


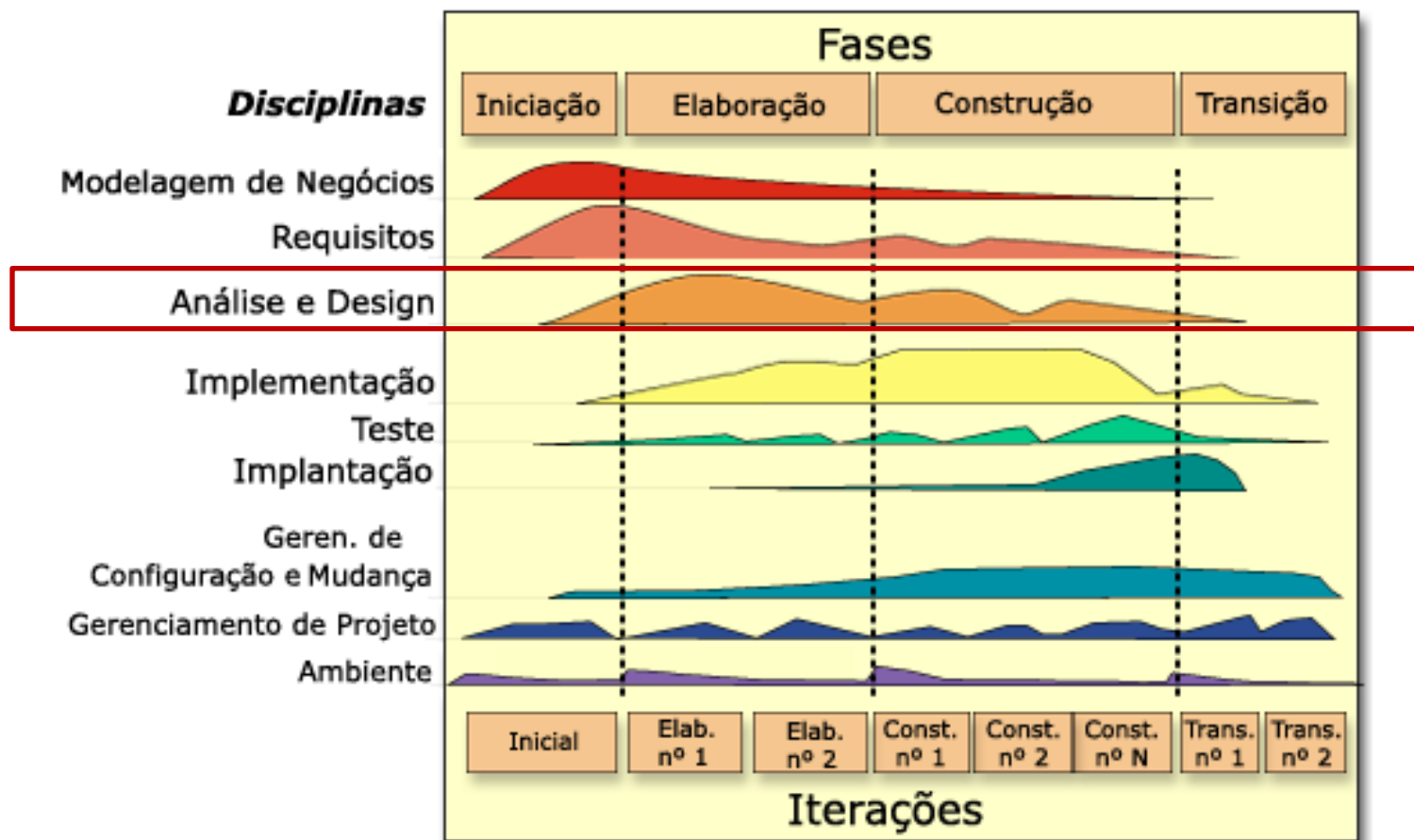


Modelo conceitual – Aula 08

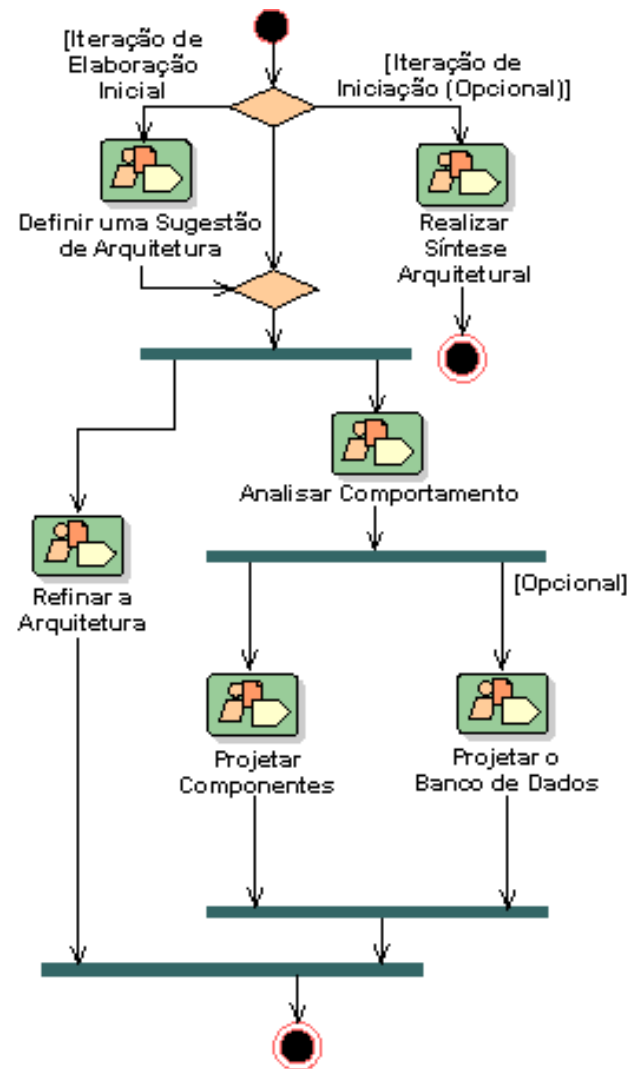
Prof. Fernando Maia da Mota

Slides gentilmente cedidos por Profa. Dra. Maria Istela Cagnin Machado
UFMS/FACOM

- 
- *Alguns slides foram preparados pela professora Dra. Rosana T. Vaccare Braga para a disciplina de Análise e Projeto Orientado a Objetos, ICMC-USP. Agradecimentos pela disponibilização...*

