

# Introdução a Bancos de Dados

UML – Unified Modeling Language

Clodoveu Davis

DCC/UFMG



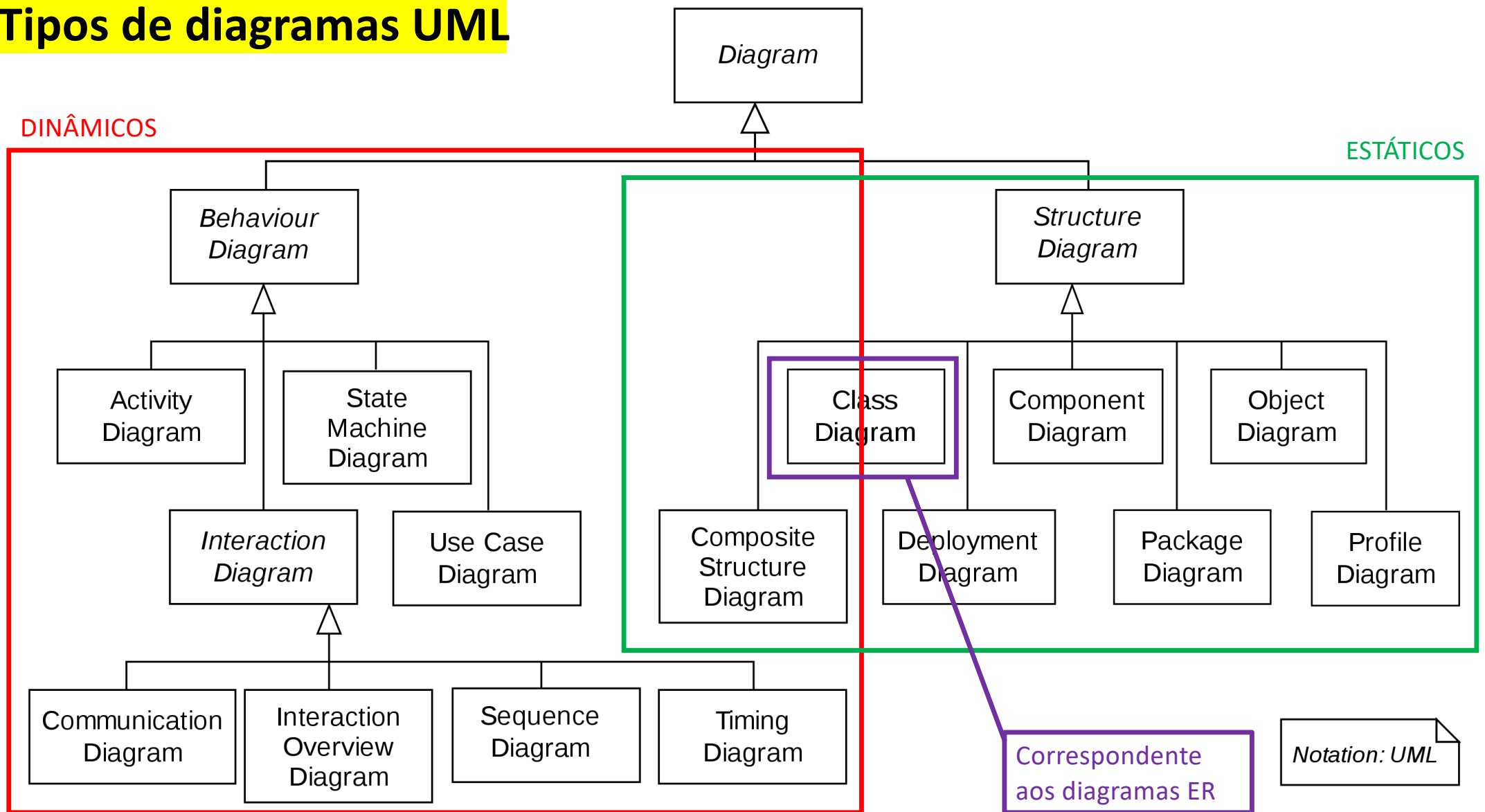
# UML

- A Unified Modeling Language é uma linguagem de modelagem para uso geral, usada em engenharia de software como um recurso padronizado para a especificação e visualização da estrutura de um sistema
- Criada na década de 1990, o final da evolução de diversas alternativas anteriores: OMT e OOSE, principalmente
  - Booch, Jacobson e Rumbaugh na Rational Software
- Versão atual: UML Specification 2.5

