

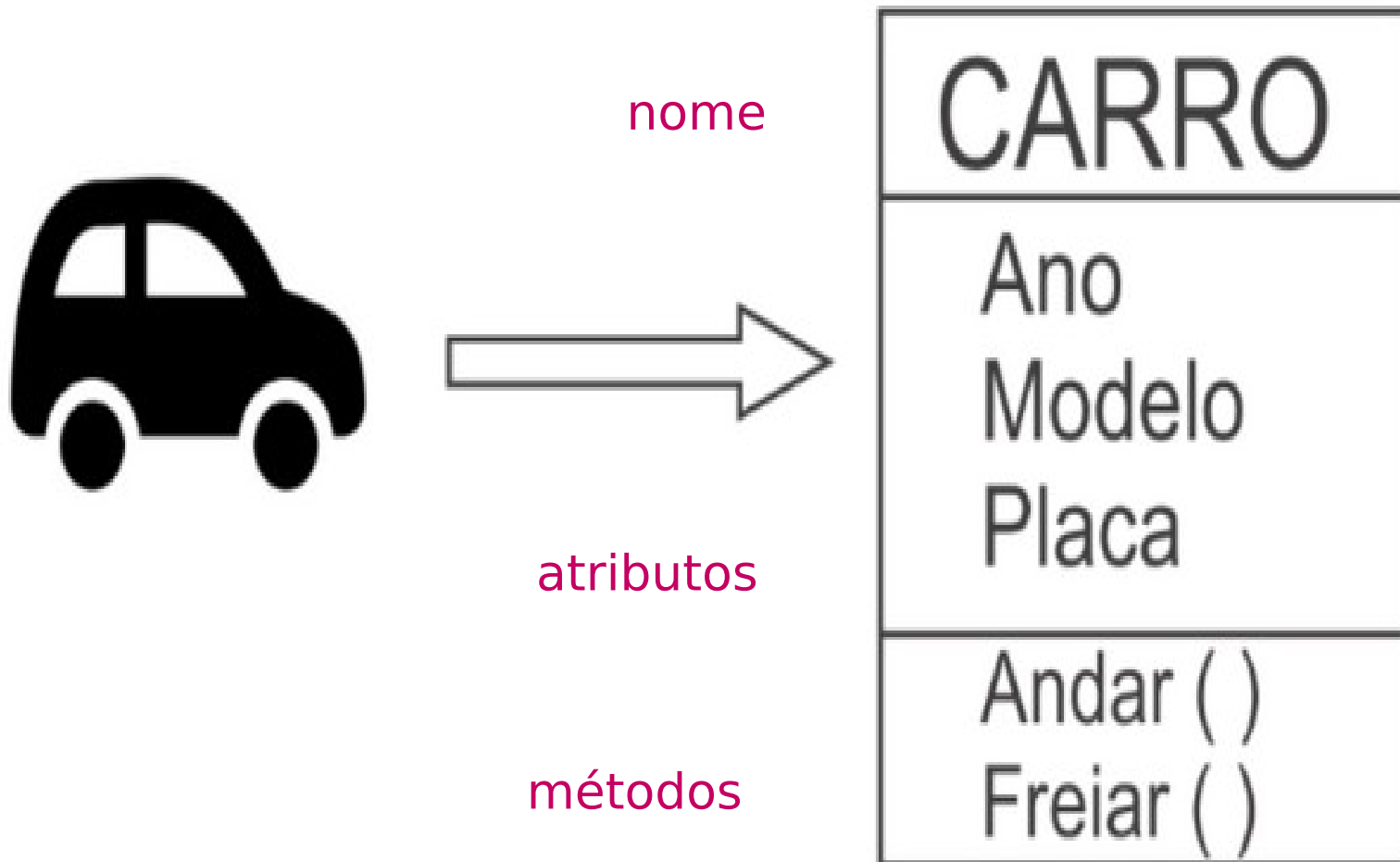
Modelagem de dados através do modelo entidade-relacionamento usando UML

Modelagem de Dados

Eduardo Furlan Miranda

2025-09-28

Adaptado de: WERLICH, C. Modelagem de Dados.
Londrina: EDE SA, 2018. ISBN 978-85-522-1154-9.



Classe

- SGBDs podem usar o conceito de objeto
- Uma classe pode ser vista como um tipo de dado
 - Similar a tipos em linguagens de programação
 - Ou SGBDs
- Classe (“projeto do prédio”)
 - Dados (atributos)
 - Comportamentos (métodos)
- Instância (“um ou mais prédios”)
 - O objeto é criado na memória
- Modelagem de classes
 - Ferramenta *Unified Modeling Language* (UML)

UML

- Padroniza projetos de sistemas
- Abrange aspectos conceituais
 - Regras de negócios
 - Artefatos concretos como as classes
 - Esquemas de banco de dados
 - Componentes de software reutilizáveis
- Diagramas UML
 - Diagrama de classes
 - É o mais usado pois pode representar um conjunto de classes e seus relacionamentos
 - Diagrama de objetos
 - Demonstra como na realidade as informações do objeto podem ficar armazenadas na classe