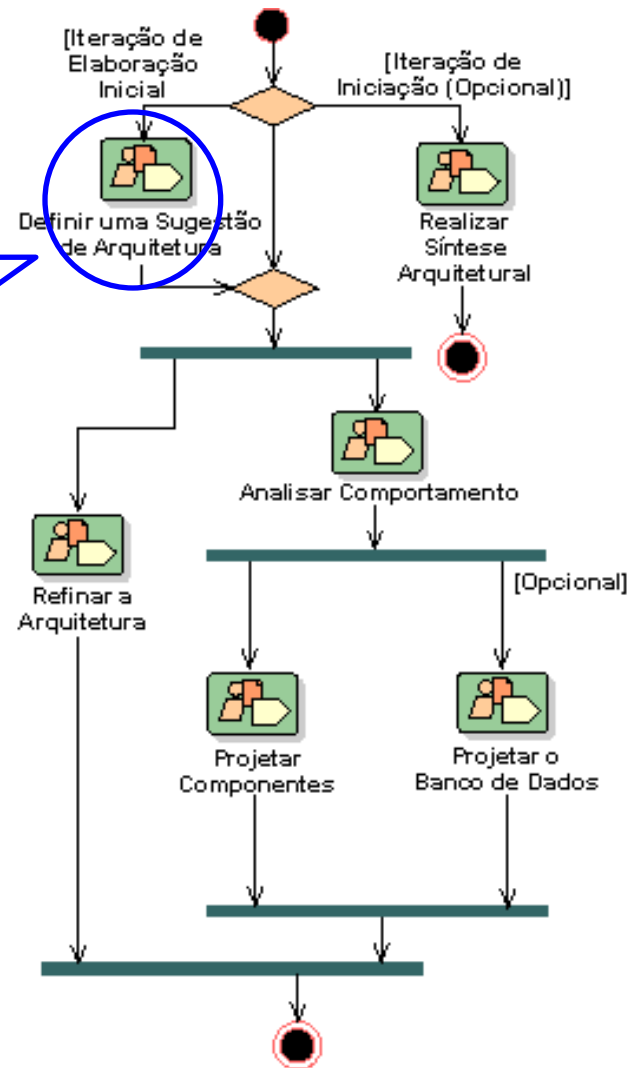


Disciplina: Análise e *Design*



Disciplina: Análise e *Design*

Artefatos (Análise):
Modelo Conceitual,
Diagramas de
Sequência do
Sistema (DSS) e
Contratos de
operações e
consultas

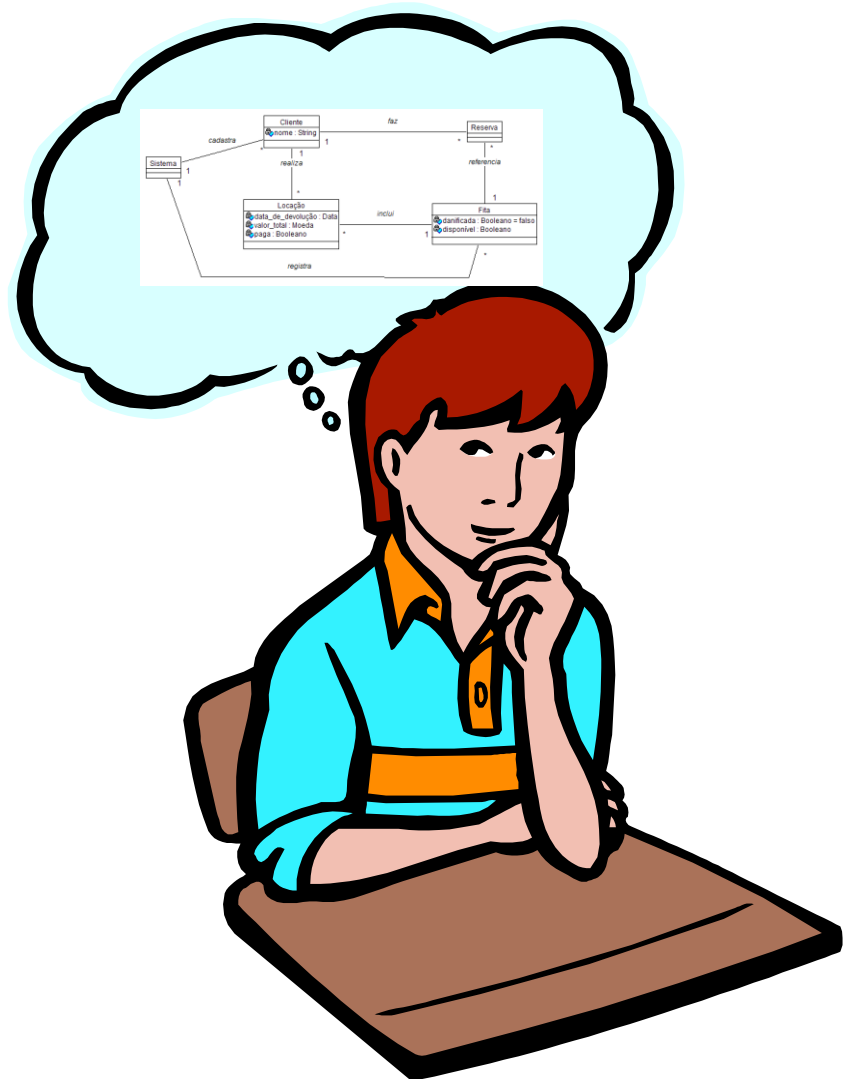




Modelo Conceitual

- artefato do domínio do problema e não do domínio da solução
- não deve ser confundido com a arquitetura do software
- não deve ser confundido com o modelo de dados
- amplamente utilizado como fonte de inspiração para projetar objetos de software

Onde está o Modelo Conceitual?