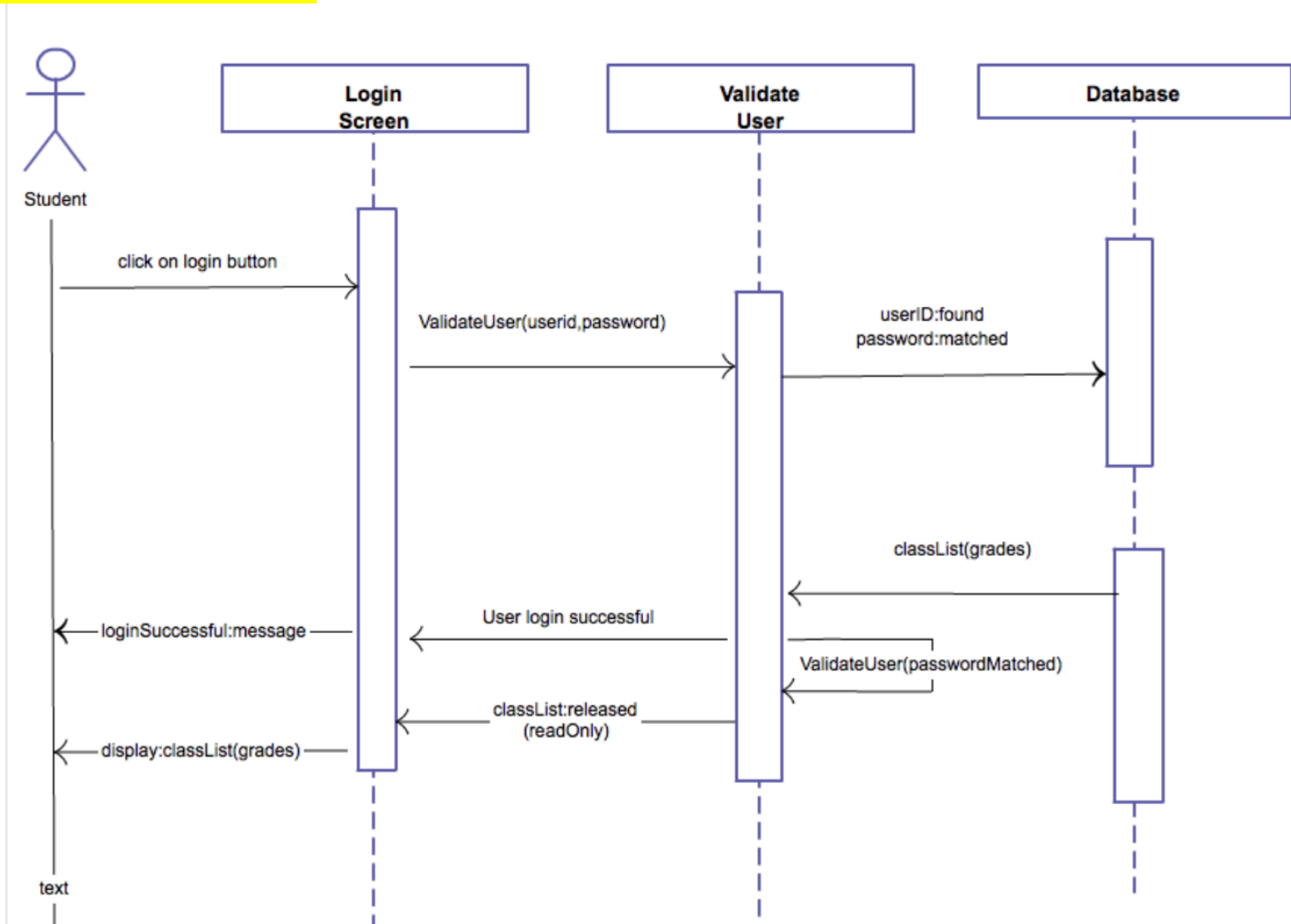
# UML

- A UML foi projetada para ir além dos modelos de dados como o ER, por conter especificações para vários outros tipos de diagramas além do que especifica a estrutura do banco de dados
  - Atividades (**jobs**)
  - **Componentes** do sistema e sua interação com outros componentes – ou seja, entidades e relacionamentos
  - **Execução** (como o sistema irá rodar)
  - Interação entre entidades (**componentes** e **interfaces**)
  - **Interface externa** com o usuário
- Seus diagramas cobrem aspectos **estáticos** e **dinâmicos** dos sistemas