(continua)

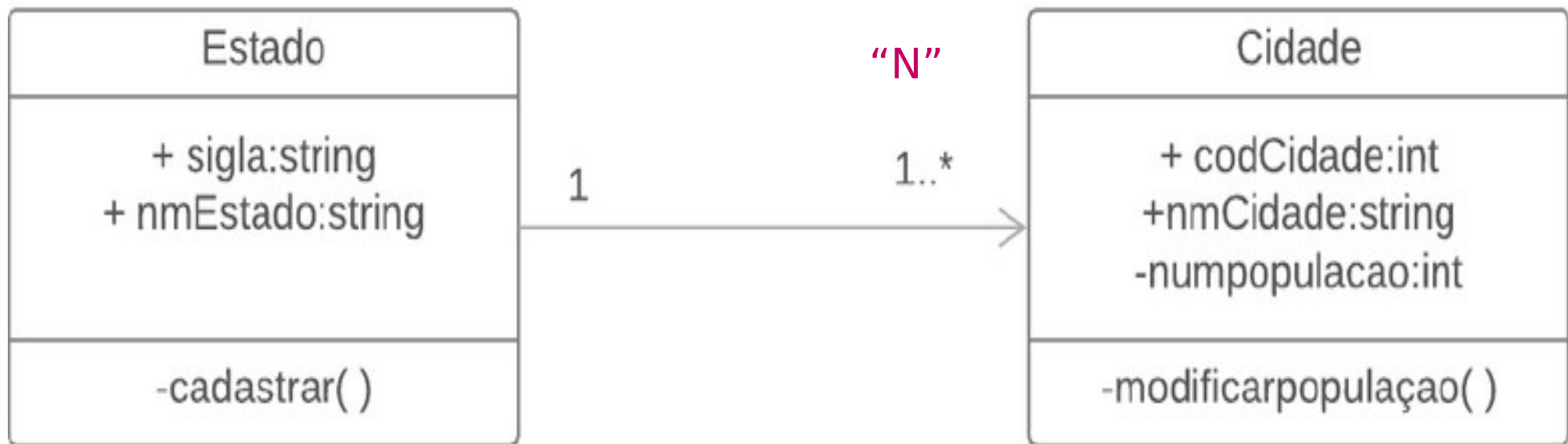
UML

- Diagrama de caso de uso
 - Complemento do diagrama de classes, utilizado principalmente na fase de especificação dos requisitos do sistema, pois demonstra os usuários e as funcionalidades do software
- Diagrama de sequência
 - Demonstra uma visão ou perspectiva, norteadas por tempo, da colaboração entre os objetos, principalmente com a ordem temporal em que as mensagens são trocadas
- Diagrama de atividades
 - Determina o fluxo de tarefas que podem ser executadas pelo software ou por um ator

- Diagrama de estados
 - Conjunto de estados que um objeto pode ter e os “gatilhos” que possam estimular a mudança ou a transição do objeto de um estado para outro
- Diagrama de componentes
 - Componentes do software e seus relacionamentos, podendo ser:
 - Bibliotecas, arquivos de ajuda e classes que podem ser anexadas ao software
- Cardinalidade utilizada em um diagrama de classe
 - Semelhante ao diagrama de entidade-relacionamento
 - Podemos determinar o número exato de ocorrências
 - P. ex., no mínimo 1 ocorrência e no máximo 5

Diagrama de classes

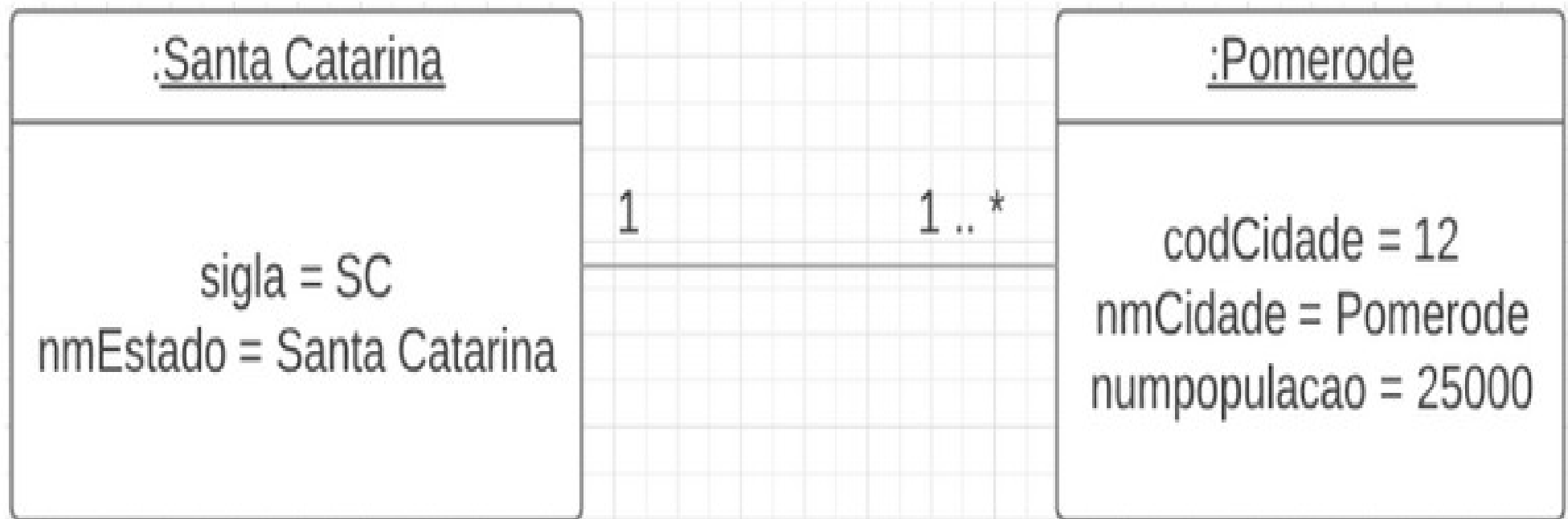
- Pode ser usada para representar o modelo lógico
 - O DRE é usado para o conceitual
- Ex. de relacionamento entre duas classes:



