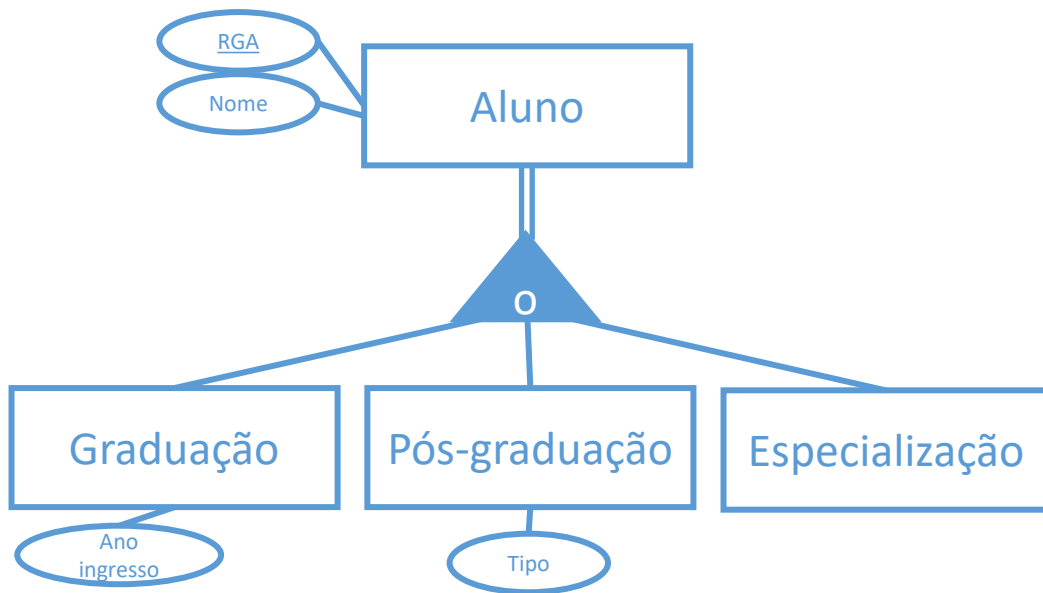
Um disciplina pode não pertencer às suas subclasses

Logo, uma disciplina pode ser ou de graduação, ou de pós-graduação ou de especialização ou não pertencer a nenhum desses subtipos (ser somente uma disciplina).





# Restrições – Sobreposta Total



## **SOBREPOSIÇÃO:**

**Um aluno** pode ser membro de mais de uma subclasse ao mesmo tempo

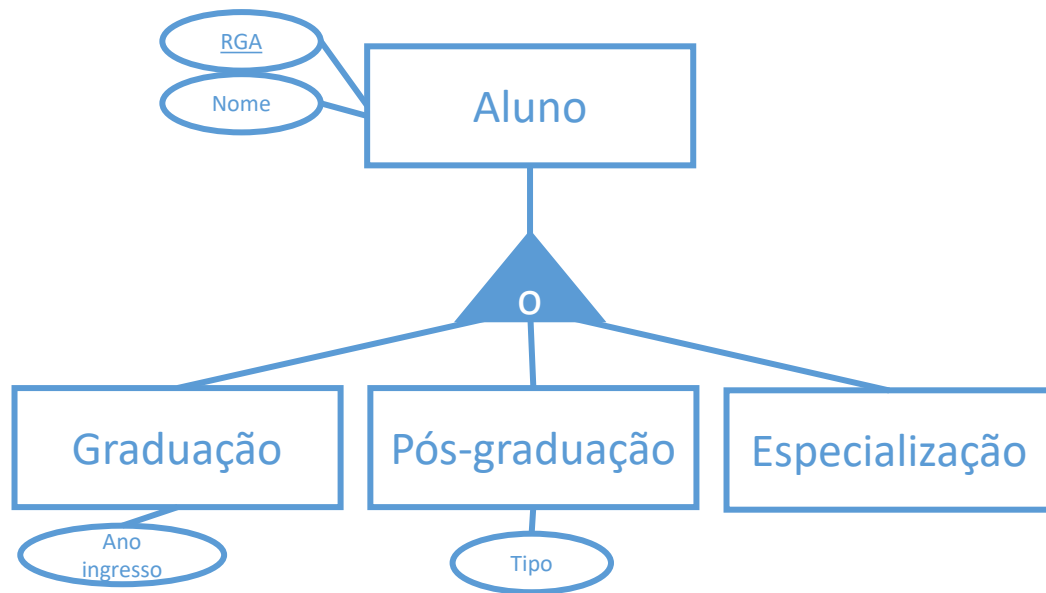
## **TOTAL:**

**Um aluno** obrigatoriamente precisa ser **membro** de uma subclasse.

Logo, um aluno precisa estar matriculado em no mínimo um nível de ensino ou em mais de um ao mesmo tempo.



# Restrições – Sobreposta Parcial



## **SOBREPOSIÇÃO:**

**Um aluno** pode ser membro de mais de uma subclasse ao mesmo tempo

## **PARCIAL:**

**Um aluno** pode não ser membro de uma subclasse.

**Logo, um aluno** pode não estar matriculado nos níveis de ensino especializados ou ele pode estar em mais de um nível ao mesmo tempo.



# Resumo

