

XDES02

Programação Orientada a Objetos

Aula 03 – Modelos de Domínio

Objetivos



- Apresentar, de maneira pormenorizada, todos os componentes de um diagrama de classes
- Aprofundar o entendimento dos relacionamentos entre as classes
- Demonstrar que um diagrama de classes bem construído é capaz de representar o modelo conceitual de um sistema

Modelos de domínio

- Tratados pelos diagramas se classes
 - Aspectos abordados: dados e funções (operações)

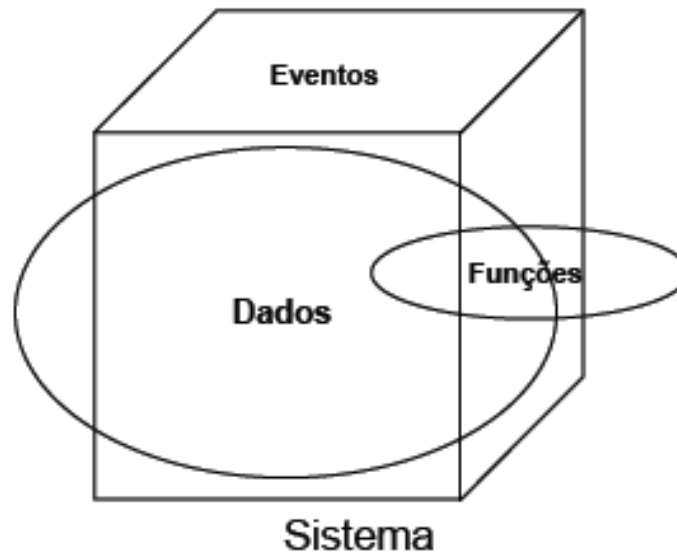


Diagrama de classes

- Notação

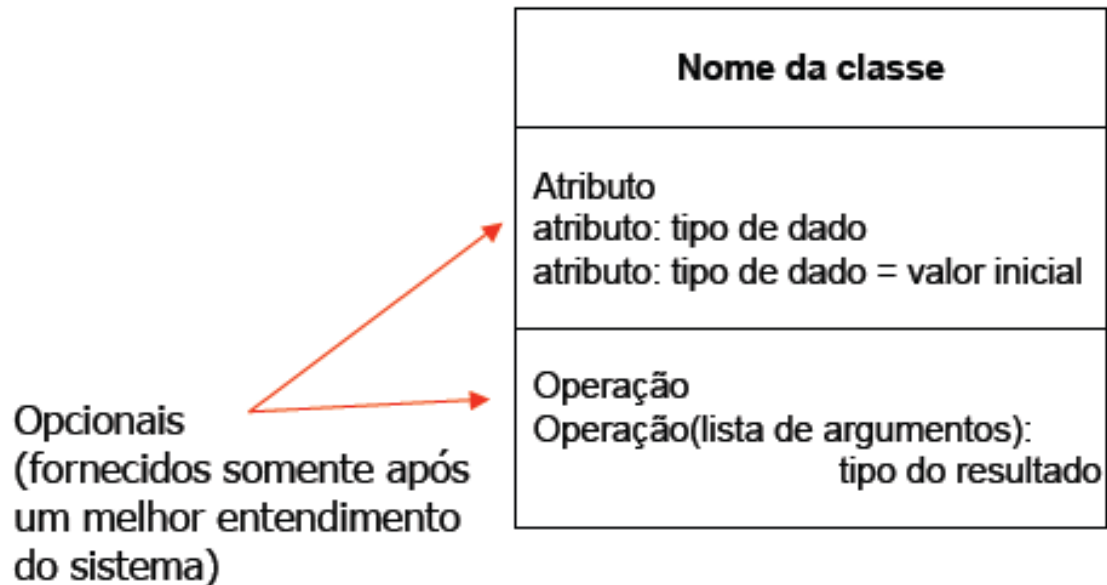
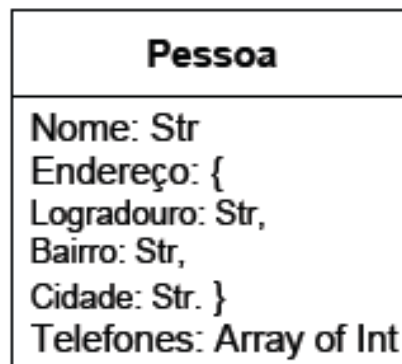