"+" = público; "-" = privado (apenas o método da classe tem acesso)

Diagrama de objetos

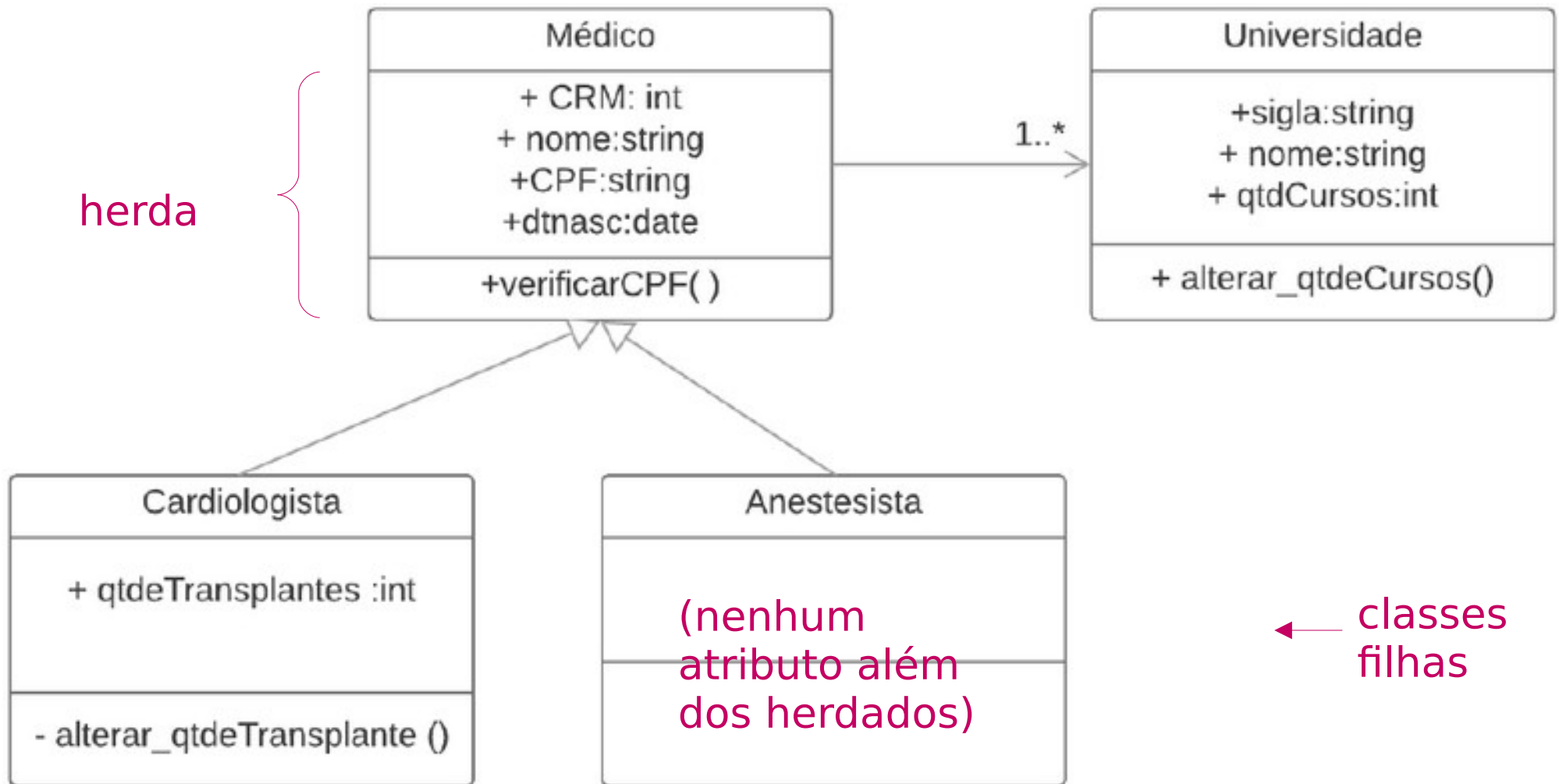
- Informações que serão armazenadas na classe