Modelo Conceitual

- O passo mais essencialmente OO na análise é a decomposição de um domínio em conceitos ou objetos importantes
- O modelo conceitual é uma representação visual de classes conceituais ou objetos do mundo real, em um domínio



Modelo Conceitual

- Também chamado de
 - Modelo de domínio
 - Modelo de objetos
 - Diagrama de classes de análise

Modelo Conceitual

- Aplicando a notação UML, um modelo conceitual é ilustrado como um diagrama de classes, fornecendo uma perspectiva conceitual e podendo mostrar:
 - objetos do domínio ou classes conceituais
 - associações entre classes conceituais
 - atributos de classes conceituais

Modelo Conceitual

- Um modelo conceitual mostra classes *conceituais* de uma situação real, não classes de software

Modelo Conceitual

- Mostra todos os **conceitos** importantes no domínio do sistema, bem como as associações entre esses conceitos
- A ideia é fazer com que o usuário que tem acesso a esse modelo entenda os principais elementos do domínio



Modelo Conceitual

- O modelo conceitual ajuda a esclarecer a terminologia ou vocabulário do domínio



Modelo Conceitual

- O objetivo da análise é estudar o problema
 - o sistema computacional seria uma solução para o problema, logo, objeto da fase de projeto
- O analista deve lembrar que a fase de análise considera apenas o mundo exterior ao sistema e nunca seu interior

Modelo Conceitual

Modelo conceitual deve ser independente da solução física que virá a ser adotada e deve conter apenas elementos referentes ao domínio do problema

- ficam delegados à fase de projeto os elementos da solução que se referem a computação, como:
 - interfaces, formas de armazenamento (banco de dados), segurança de acesso, comunicação, etc.



Elementos Básicos do MC

- Conceitos
- Atributos
- Relacionamentos entre conceitos

Elementos básicos do MC

- Quando se trabalha modelagem conceitual com UML, existem 3 elementos para representar informação:
 - **Conceitos**, que são representados por **classes**:
representação da informação complexa, que não pode ser descrita meramente por tipos alfanuméricos
 - Exemplos: fita, cliente, reserva

Elementos básicos do MC

- **Atributos**, que são informações alfanuméricas diretamente ligadas aos conceitos
 - Exemplos: idade do cliente, data do pagamento, nome do filme

- **Associações**: um tipo de informação que liga diferentes conceitos entre si
 - Exemplos: uma pessoa “é dona de” um automóvel; uma empresa “emprega” funcionários

Como Identificar Conceitos ?

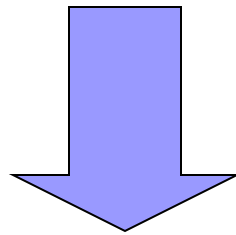
- Alguns conceitos são bastante óbvios quando se descreve o sistema, por exemplo os conceitos Leitor e Livro quando se fala em um sistema de Biblioteca

Leitor
nome tipo : char

Livro
titulo autor ano ISBN editora tipo

Como Identificar Conceitos ?

- Entretanto, muitos conceitos estão geralmente implícitos e requerem uma maior maturidade do analista para descobri-los



Regras básicas

Como Identificar Conceitos ?

- **Passo 1**: isole, no documento de requisitos ou na descrição dos casos de uso elaboradas, todos os substantivos presentes no texto

Passo 1: Isolar os substantivos

1. O Leitor chega ao balcão de atendimento da biblioteca e diz ao atendente que deseja emprestar um ou mais livros da biblioteca.
2. O Atendente seleciona a opção para adicionar um novo empréstimo.
3. O Atendente solicita ao leitor sua carteirinha, seja de estudante ou professor.
4. O Atendente informa ao sistema a identificação do leitor.
5. O Sistema exibe o nome do leitor e sua situação.
6. O Atendente solicita os livros a serem emprestados.
7. Para cada um deles, informa ao sistema o código de identificação do livro.
8. O Sistema informa a data de devolução de cada livro.

Exemplo: Biblioteca

Lista de candidatos:









Leitor	Carteirinha	Data de devolução
Balcão	Estudante	. . .
Biblioteca	Professor	
Atendente	Sistema	
Livros	Identificação de leitor	
Opção	Nome do leitor	
Empréstimo	Código de identificação do livro	

Como Identificar Conceitos ?