# Tipos de diagramas UML

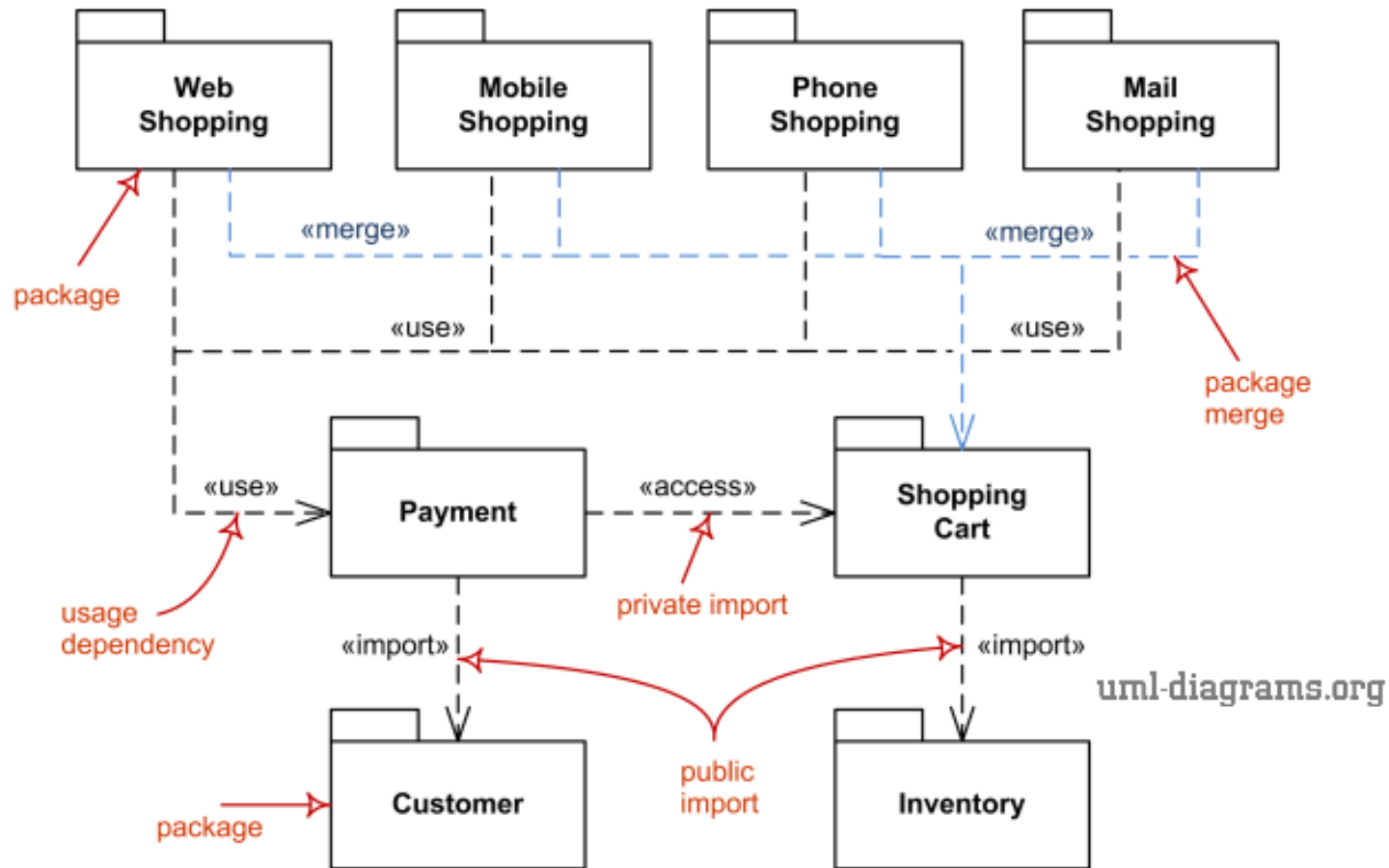


# Diagrama de sequência



<https://i.pinimg.com/originals/57/c4/d7/57c4d70b98744e14aa25f024433a0299.png>

# Diagrama de pacotes



# UML: Diagrama de Classes

- Classe (de objetos)
  - Corresponde aproximadamente à entidade
  - Instância = objeto, que inclui **atributos** e **métodos**, herdados da classe
- Relacionamentos
  - Entre instâncias: dependência, associação, agregação, composição
  - Entre classes: generalização (herança), realização (implementação)

# UML Notation

## Entities & attributes

### Meaning

### Notation

Entity

Name

Attributes

...

...

Attribute (mandatory)

Attribute name [1]

Primary identifier  
attribute

Attribute name {id}

Alternate identifier  
attribute

<<ai>> Attribute name [1]

Multi-valued  
attribute

Attribute name [x..y]

Derived attribute

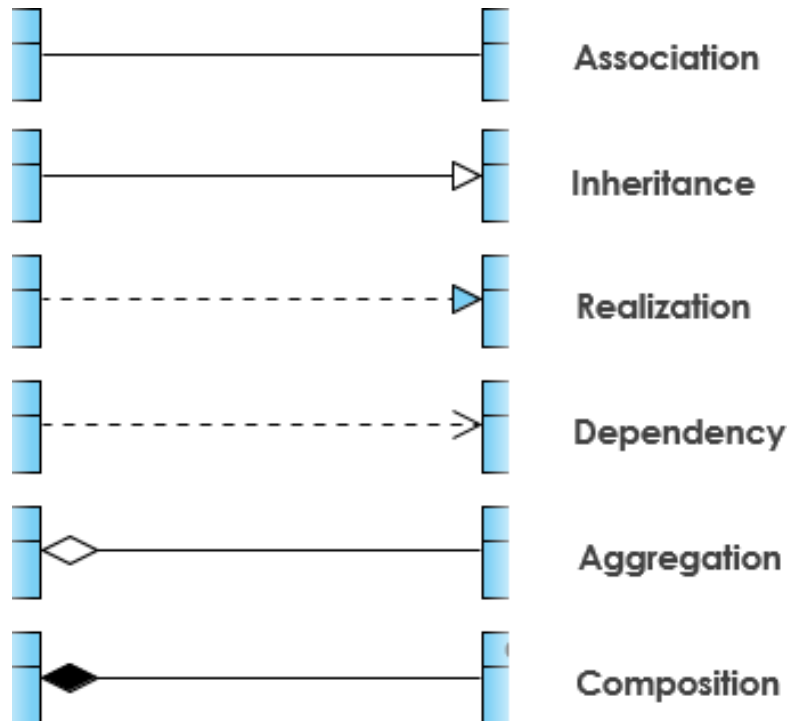
/Attribute name

Optional attribute

Attribute name [0..1]

Métodos,  
restrições,  
funções  
associadas





|             |                               |
|-------------|-------------------------------|
| <b>0</b>    | No instances (rare)           |
| <b>0..1</b> | No instances, or one instance |
| <b>1</b>    | Exactly one instance          |
| <b>1..1</b> | Exactly one instance          |
| <b>0..*</b> | Zero or more instances        |
| <b>*</b>    | Zero or more instances        |
| <b>1..*</b> | One or more instances         |