Diagrama de classes

- Atributos



Obs: Atributos compostos e Multivalorados são permitidos pelo modelo de dados OO

Diagrama de classes

- Associações

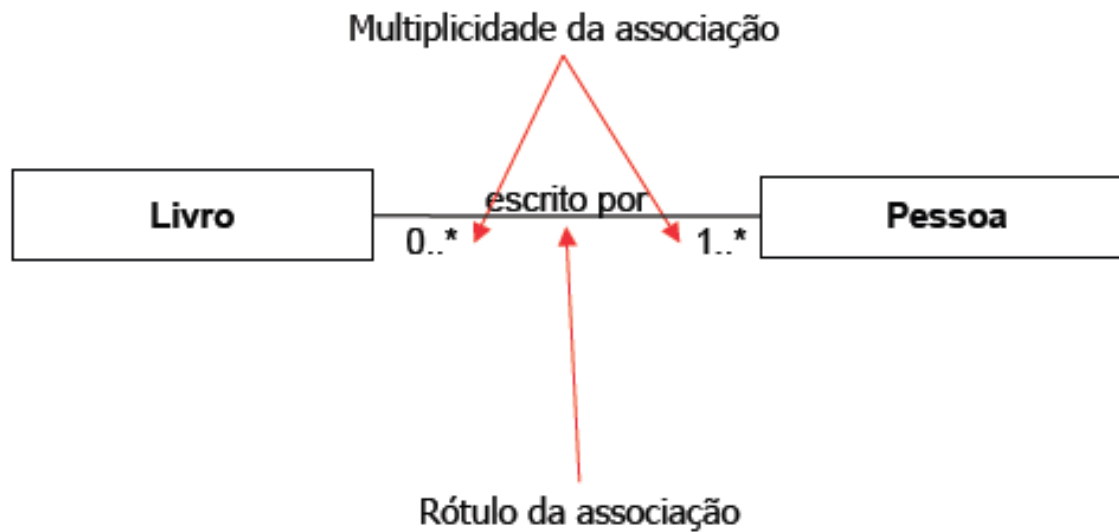


Diagrama de classes

- Associações + atributos

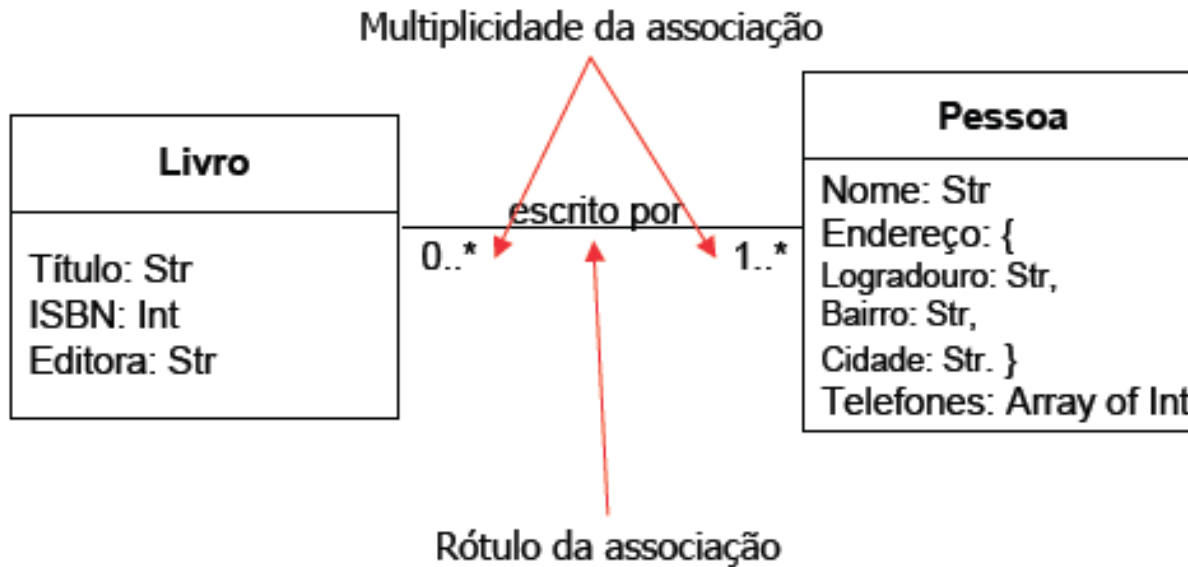


Diagrama de classes

- Associações + atributos + métodos

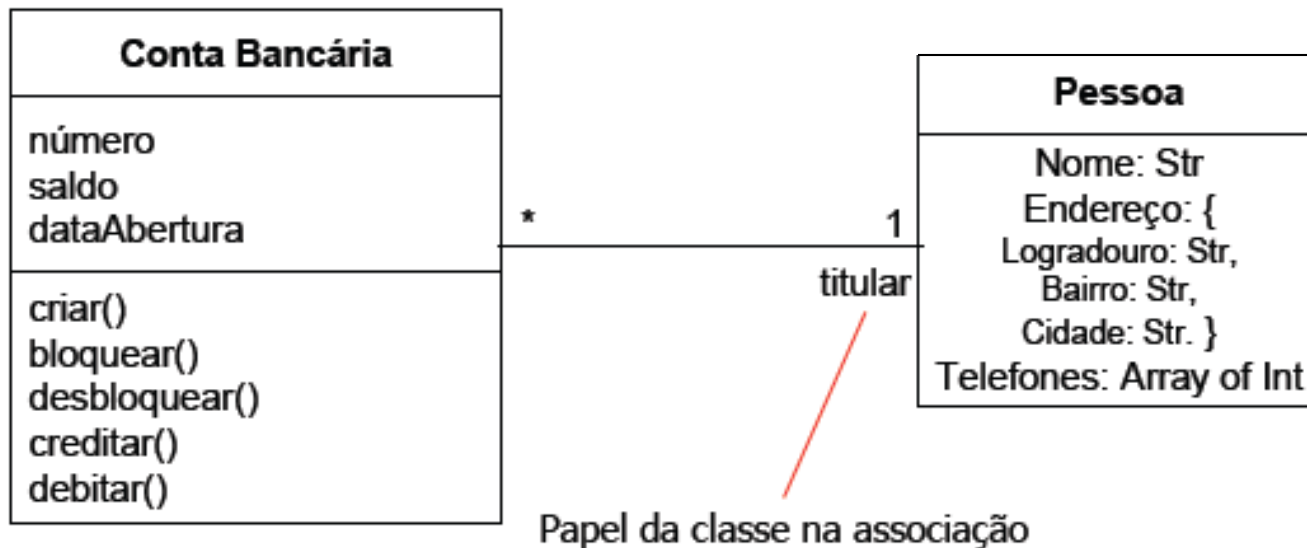


Diagrama de classes

- Associações entre objetos

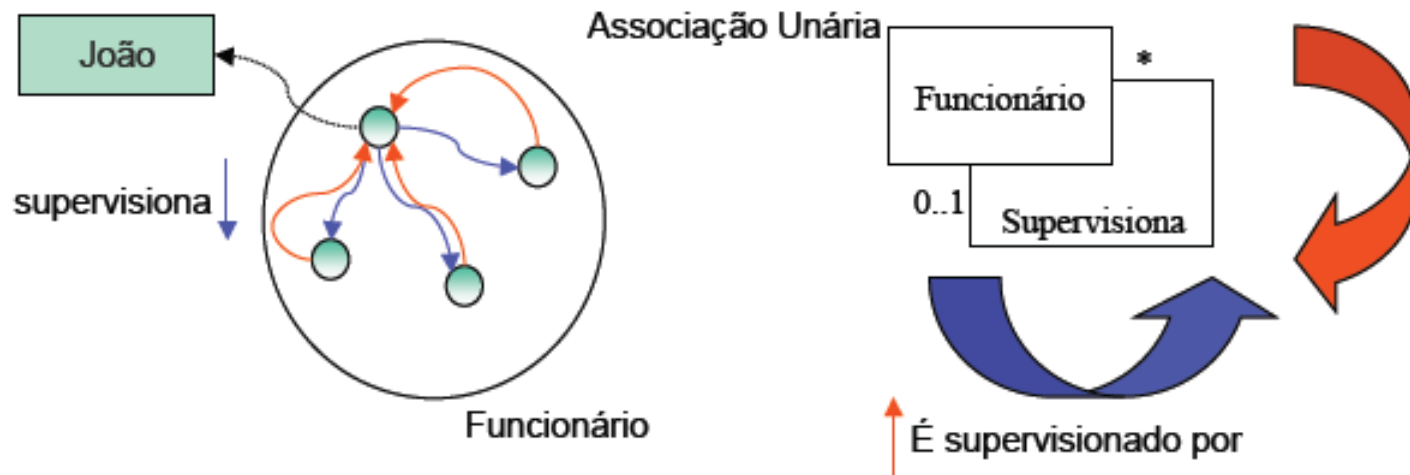
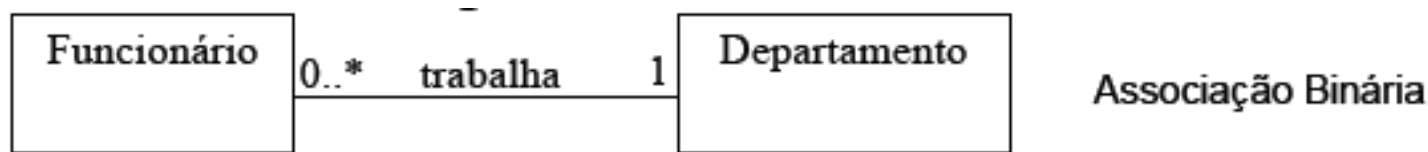


Diagrama de classes

- Navegabilidade: descreve como as associações devem ser navegadas
- Unidirecional
 - Só é possível navegar em uma direção
- Bidirecional
 - Navegabilidade ocorre nas duas direções

Diagrama de classes

- Associações entre objetos
 - Navegabilidade



Funcionário *trabalha* em Departamento

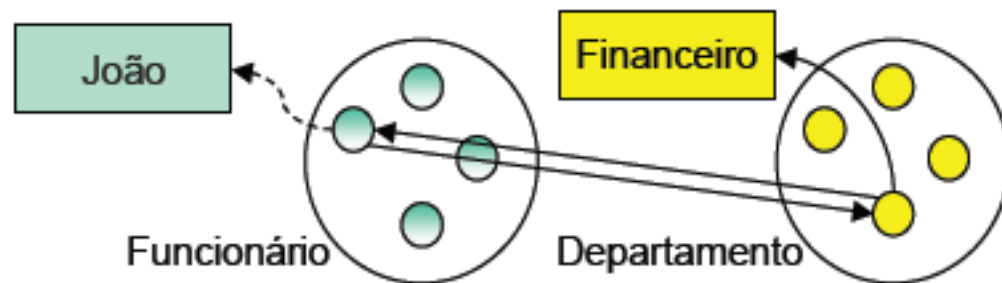


Diagrama de classes

- Associações entre objetos
 - Navegabilidade

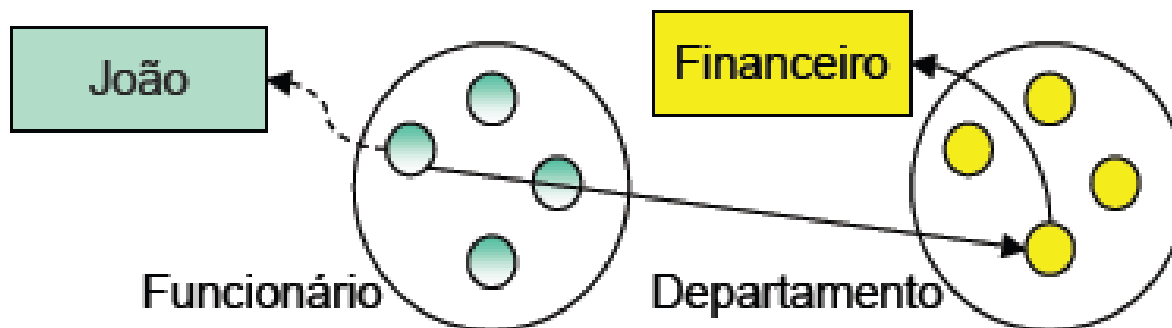


Diagrama de classes

- Associações entre objetos
 - Multiplicidade

| Multiplicidade | Significado |
|----------------|------------------------------|
| 0..1 | Zero ou um |
| 1 | Somente 1 (opcional) |
| 0..* | Maior ou igual a zero |
| * | Maior ou igual a zero |
| 1..* | Maior ou igual a 1 |
| 1..15 (m..n) | De 1 a 15 (m a n), inclusive |

Diagrama de classes

- Multiplicidade
 - Decisões sobre multiplicidades expõe muitas suposições
 - Exemplo:



Diagrama de classes

- Multiplicidade
 - Qual o correto?