- **Passo 2:** para cada substantivo
 - verifique se é relacionado a assuntos importantes no domínio do sistema
 - **Descarte:**
 - os que fogem do escopo do sistema,
 - os que são sinônimos a outros conceitos já identificados
 - os que são atributos

Exemplo: Biblioteca

Lista de candidatos:

Leitor	Carteirinha 	Data de devolução 
Balcão 	Estudante	. . .
Biblioteca	Professor	
Atendente	Sistema 	
Livros	Identificação de leitor 	
Opção 	Nome do leitor 	
Empréstimo	Código de identificação do livro 	

Como Identificar Conceitos ?

■ **Passo 3:** isole os verbos

- que poderiam ser transformados em substantivos (possivelmente com a ajuda de outras palavras)
- Concentre-se nos verbos que representam ações de interesse para o sistema
 - aqueles relacionados a eventos e transações que possuem informações importantes e que devem ser lembradas pelo sistema



Exemplo: Biblioteca

1. O Leitor chega ao balcão de atendimento da biblioteca e diz ao atendente que deseja emprestar um ou mais livros da biblioteca.
2. O Atendente seleciona a opção para adicionar um novo empréstimo.
3. O Atendente solicita ao leitor sua carteirinha, seja de estudante ou professor.
4. O Atendente informa ao sistema a identificação do leitor.
5. O Sistema exibe o nome do leitor e sua situação.

. . .

Exemplo: Biblioteca

1. O Leitor chega ao balcão de atendimento da biblioteca e diz ao atendente que deseja emprestar um ou mais livros da biblioteca.
2. O Atendente seleciona a opção para adicionar um novo empréstimo.
3. O Atendente solicita ao leitor sua carteirinha, seja de estudante ou professor.
4. O Atendente informa ao sistema a identificação do leitor.
5. O Sistema exibe o nome do leitor e sua situação.

. . .

Exemplo: Biblioteca

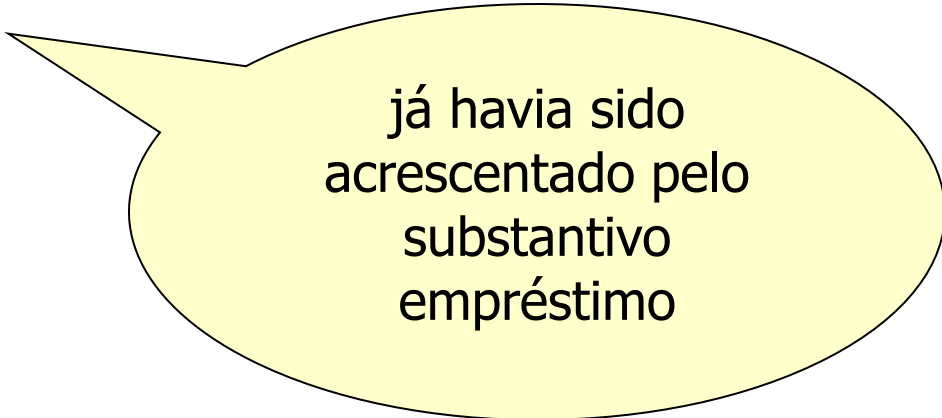
Lista de candidatos:

Emprestar x

Adicionar x

Informar x

...



já havia sido
acrescentado pelo
substantivo
empréstimo

Como Identificar Conceitos ?