# UML Notation

## Relationships & participation

### Meaning

### Notation

Relationship

Optional symbol

0

One symbol

1

Many symbol

\*

Zero or one

0..1

Only one

1

Zero or more

0..\*

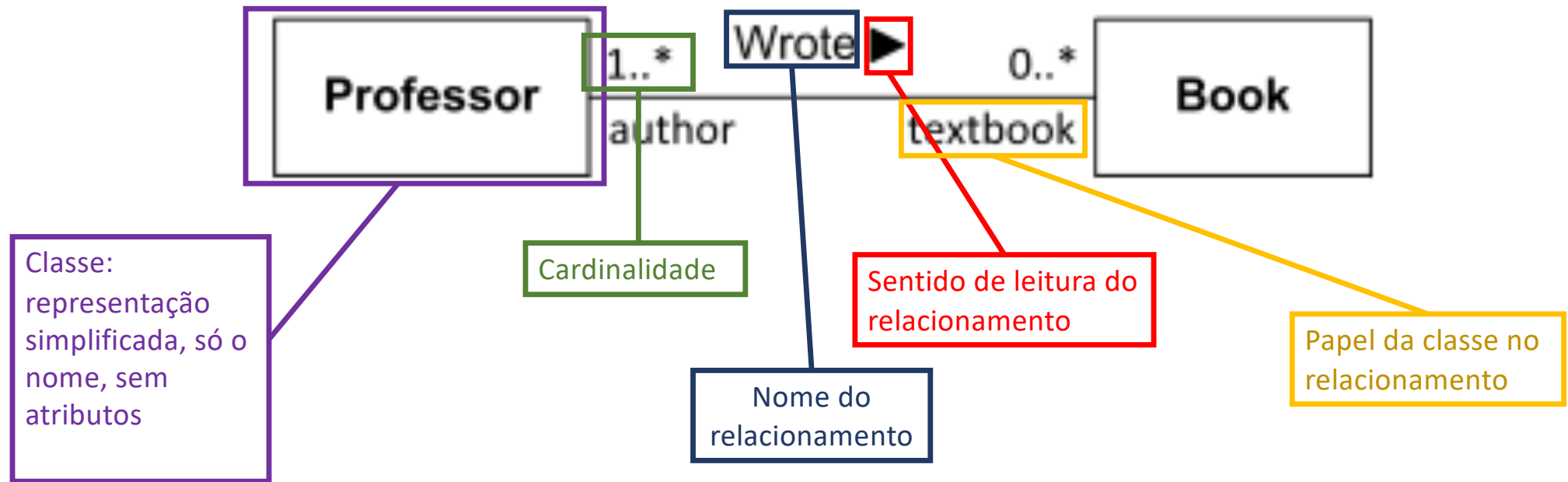
One or more

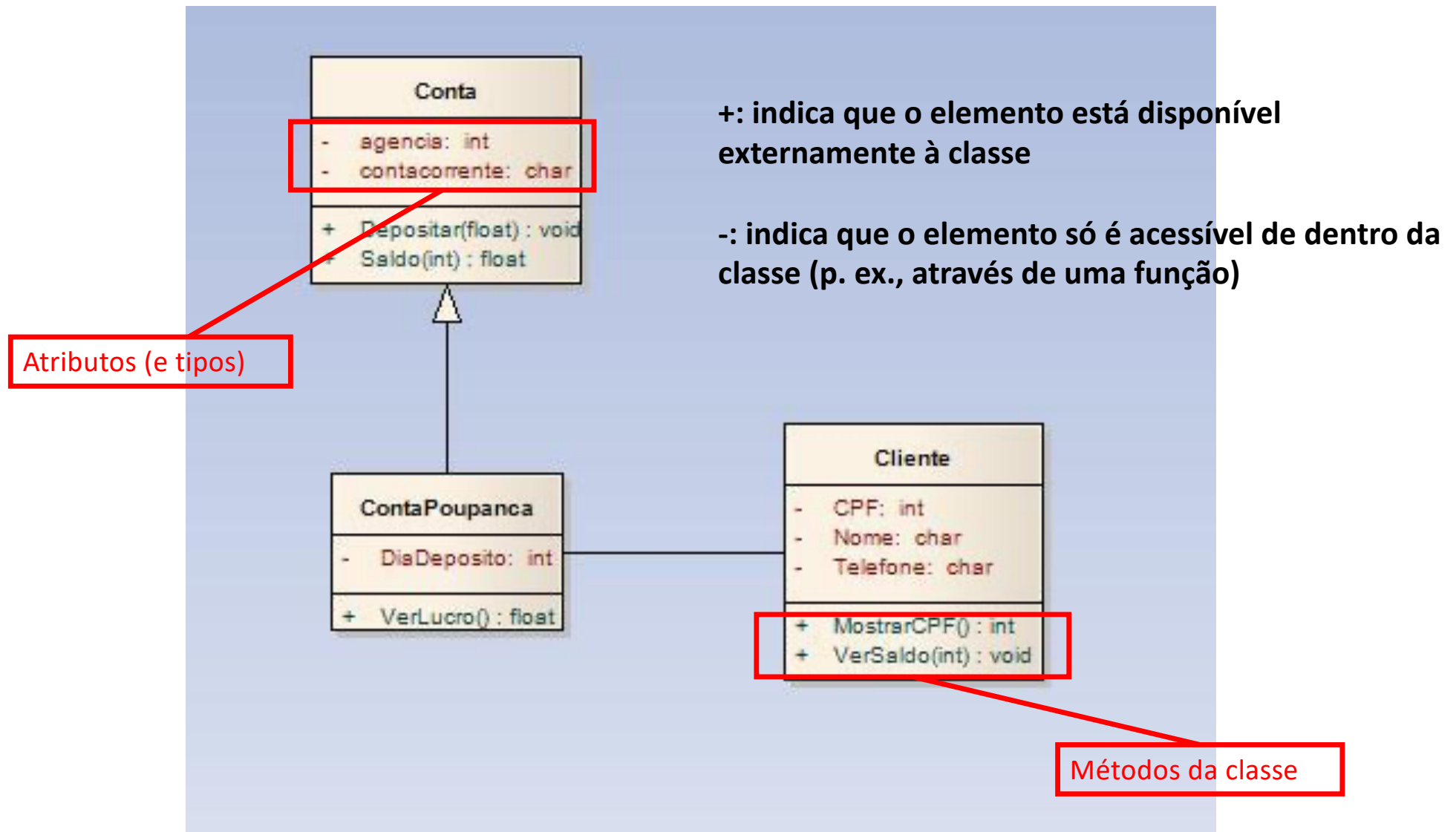
\*

Identifying relationship

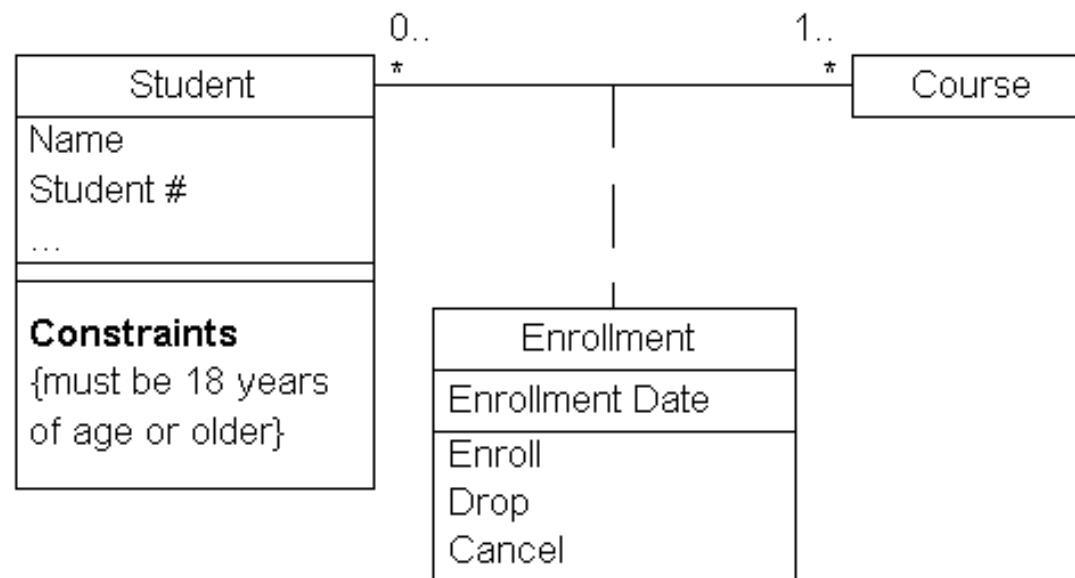
Generalization

Exemplos

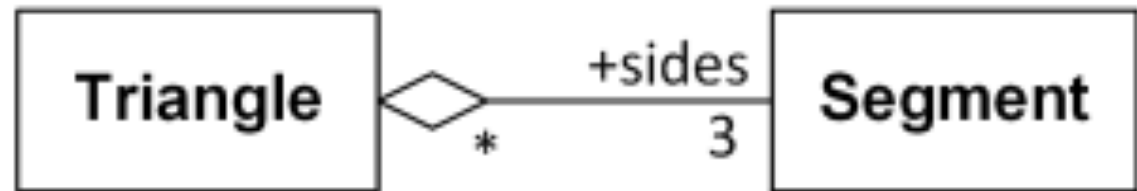