Diagrama de classes

- Multiplicidade
 - Papéis

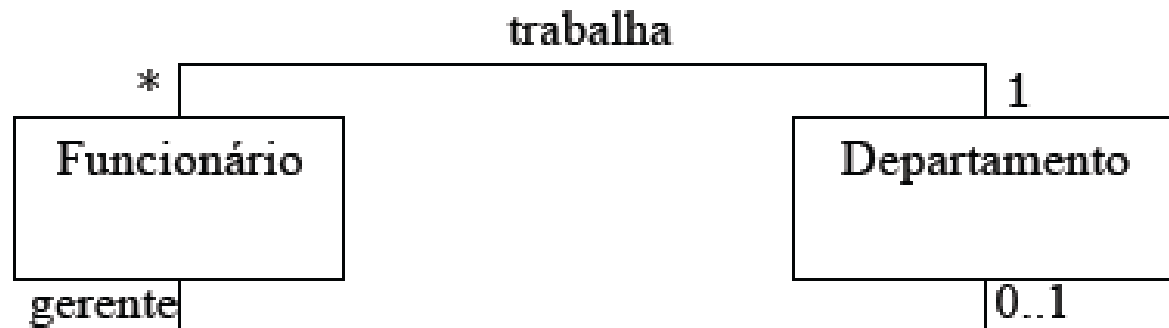


Diagrama de classes

- Multiplicidade
 - Outro exemplo

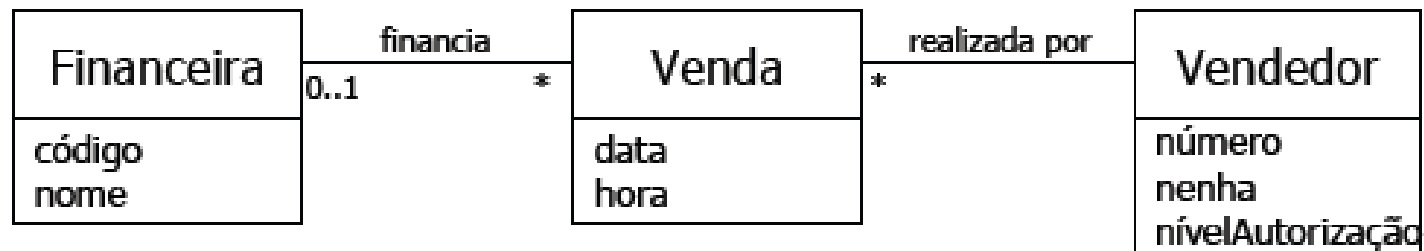


Diagrama de classes

- Classes associativas
 - Informação que surge a partir da associação de duas outras classes, ou de duas instâncias de uma outra classe

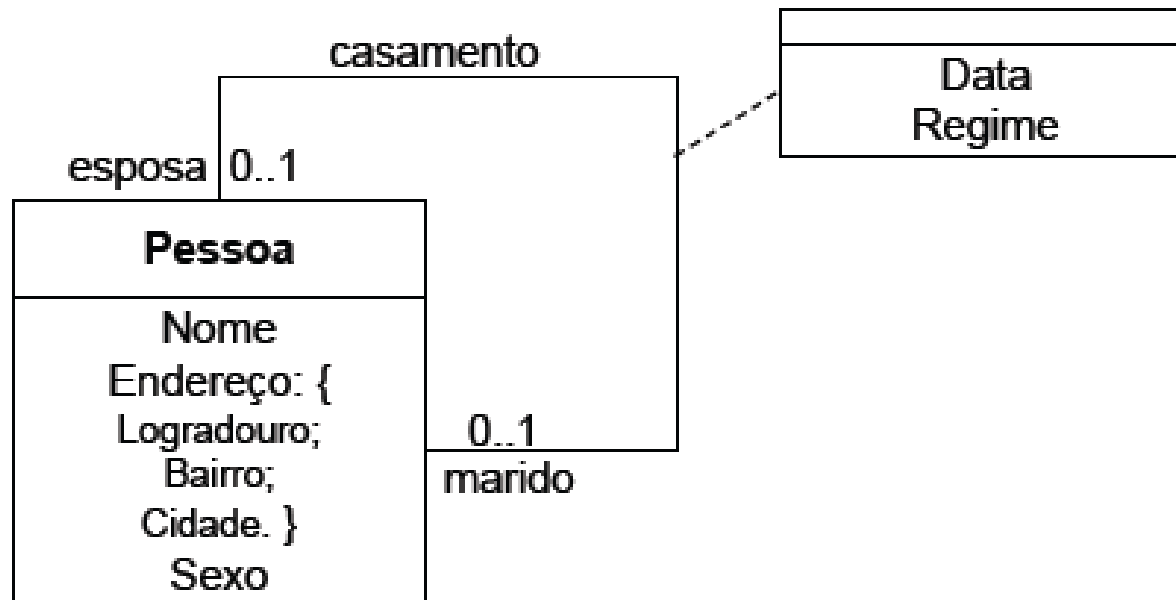


Diagrama de classes

- Classes associativas – exemplo
 - Conceito não é uma característica de Aluno nem uma característica de Disciplina
 - é uma propriedade do relacionamento entre ambos

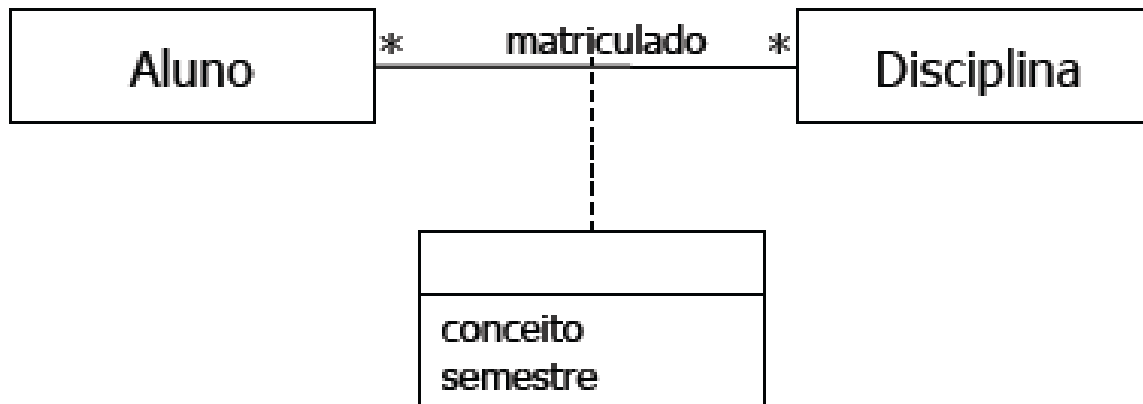


Diagrama de classes

- Classes associativas – usar
 - quando existem atributos que não pertencem às classes comuns
 - quando as classes comuns podem participar de associações com outras classes

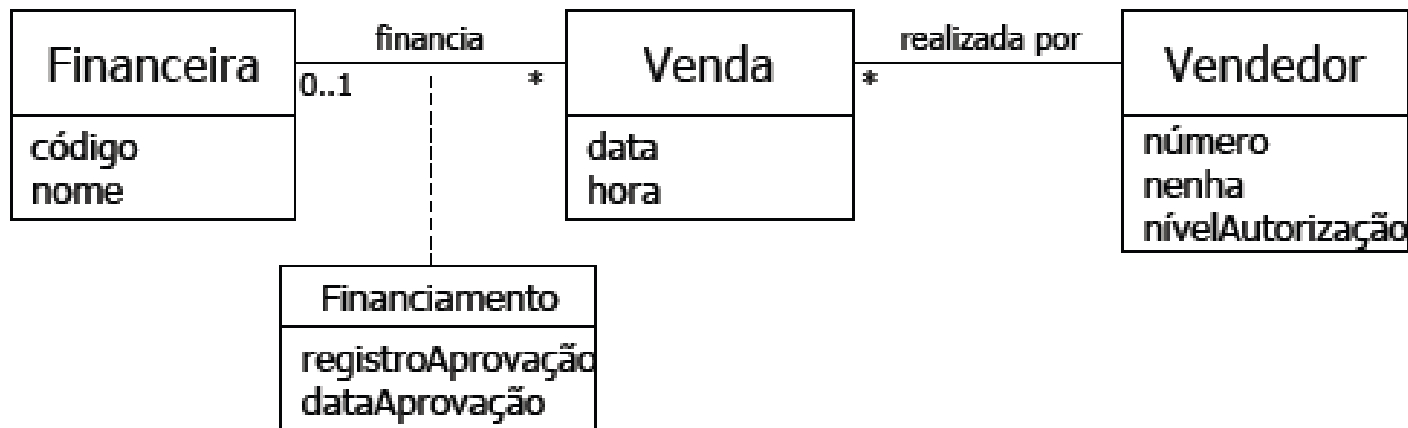
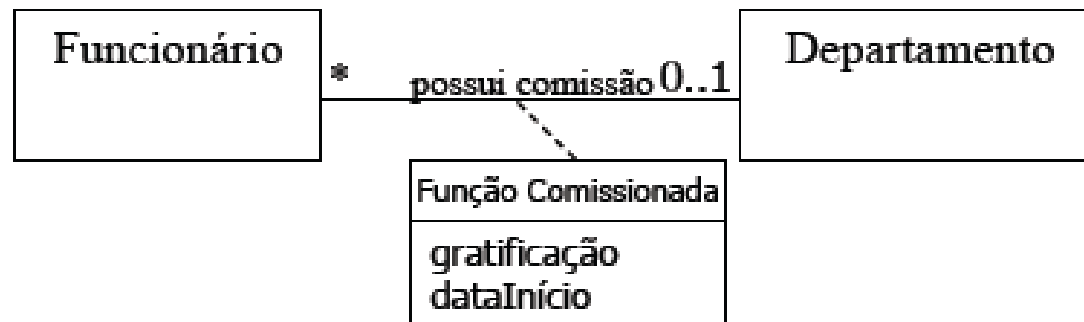