- UML padroniza modelagem de SGBD e sistemas OO
- Pode ser usada como uma notação em projetos de BD
- Classe de persistência
 - Instância da classe
 - Armazenar ou recuperar informação em um meio persistente
 - Pode ser recuperado posteriormente
- Persistência de objetos
 - Dados e métodos resistem a um processo e podem ser utilizados em outro
- Técnicas de persistência
 - DAO (Data Access Object), Força Bruta, Active Record, Persistência Robusta (Framework) e JDO (Java Data Object)

Herança

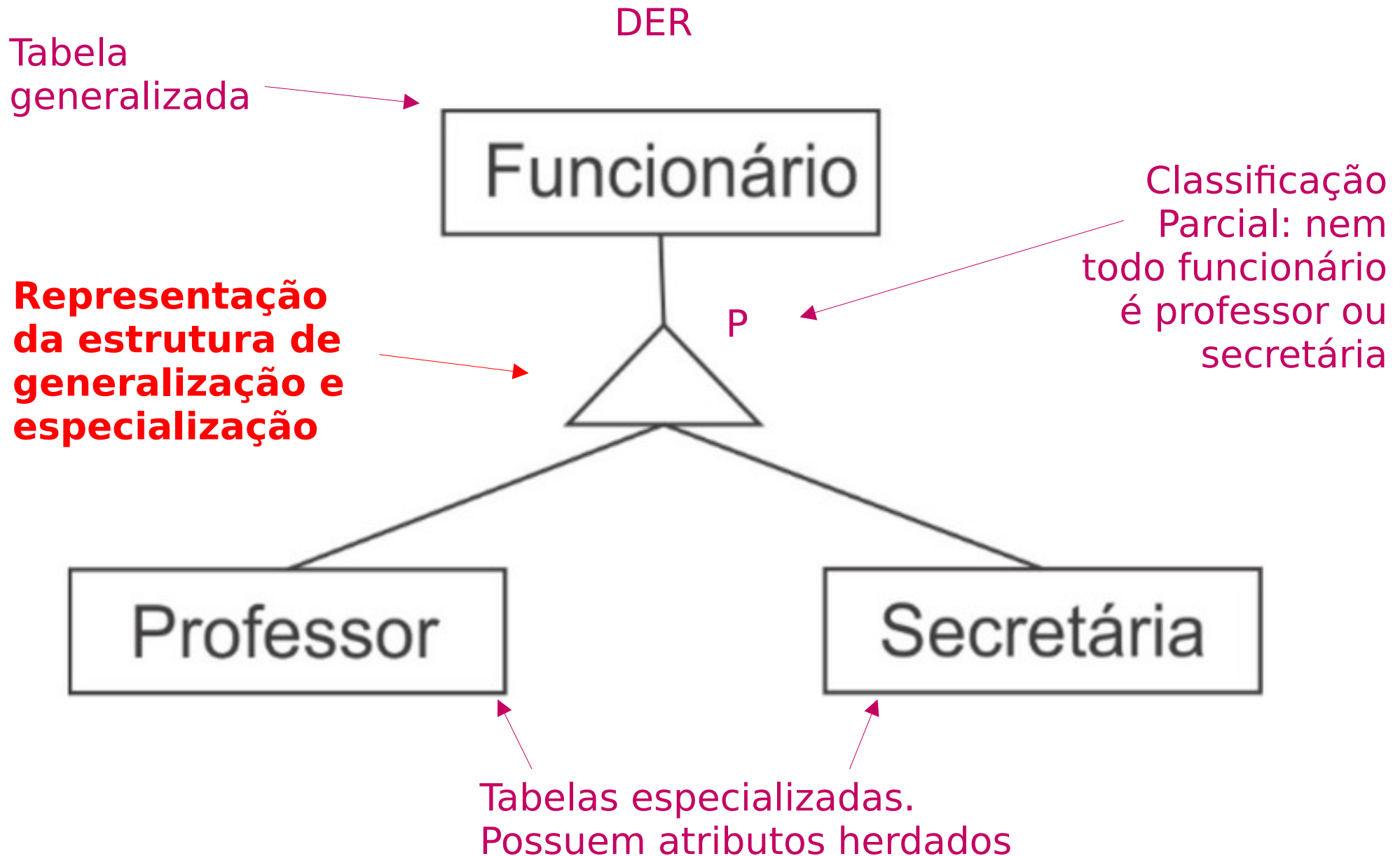
- Uma subclasse herda as características de outra classe (classe mãe ou superclasse)