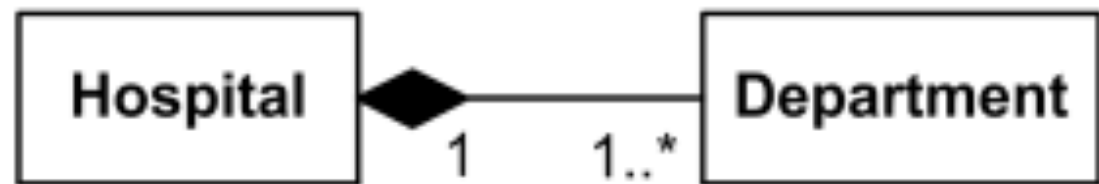
## Atributos de relacionamentos



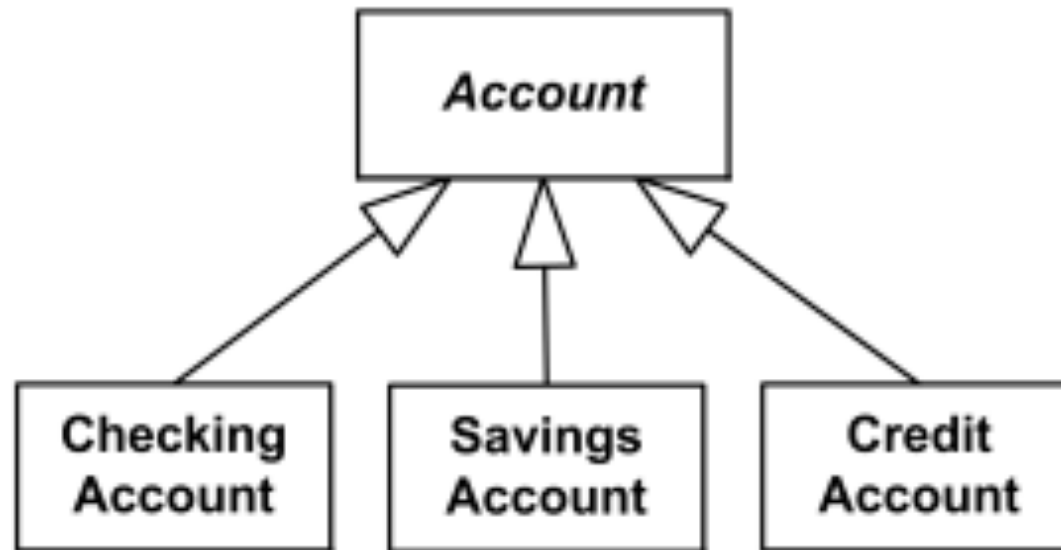
## Agregação



## Composição

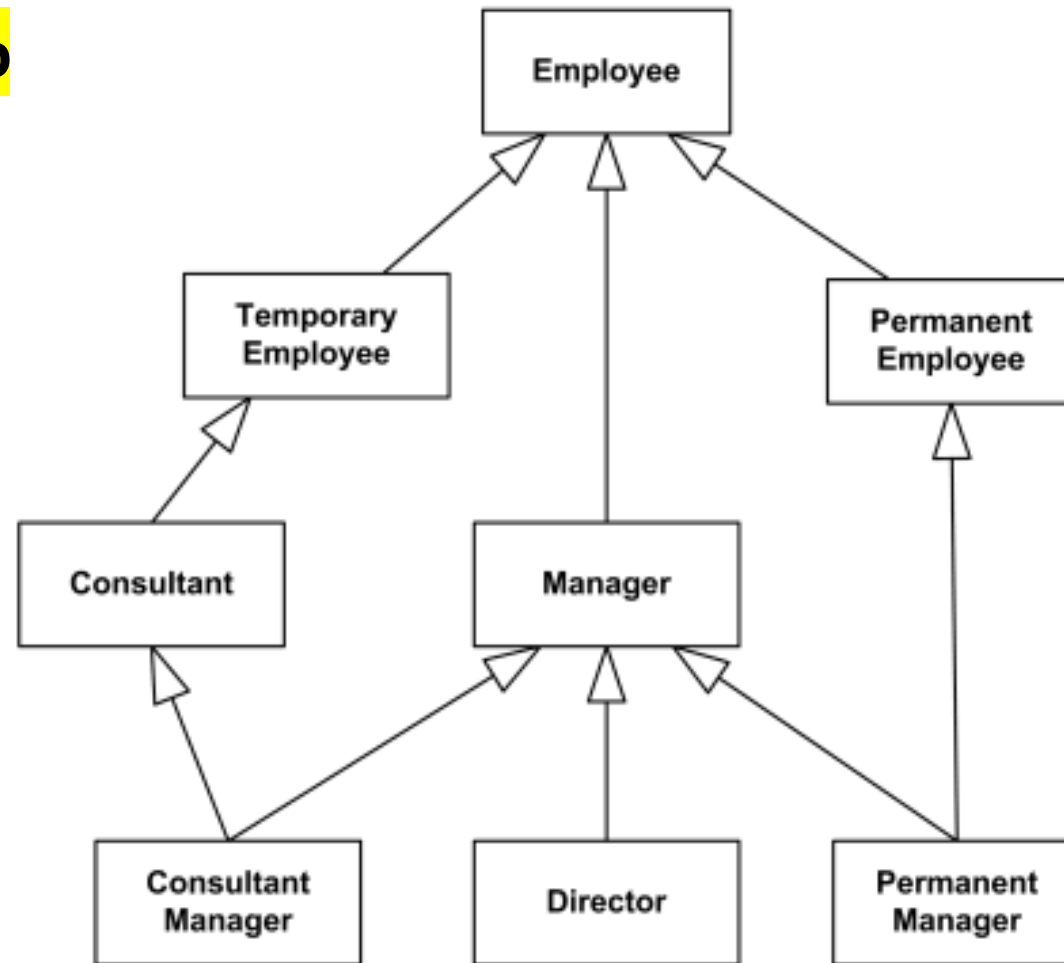