Diagrama de classes

- Classes associativas – obs. Importantes
 - O conceito de “Classe Associativa” não é permitido em todas as linguagens de programação e banco de dados OO
 - Assim, se for o caso, deve-se substituir as classes associativas por classes regulares

Diagrama de classes

- Classe associativa substituída por classe normal



Exercício: definir a multiplicidade para manter o mesmo significado do modelo acima

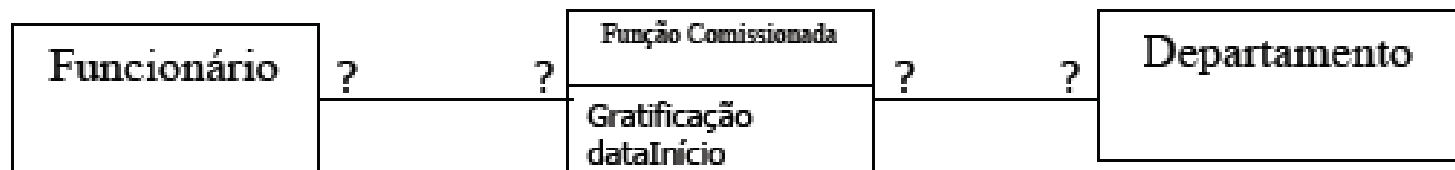


Diagrama de classes

- Agregação
 - Tipo especial de associação “todo-parte”
 - Ação realizada sobre o todo atinge as partes

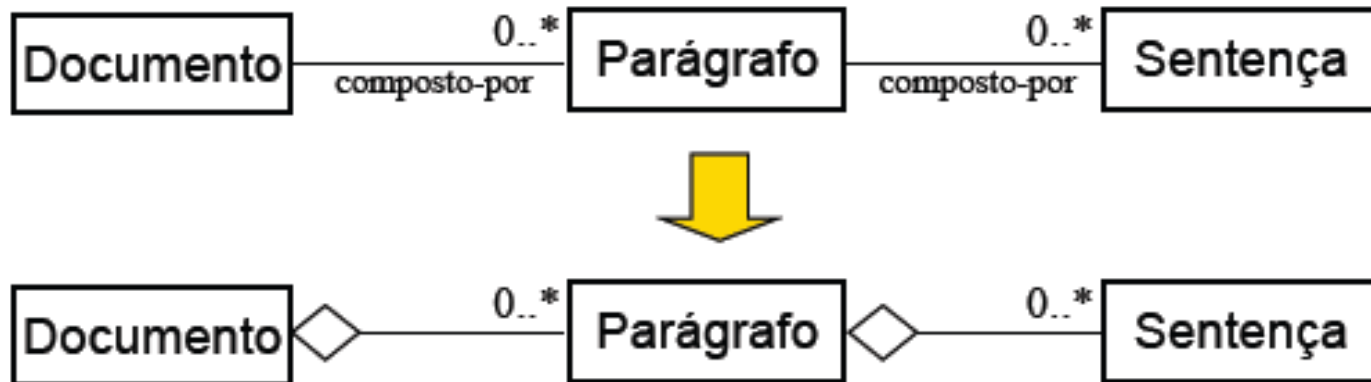


Diagrama de classes

- Agregação – exemplo



Diagrama de classes

- Agregação vs Composição
 - Composição é um tipo especial de agregação
 - Semanticamente equivalente a um atributo

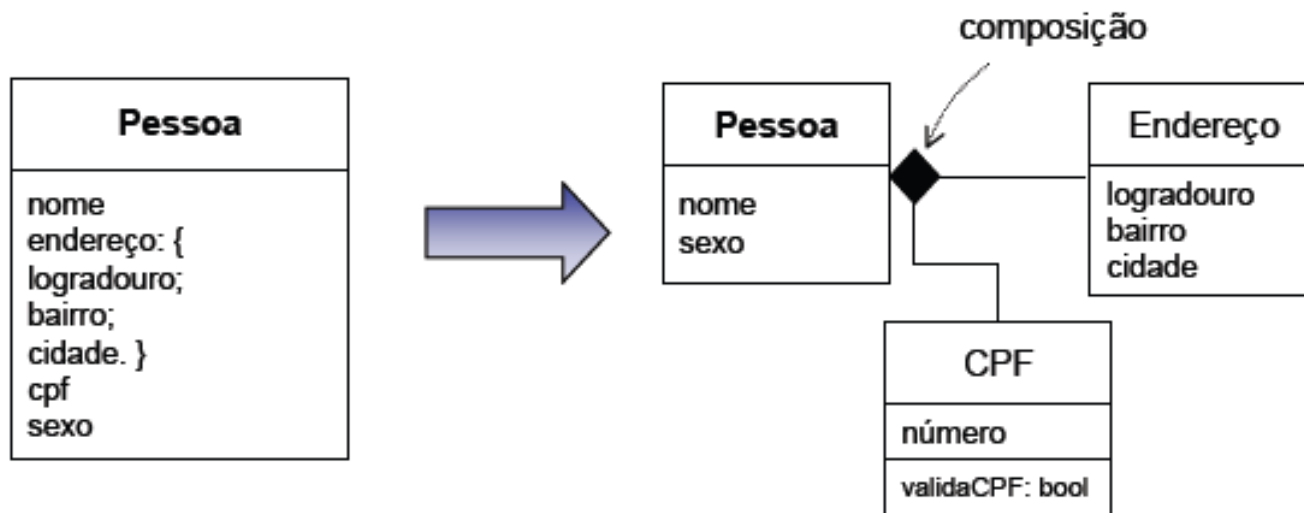


Diagrama de classes

- Composição
 - A remoção do todo implica na remoção das partes
 - O acesso às partes é restrito ao todo

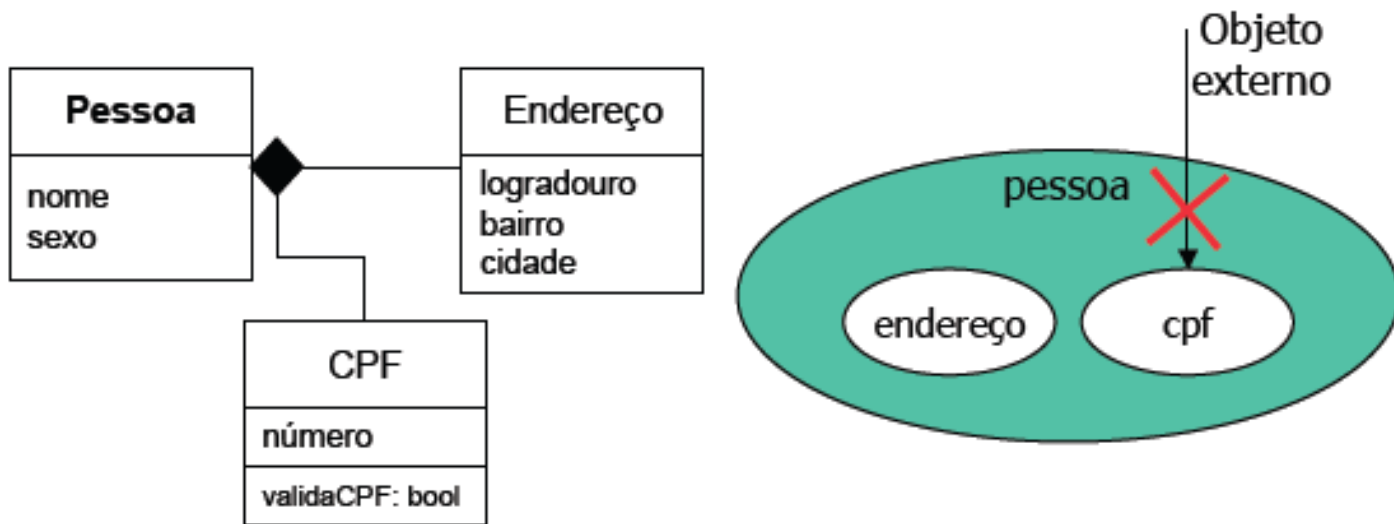
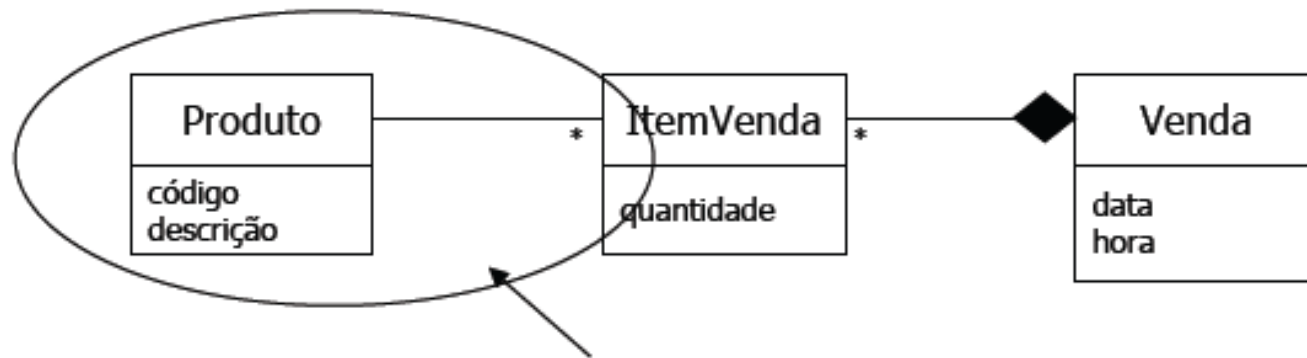