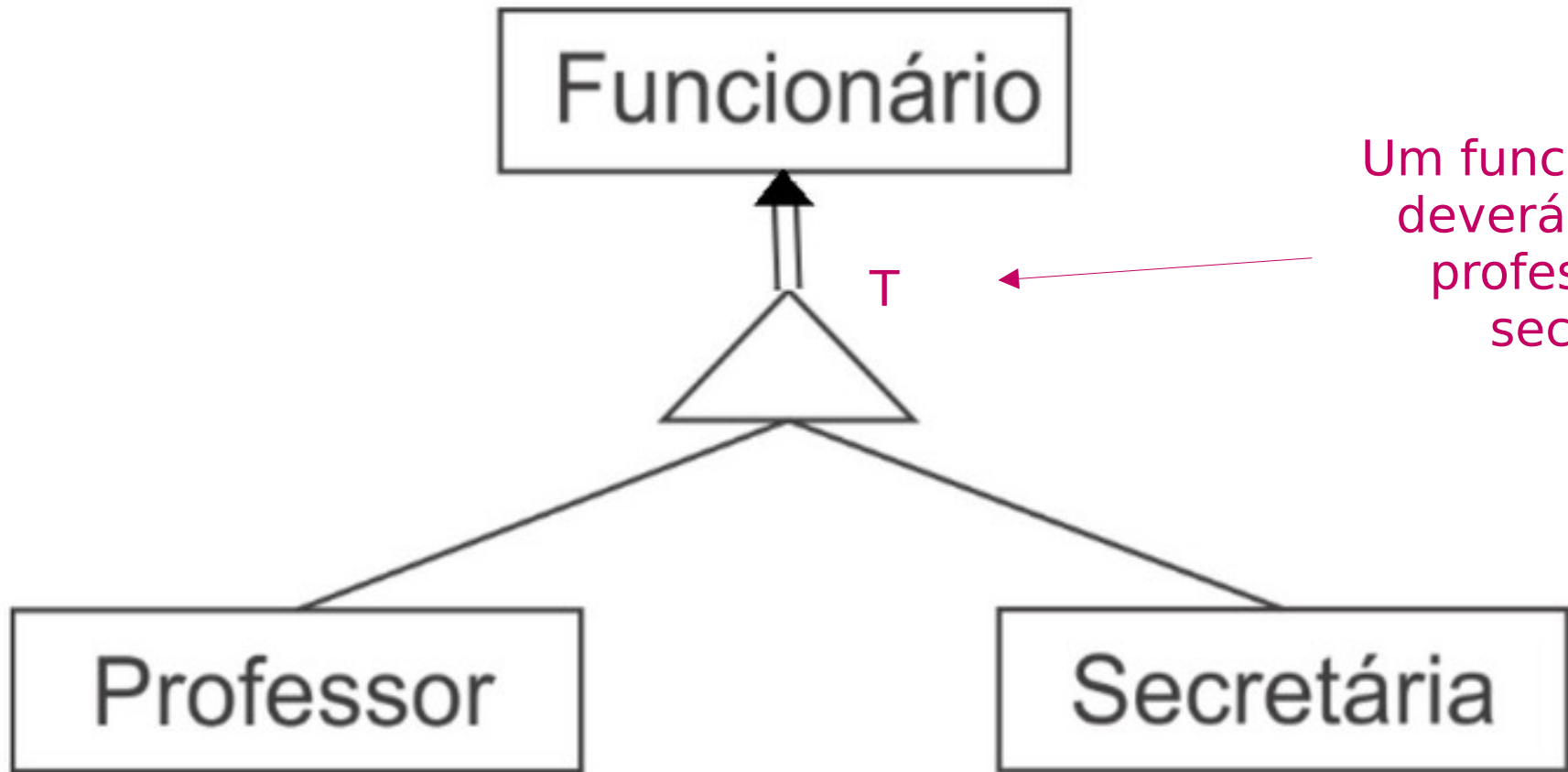
Herança

- Generalização
 - Várias entidades (tabelas) são agrupadas em uma única entidade genérica
- Especialização
 - Processo inverso, em que novas entidades são criadas com atributos que acrescentam detalhes à entidade genérica
- Em DER, a generalização e a especialização determinam que uma entidade contém a outra
 - Uma entidade superior contém um ou mais conjuntos de entidades inferiores