## Generalização



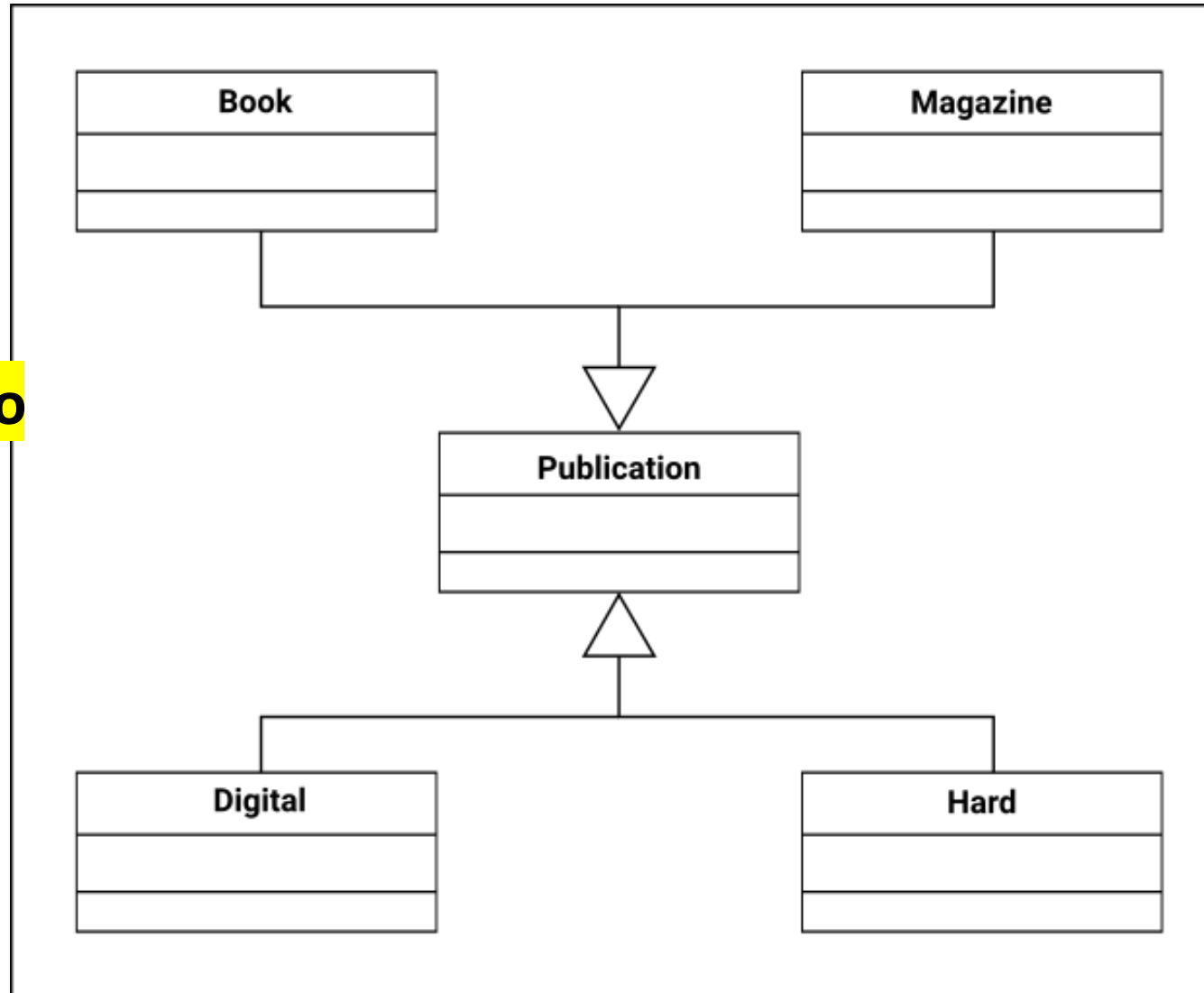


# Generalização



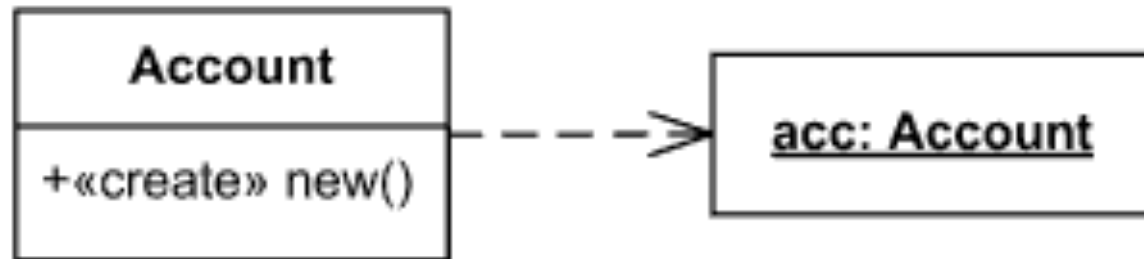
<https://www.uml-diagrams.org/class-diagrams/class-multiple-inheritance.png>