Diagrama de classes

- Composição
 - Exemplo



Uso inadequado de composição:
Partes de uma composição não podem ser referenciadas
por objetos externos

Diagrama de classes

- Composição
 - Exemplo

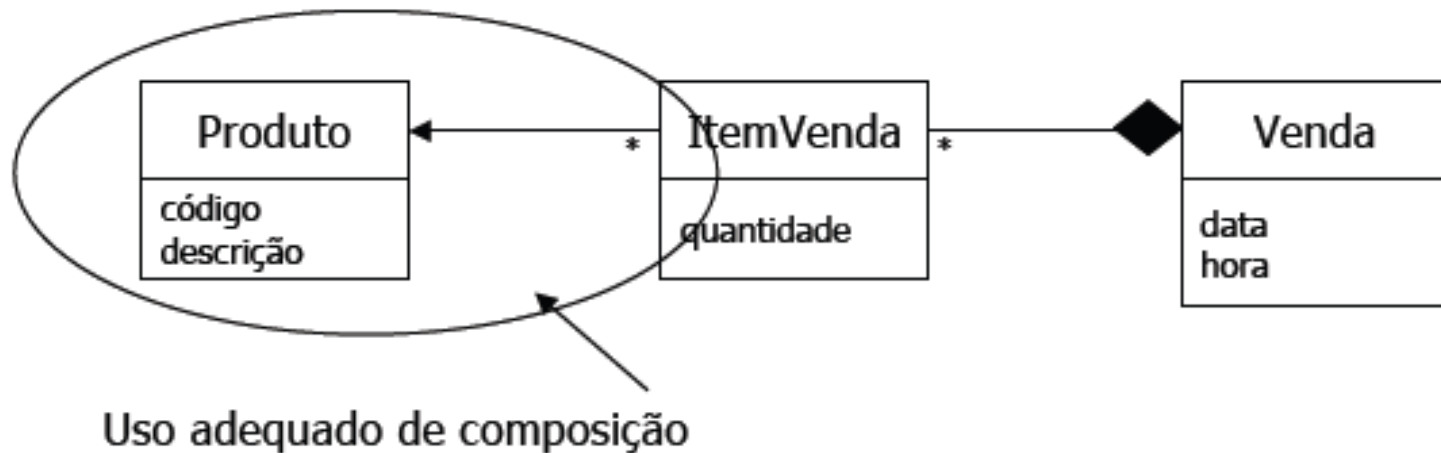


Diagrama de classes

- Generalização / Especialização
 - Herança de propriedades
 - Relações do tipo “é um”

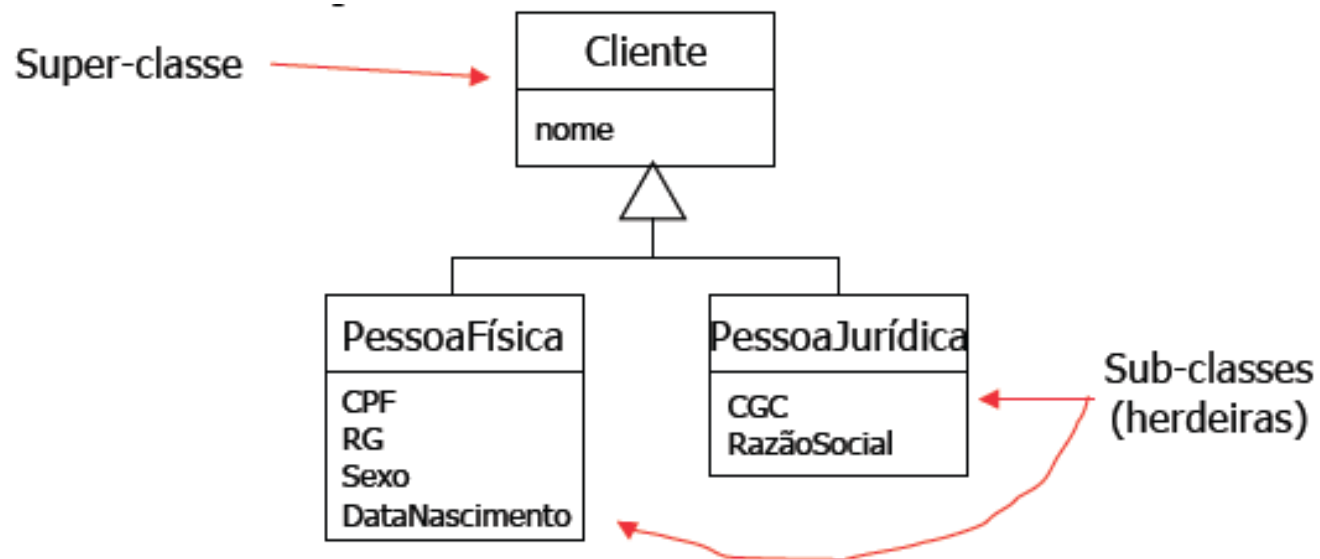


Diagrama de classes

- Generalização / Especialização
 - Polimorfismo: não há necessidade de se criar uma associação entre Compra e subclasses de Cliente