Ex. de generalização e especialização



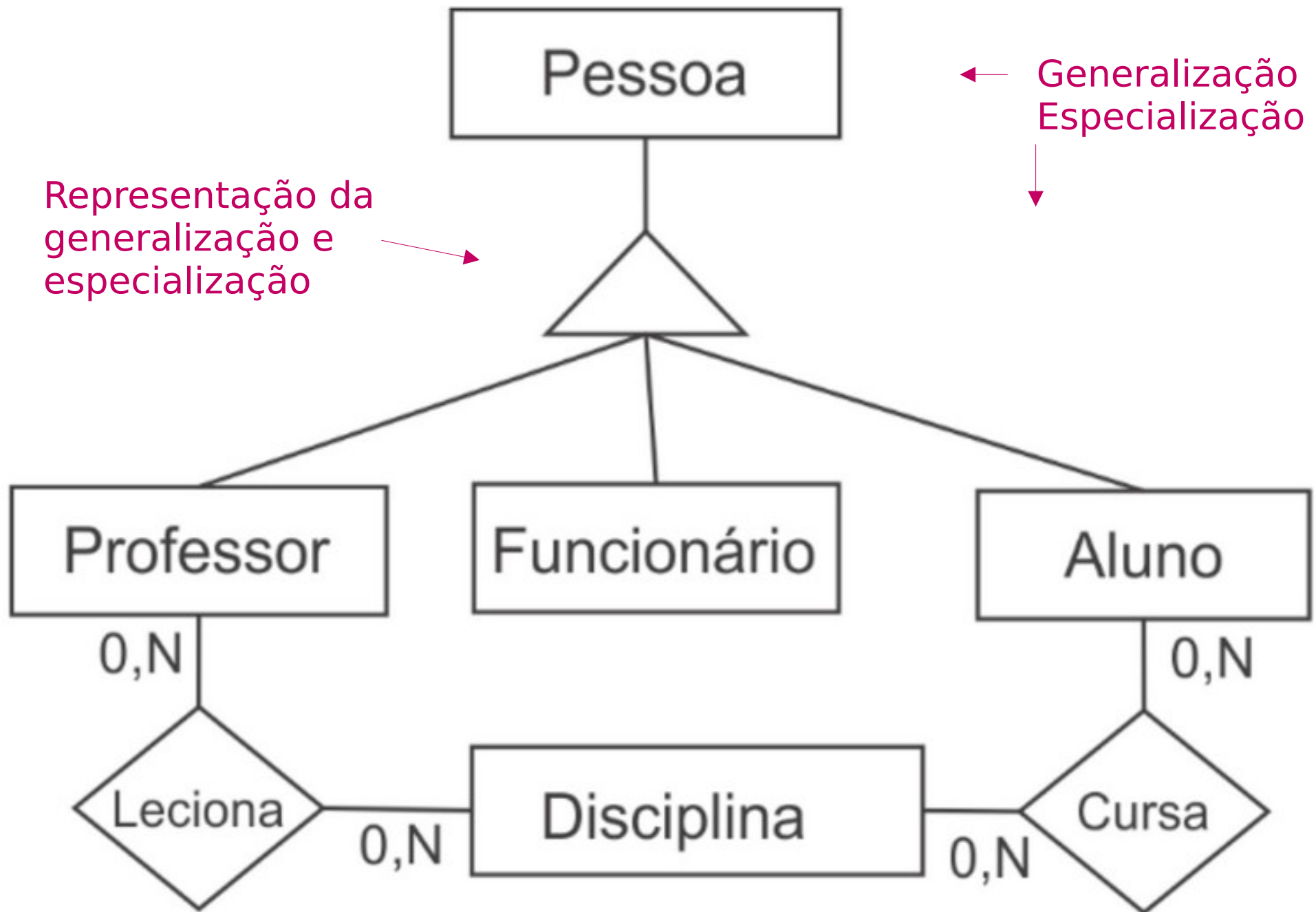
Ex. de generalização e especialização **total**