## Generalização



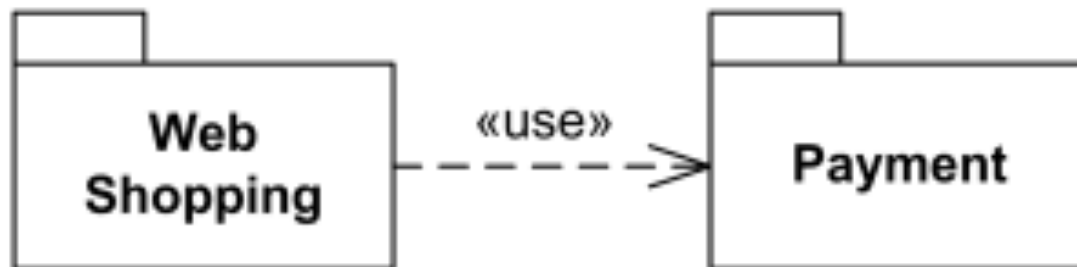
<https://i.stack.imgur.com/Bm8ZJ.png>

## Dependência (create)

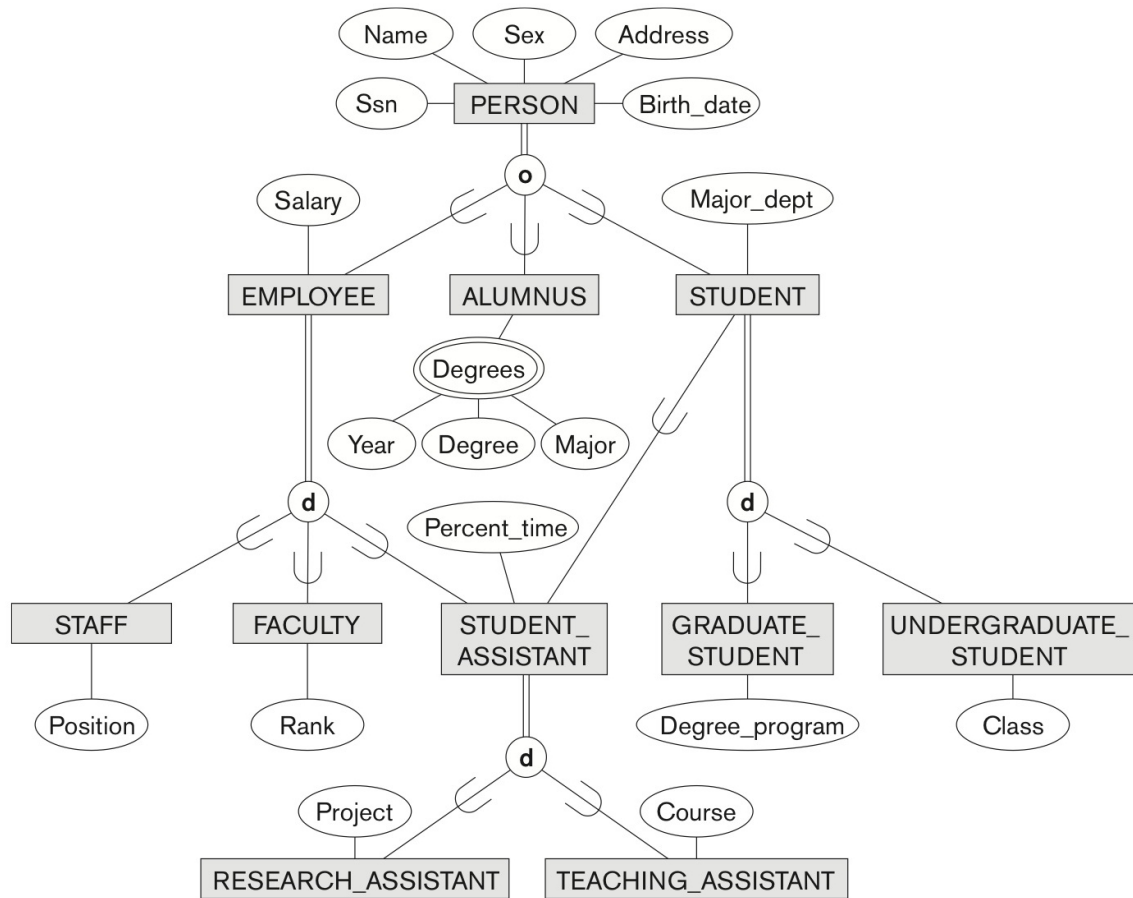


Uma instância de “account” só existe se for criada pela função `new()` na classe **Account**;

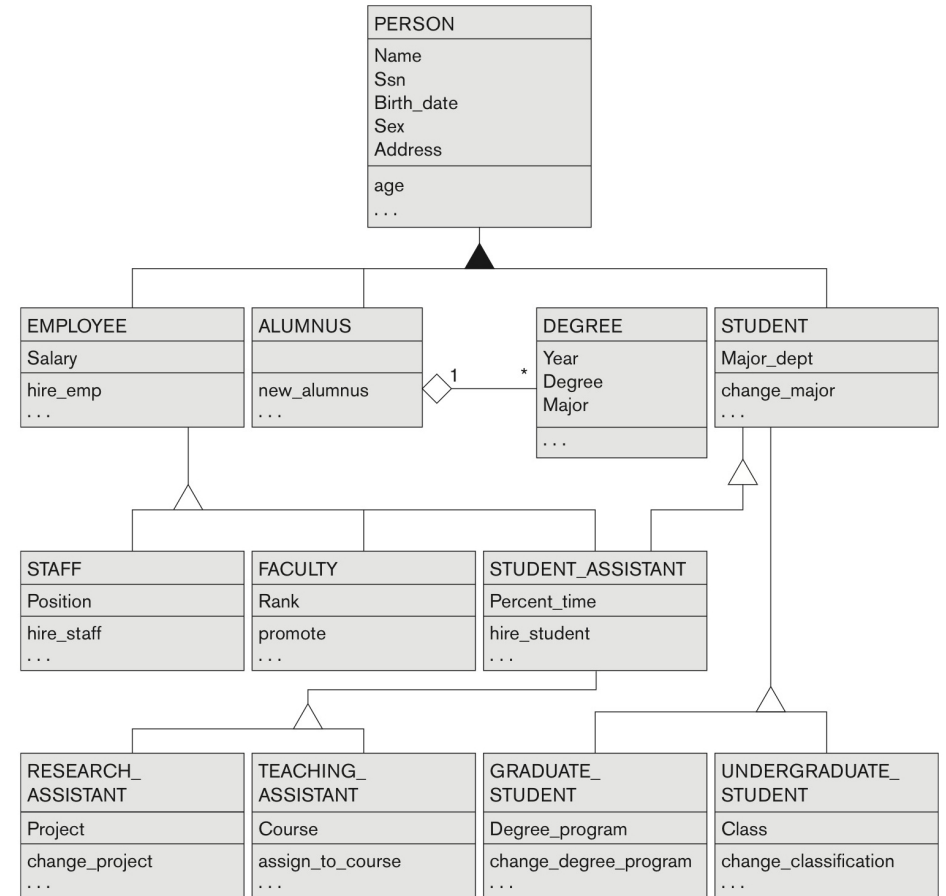
## Dependência (use)



O package **Web Shopping** depende do (usa o) package **Payment**



**ERE**



**UML**

... E muito mais!

- Classifier, templateable element, redefinable element, namespace, class, interface, association, data type, actor, use case, componente, collaboration, artifact, node, signal, encapsulated classifier, structured classifier, enumeration, property, attribute, member, part, visibility, private, public, id, readOnly, ordered, unique, nonunique, sequence, union, subset, property-constraint, multiplicity, public, private, protected, package, constraint (OCL), association ownership, navigability, arity, etc etc etc etc...

# Ferramentas

- ArgoUML (livre, gratuito, desktop)
  - StarUML (livre em parte, desktop)
  - Enterprise Architect (comercial, desktop, 30-day trial)
  - SoftwareIdeas (comercial, mas com community version)
  - VisualParadigm
  - ...
- 
- Muitas alternativas dentro do escopo de ferramentas corporativas complexas, dedicadas a Computer-Aided Software Engineering
    - Vão além da modelagem de dados, visando o projeto de todo o sistema

clodoveu@dcc.ufmg.br



Links



vCard