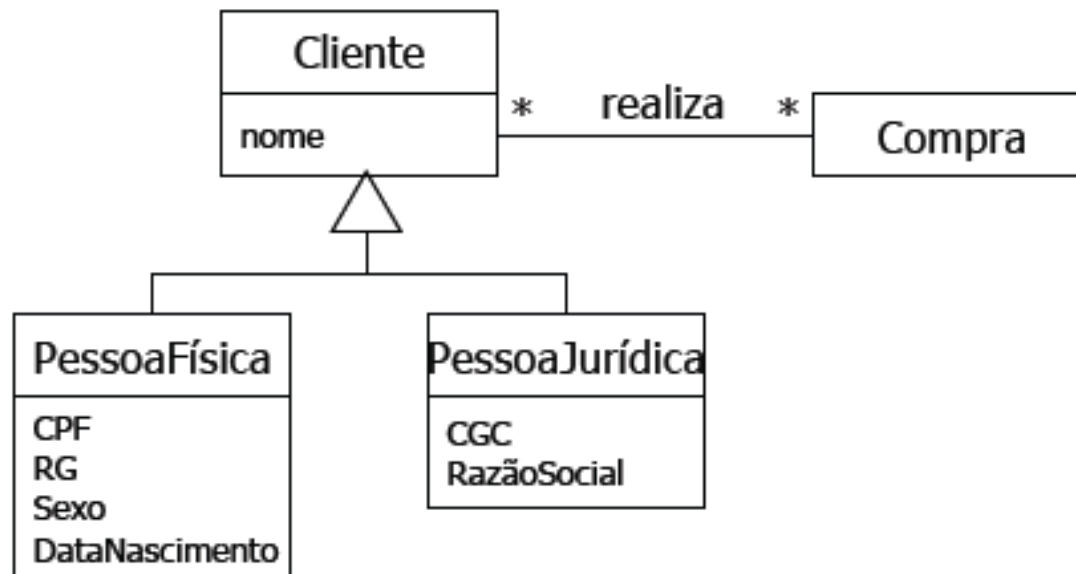


Diagrama de classes

- Generalização / Especialização
 - Classes abstratas

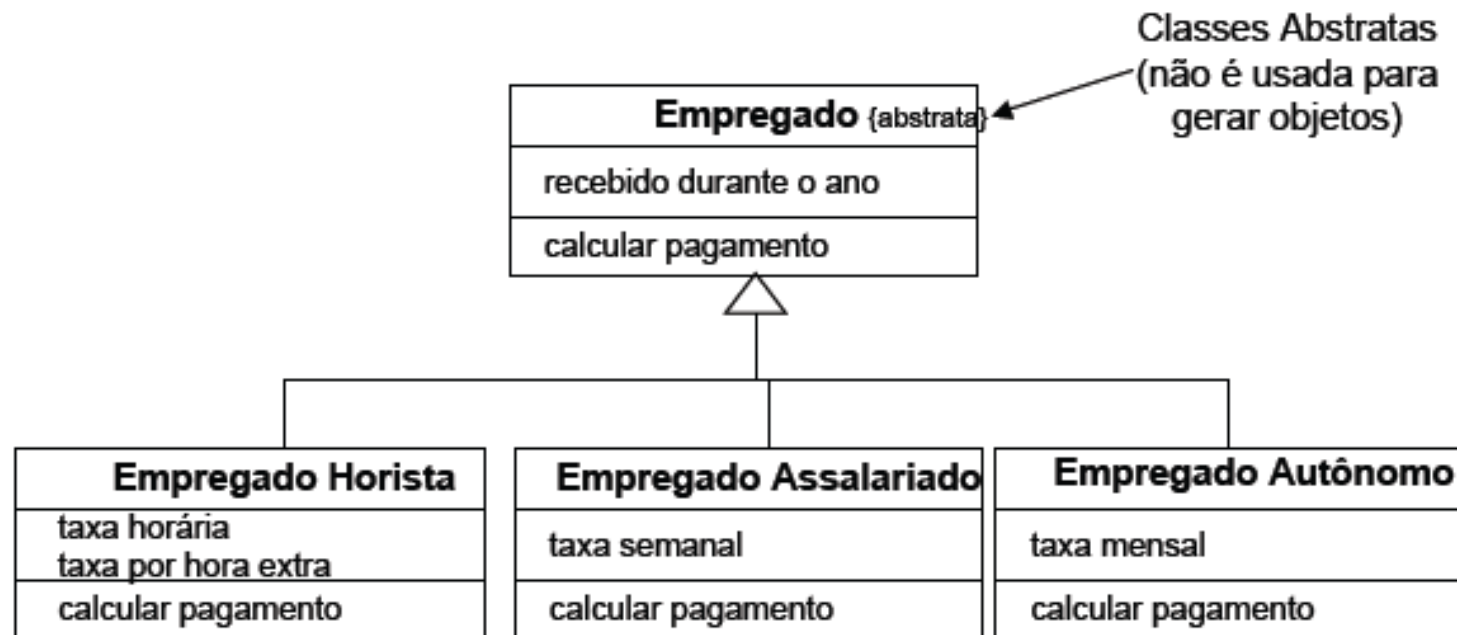
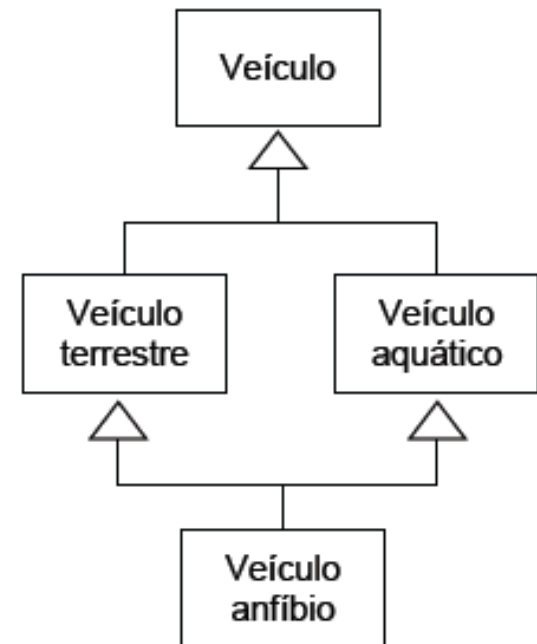


Diagrama de classes

- Generalização / Especialização – quando usar
 - Deve-se analisar as especificações para verificar quais delas estão no domínio do problema
 - As que não estão presentes não devem ser representadas
 - Por exemplo, a modelagem de homem e mulher não precisa ser feita, pois o pagamento não é feito em função do sexo do indivíduo
 - Assim não se deve criar subclasses. A especialização, nesse caso, é feita através da utilização de um atributo “sexo”

Diagrama de classes

- Herança múltipla
 - Conceito pouco usado na prática
 - Não é suportado por todas as linguagens OO
 - Adiciona maior complexidade ao modelo

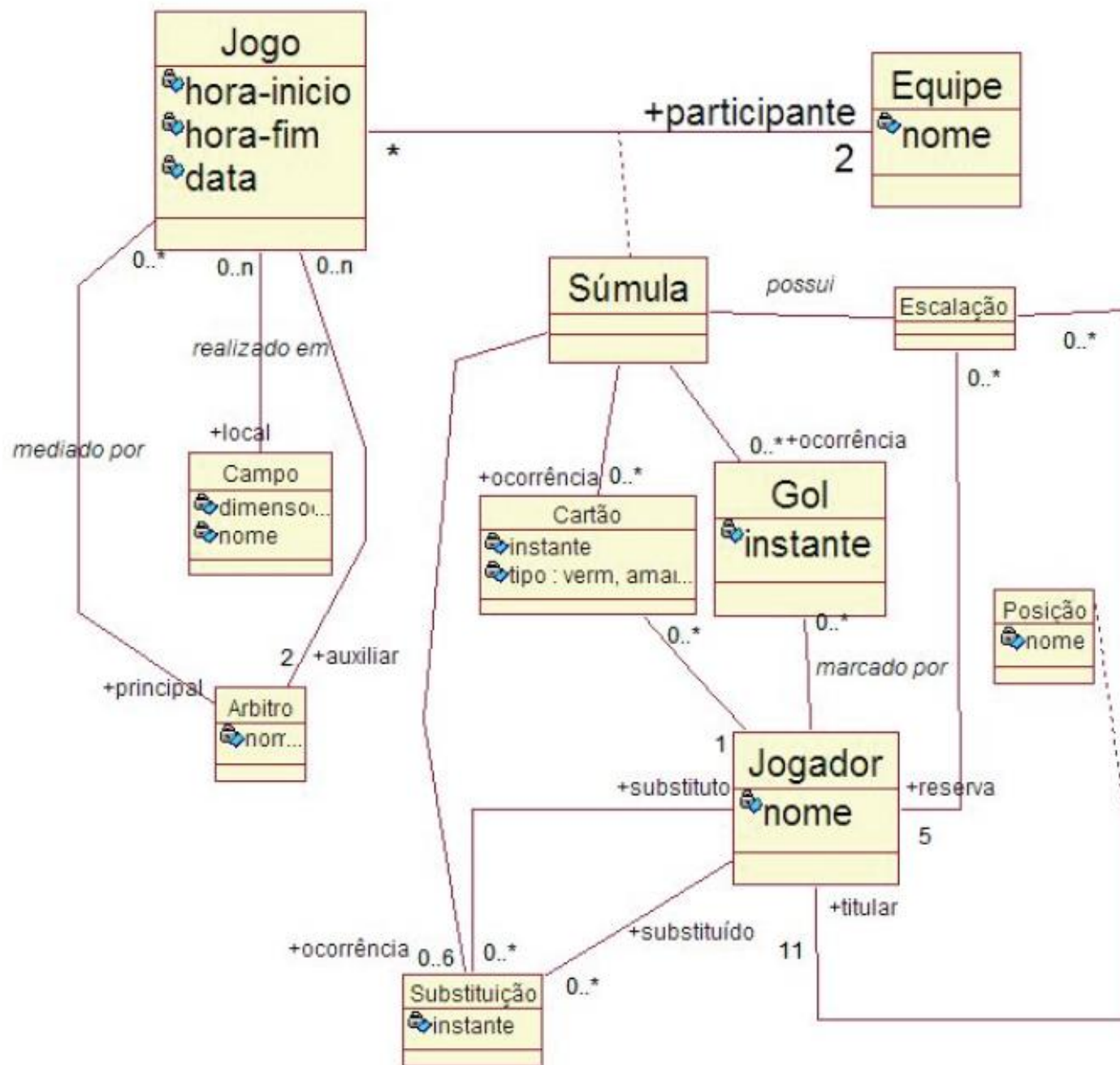


Dicas

- Não comece a projetar diretamente o diagrama de classes. Primeiro entenda o problema.
- Mantenha o modelo de objetos o mais simples possível. Evite complicações desnecessárias.
- Escolha os nomes cuidadosamente
 - Se você não consegue escolher um bom nome, é provável que sua classe esteja mal projetada
- Não é necessário utilizar todos os conceitos da UML
- Evite árvores de herança muito profundas
 - Classes profundas são mais complexas

Exemplo de modelagem

- Usar classes e associações para modelar um jogo de futebol, descrito a seguir:
 - O jogo de futebol é realizado por 2 equipes de 11 jogadores cada. Cada jogador tem uma função: goleiro, zagueiro, meio-campo, atacante, lateral e ponta de lança. O jogo é realizado num campo com medidas retangulares (em comprimento e largura), tem duas metas, cada qual em lados opostos do campo. Ganha o jogo a equipe que marcar mais gols (i.e. colocar a bola na meta do adversário). No jogo existe uma única bola que apresenta características (peso, diâmetro) regulamentares. O jogo de futebol é mediado por uma equipe de 3 árbitros, sendo um deles o árbitro principal e os dois outros chamados auxiliares...