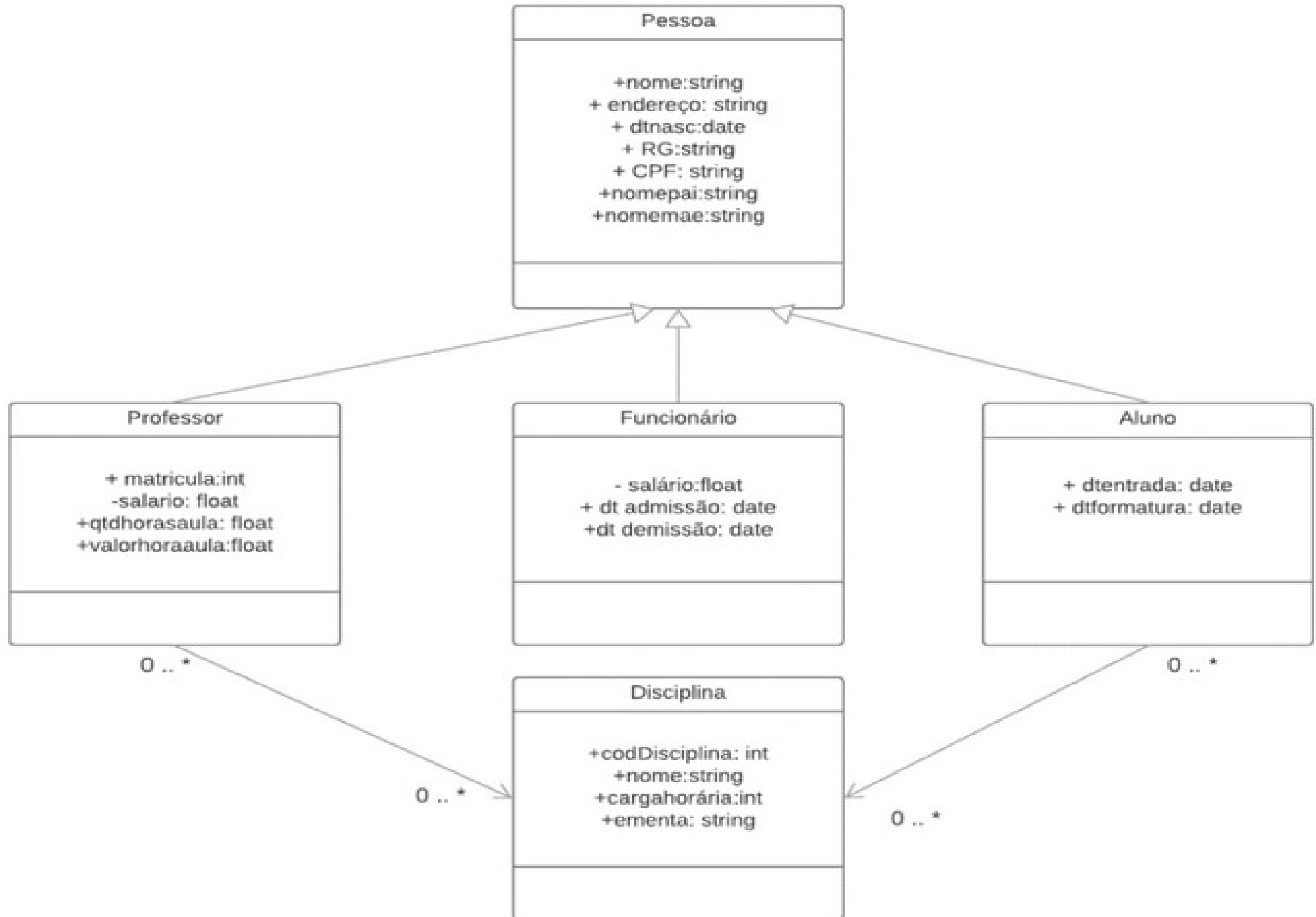
Um funcionário
deverá ser ou
professor ou
secretária

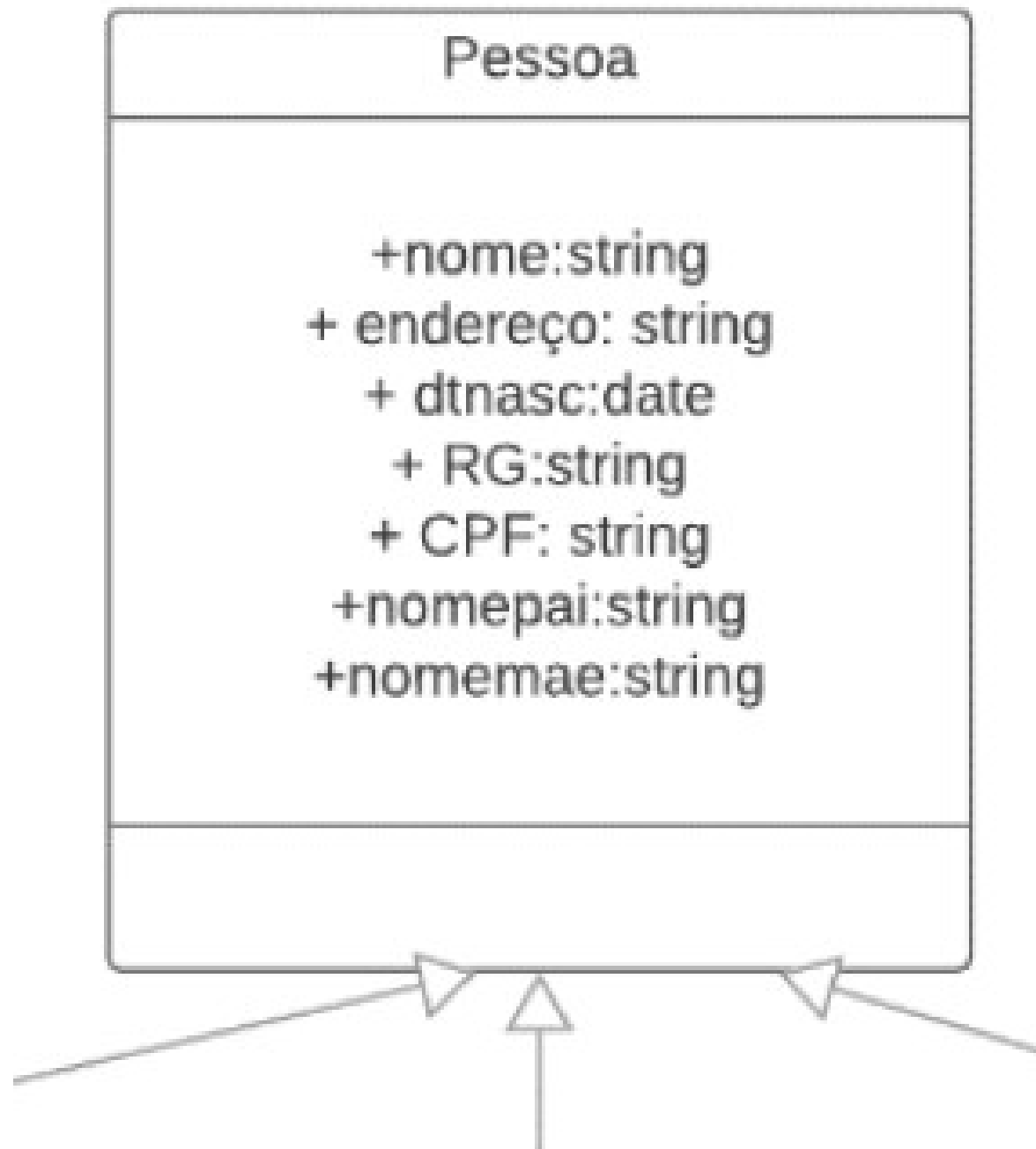
Ex. do modelo conceitual com generalização e especialização