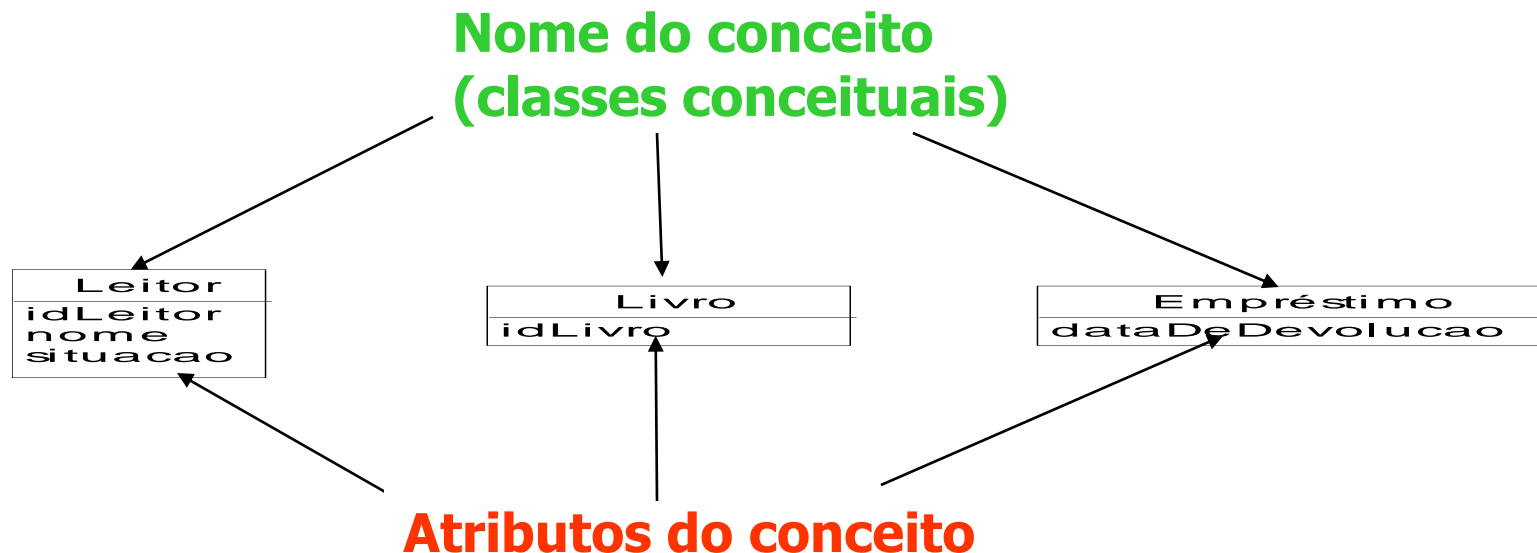
- **Passo 4:** Para cada candidato a conceito,
 - verifique se ele é composto de outras partes que sejam de interesse para o sistema, mesmo que essas não apareçam explicitamente no texto
 - Por exemplo, um empréstimo normalmente refere-se a vários livros emprestados em uma mesma ocasião para um mesmo leitor

Como Identificar Atributos?

- Substantivos sublinhados a partir do texto do caso de uso podem ser candidatos a atributos dos conceitos já identificados
- **Cautela:** não torne o modelo conceitual muito complexo desnecessariamente
 - limite-se a adicionar os atributos importantes para compreender o conceito, ou atributos que serão importantes para o futuro projeto do sistema

Como Identificar Atributos ?

■ Exemplo



Atributo ou conceito? Como saber?

- *Se observarmos que X não pode ser representado apenas como um número ou um texto no mundo real, X provavelmente é uma classe conceitual, não um atributo*
- Por exemplo, loja deve ser um atributo de venda ou uma classe conceitual separada?

Venda
loja

Ou?

Venda

Loja
N_telefone

Atributo ou classe? Como saber?

- No mundo real, uma loja não é considerada um número ou texto
 - o termo sugere uma entidade legal, uma organização e algo que ocupa espaço
- Portanto loja deve ser uma classe conceitual

Resumindo...

- Como encontrar conceitos e atributos:
 - Verificar o texto dos casos de uso expandidos (ou do documento de requisitos)
 - Selecionar termos que representam informação transmitida “do” e “para” o sistema
 - Agrupar sinônimos

Associações

- Associação é um relacionamento entre conceitos, que precisa ser lembrado pelo sistema durante seu funcionamento
- Por exemplo, existe uma associação entre Empréstimo e Leitor, que precisa ser lembrada pelo sistema, pois quando o Empréstimo termina o Leitor deve ter sua situação regularizada.

Como identificar Associações?

- Algumas associações são bastante simples de serem identificadas, bastando aplicar certas regras básicas
- Outras associações são implícitas e podem requerer mais experiência do analista
- **Cautela:** incluir associações em demasia causa um efeito indesejado, levando a um modelo confuso e, conseqüentemente, com pouca legibilidade
 - Portanto, uma regra básica é evitar colocar associações redundantes

Como identificar Associações?