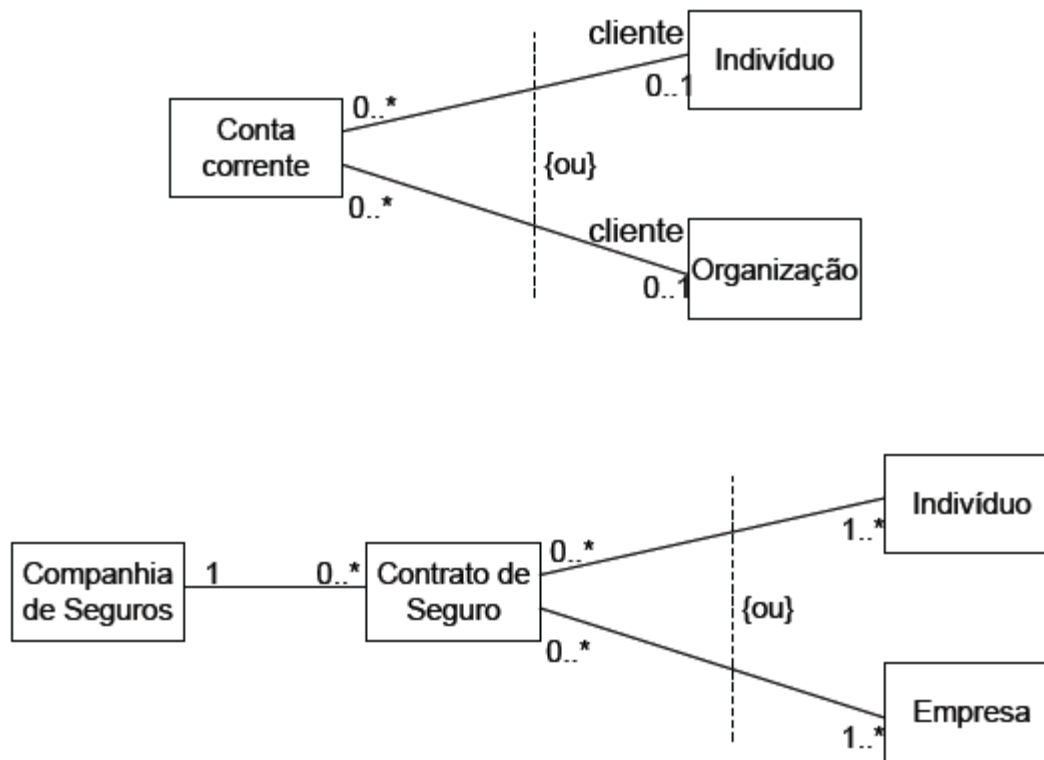


Evento Lembrado

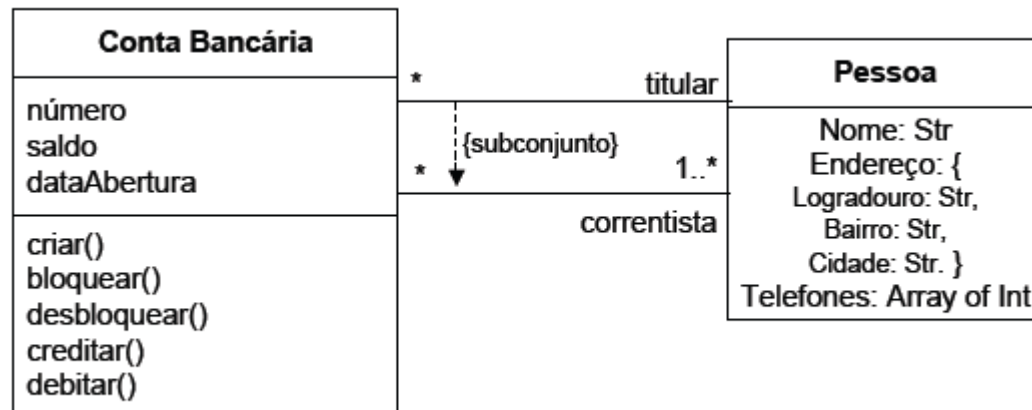
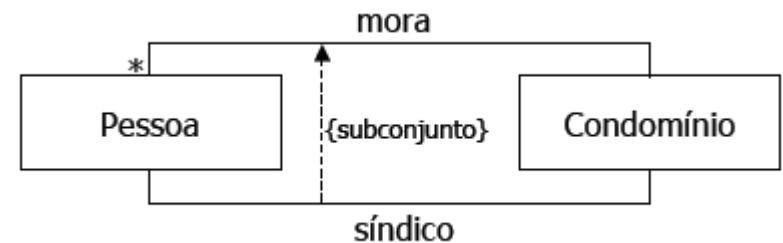
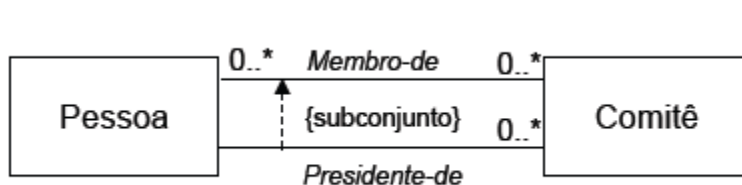
- Todo evento que precisa ser registrado e lembrado pelo sistema pode ser considerado um potencial candidato a classe
- Idéia: Existe algum instante ou evento histórico que precisa ser lembrado pelo sistema?
 - Um sistema de registro de carros tem que lembrar quando alguém comprou um carro
 - Lembrar quando alguém reservou/alugou um jogo
 - Matrícula de um aluno
 - Aumento de salário de funcionário

Restrição {ou}

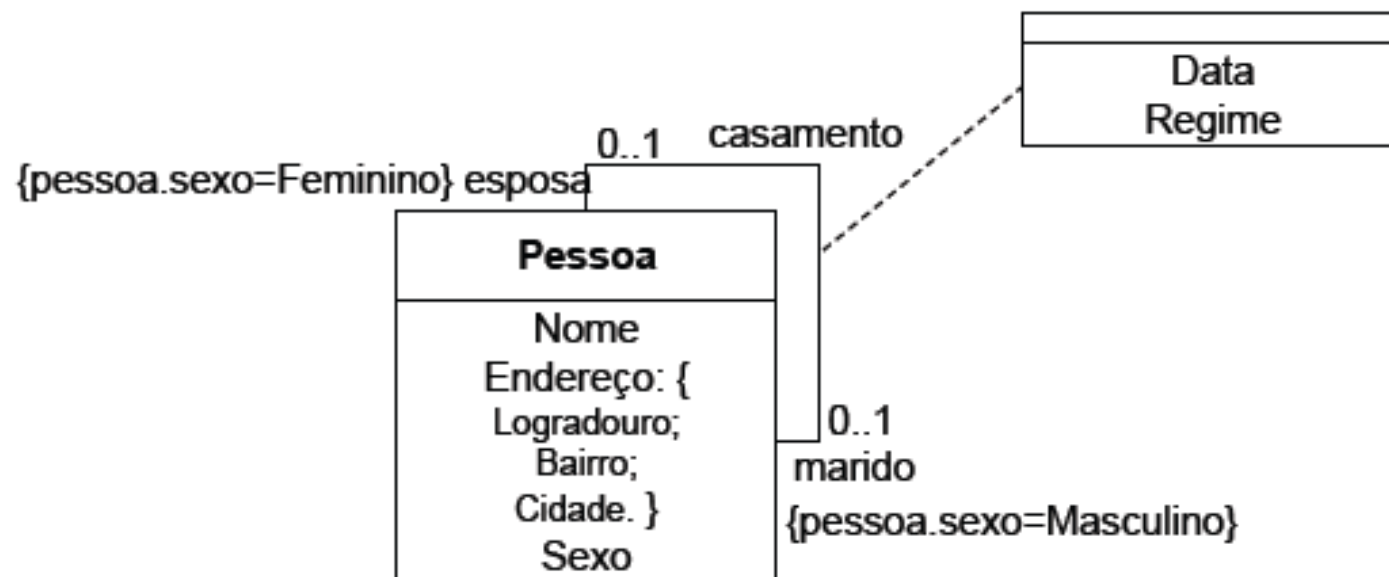
- Implica na seleção exclusiva entre duas ou mais associações existentes em uma classe