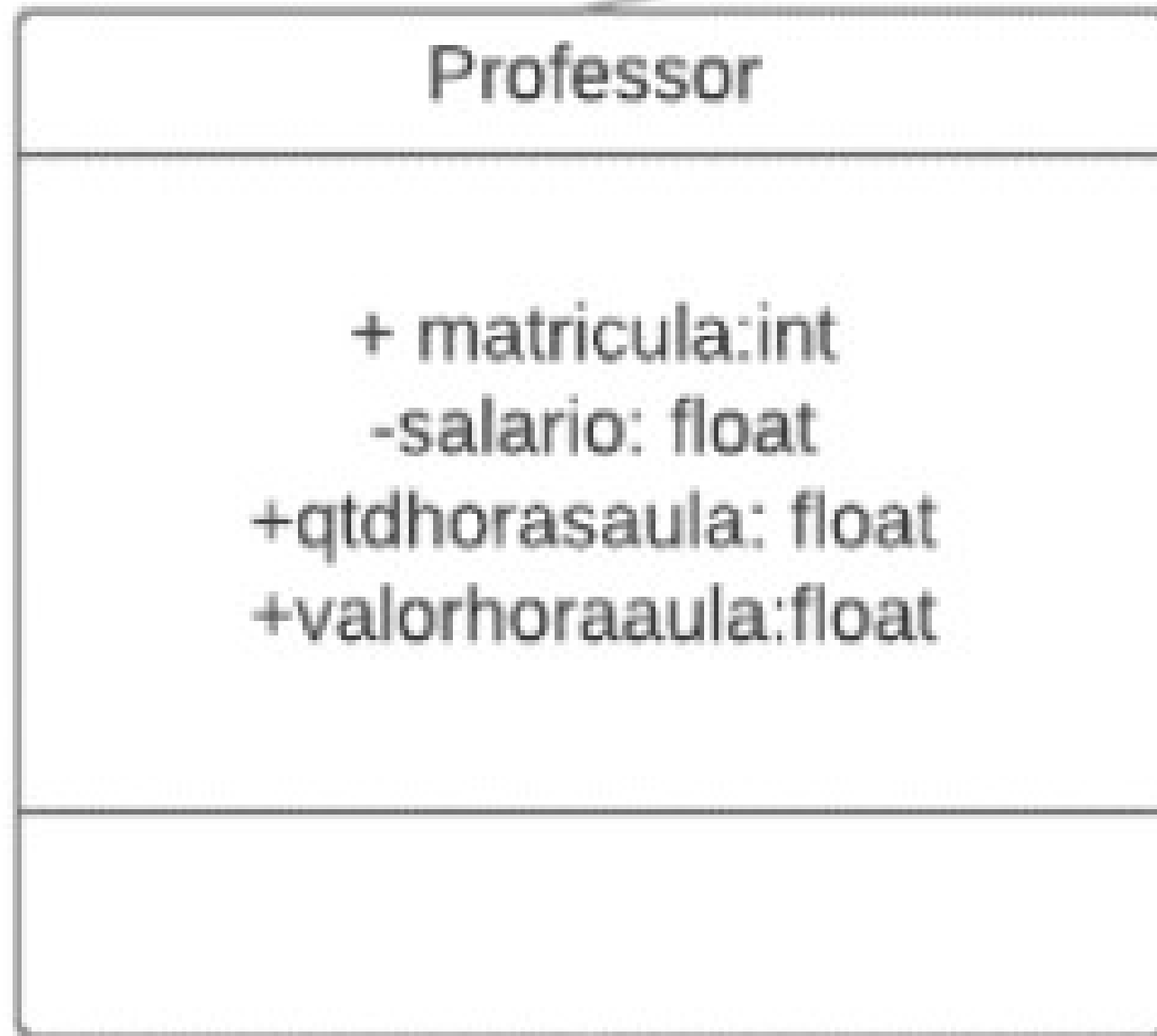
14/20



DER com notação UML







0 .. *