- **Regra 1:** Um conceito que, fisicamente ou logicamente, faz parte de outro
 - Por exemplo, um exemplar logicamente faz parte do Empréstimo

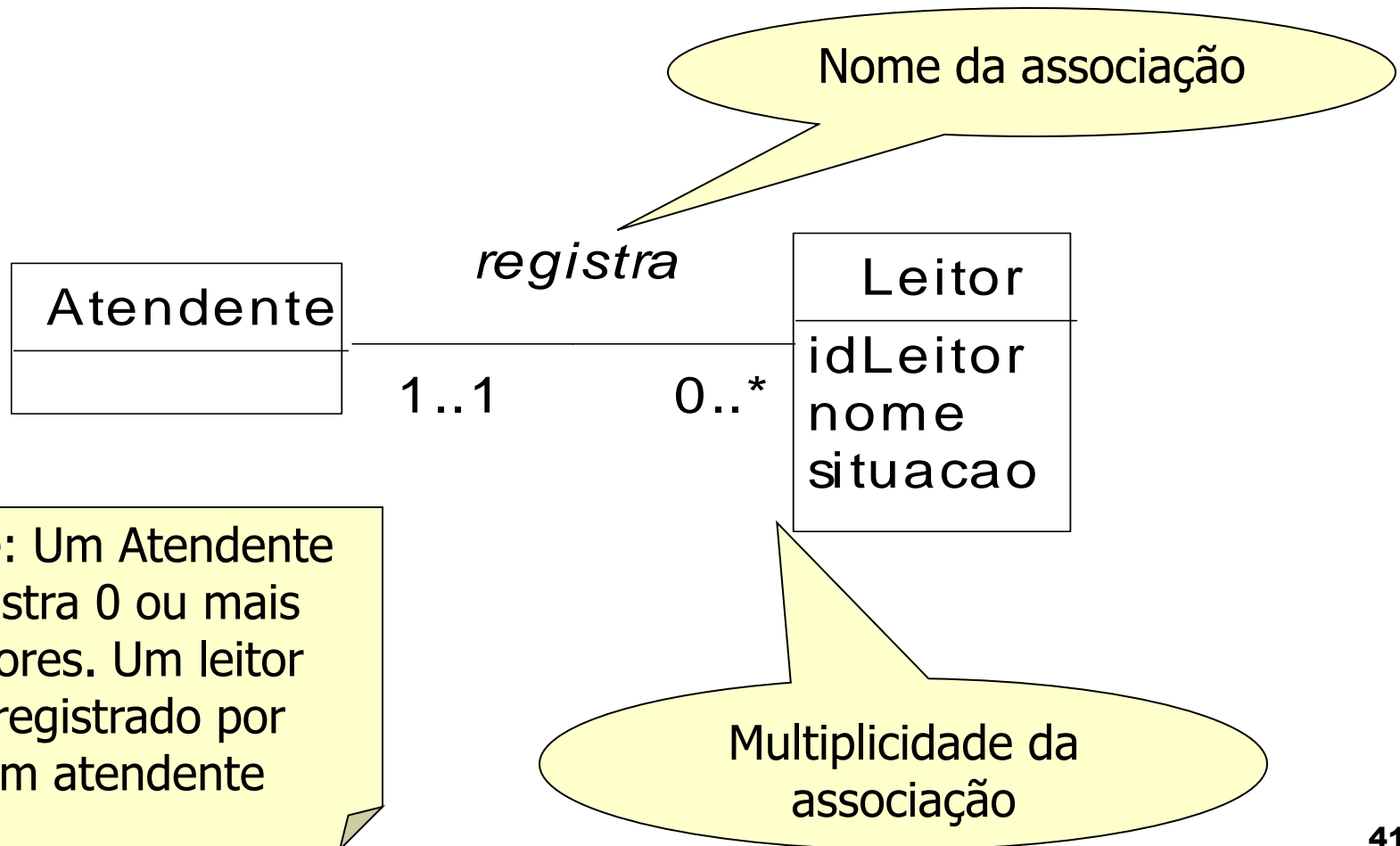
Como identificar Associações?