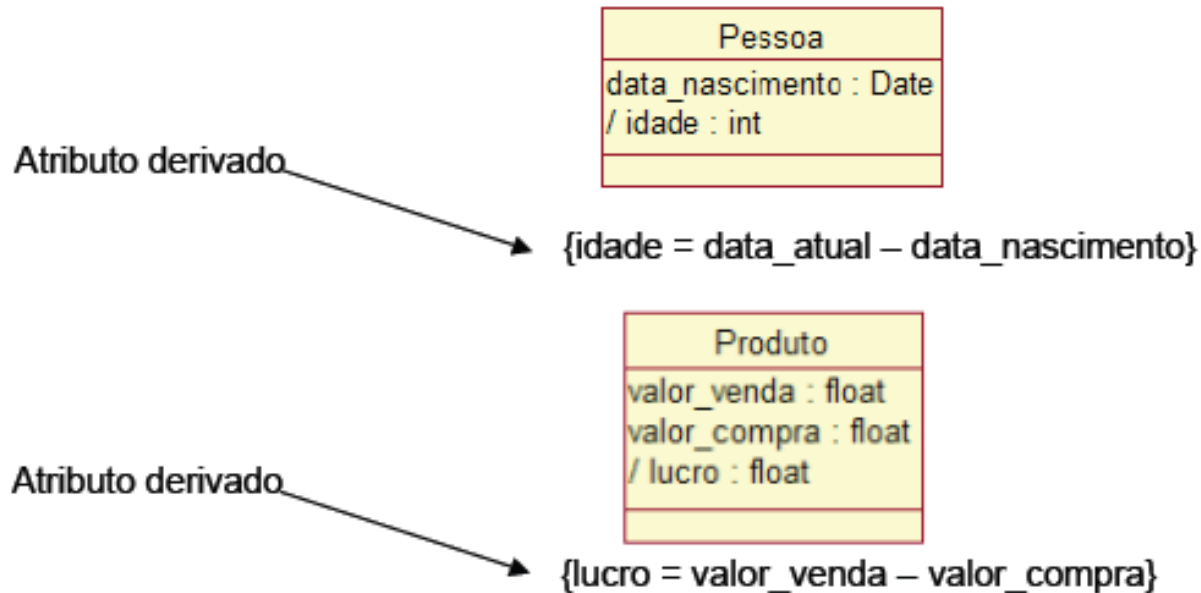
Restrição {subconjunto}



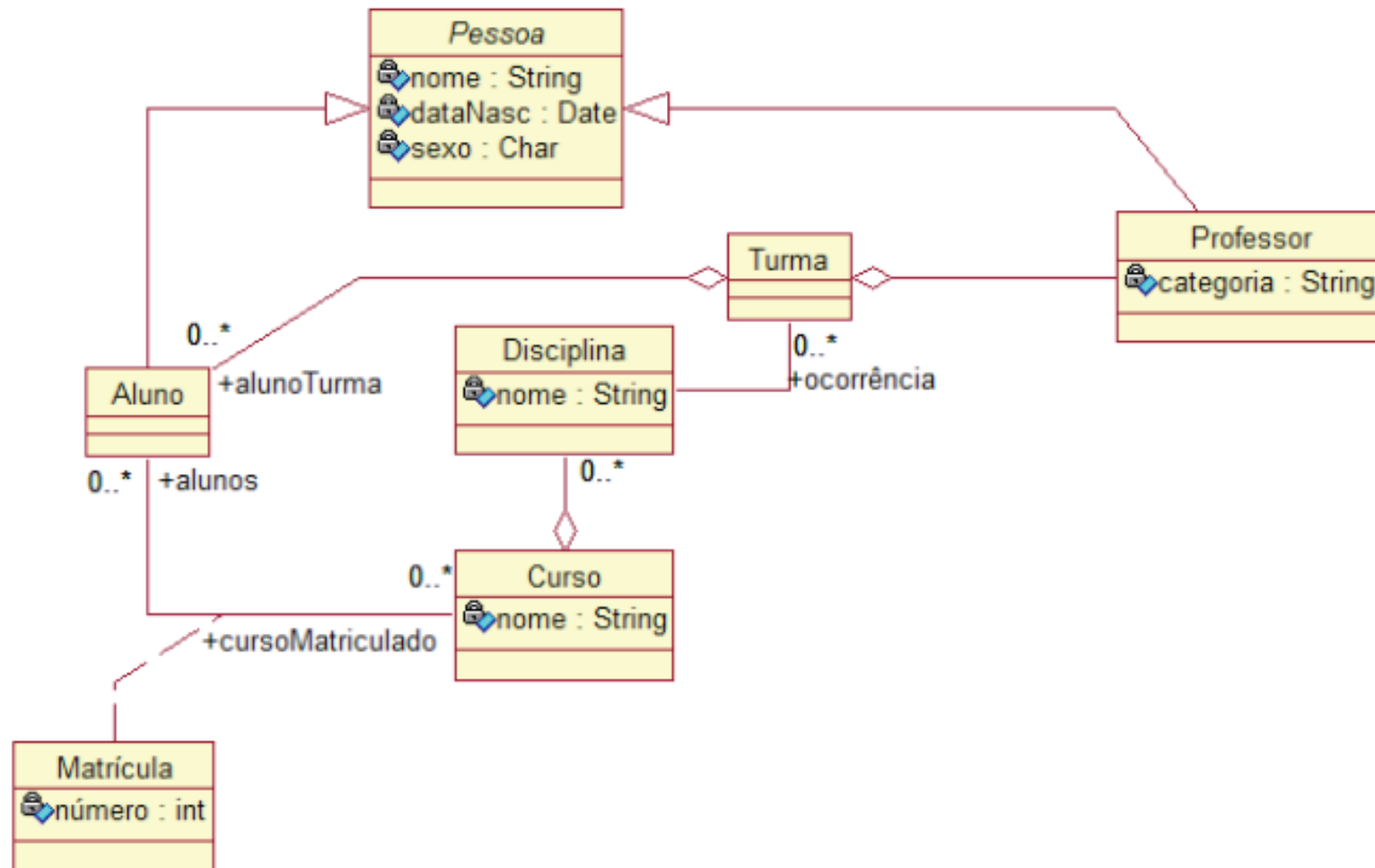
Restrições



Atributo Derivado

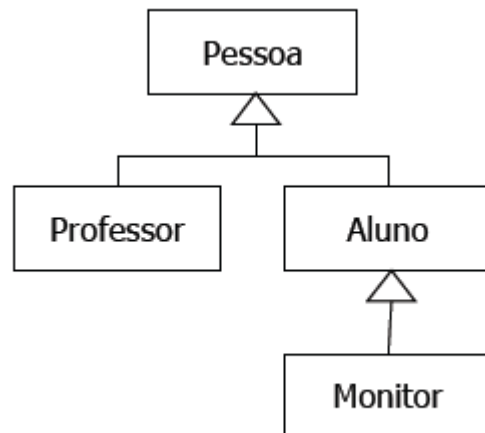


Considere o seguinte modelo de Controle Acadêmico



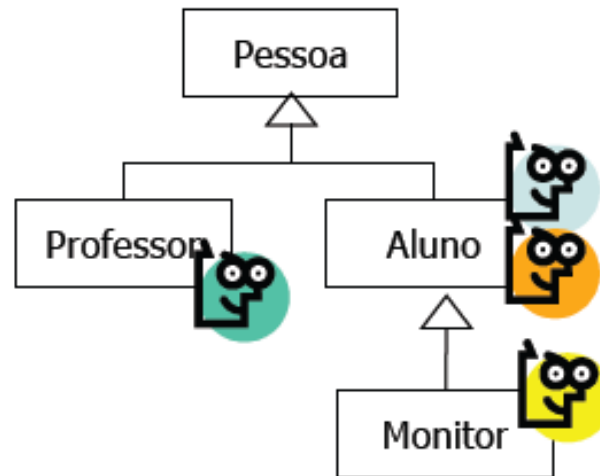
Transmutação

- Complementar o modelo de controle acadêmico
 - Acrescentar a figura do “monitor”
 - Aluno pode exercer o papel de monitor durante um período
 - Permitir que uma pessoa exerça simultaneamente o papel de Professor e Aluno da universidade
- Possível solução:



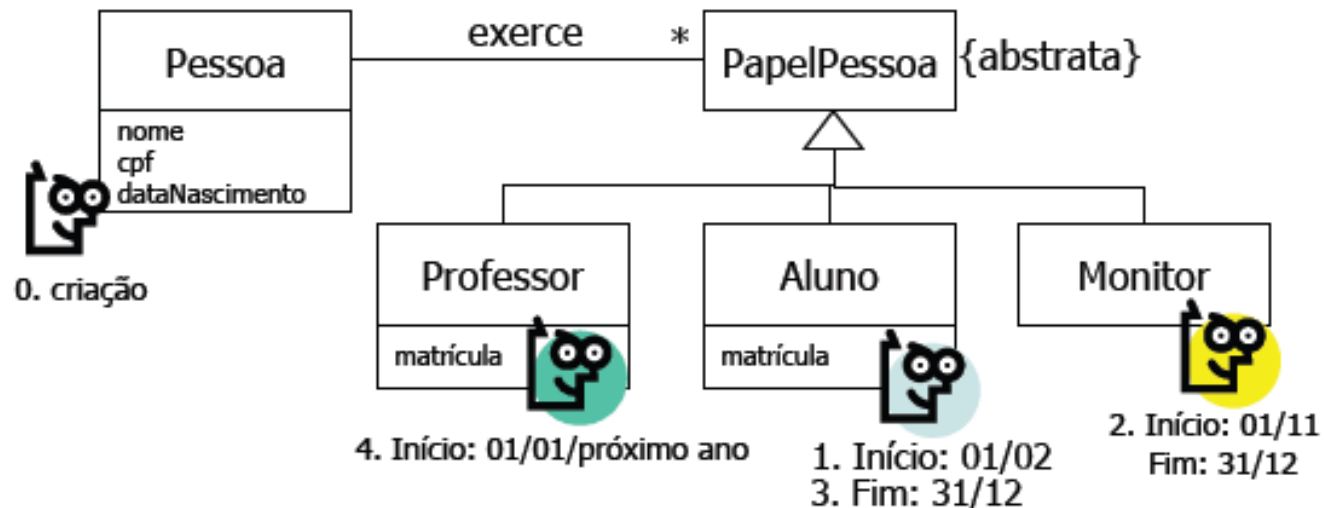
Transmutação

- Problemas
 - Acomodação inadequada de objetos que mudam de classe
 - Transmutação ou metamorfose



Transmutação

- Solução
 - Combinar herança e associação



Exercício - Transmutação

- Em uma certa universidade, o plano de carreira para professores é dividido em quatro etapas
 - Auxiliar, Assistente, Adjunto e Titular
- Construa um modelo de classes que preserve o histórico de um professor durante sua vida profissional
 - Deve-se manter o registro do início e término do desempenho em cada uma das etapas