- **Regra 2:** Um conceito que serve para descrever ou qualificar outro conceito
 - Por exemplo, um Livro pode ser classificado em diversas Categorias ou por Autor

Como identificar Associações?

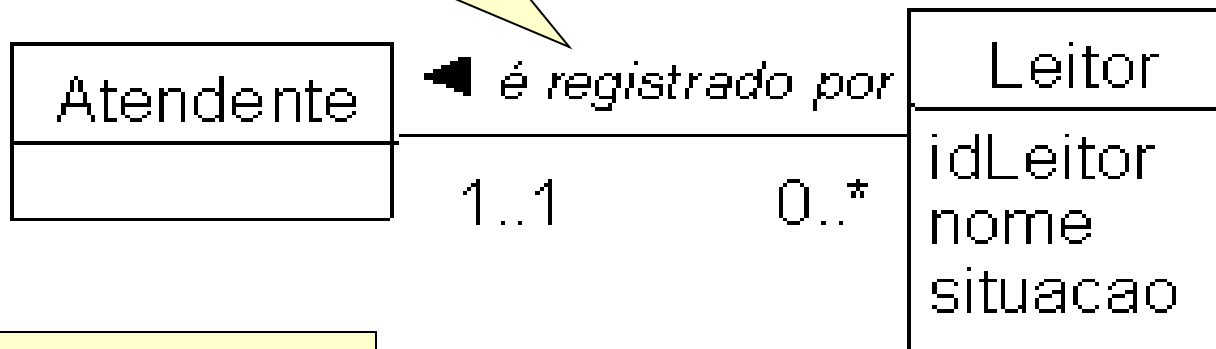
- **Regra 3:** Um conceito que é responsável por registrar ou manter informações sobre outro
 - Por exemplo, o Atendente é quem registra e atende o Leitor
 - a Bibliotecária é responsável pelos Livros

Associações na UML



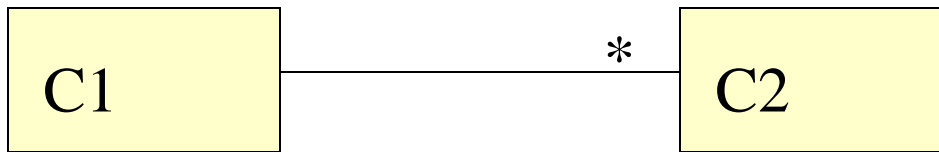
Associações na UML

Direção de leitura do
nome da associação

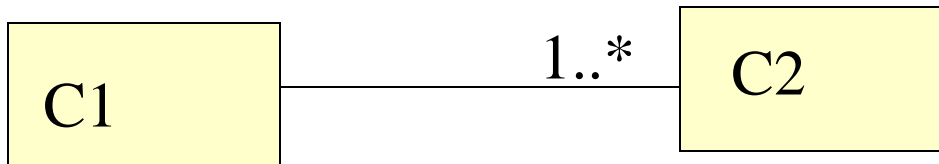


A direção só deve ser colocada se for da direita para a esquerda ou de baixo para cima.

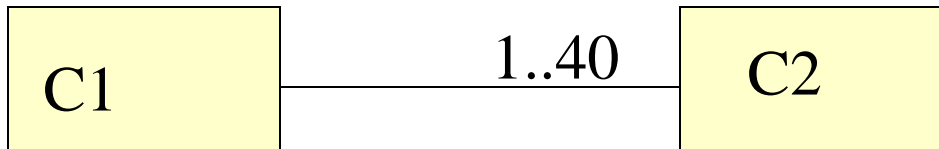
Multiplicidade



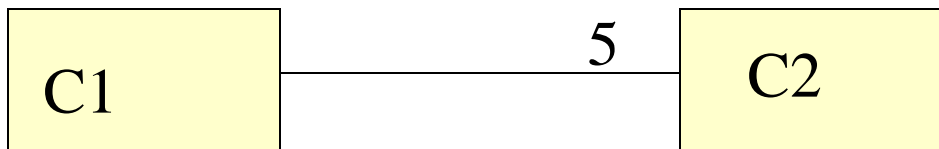
zero ou mais
Muitos (as)



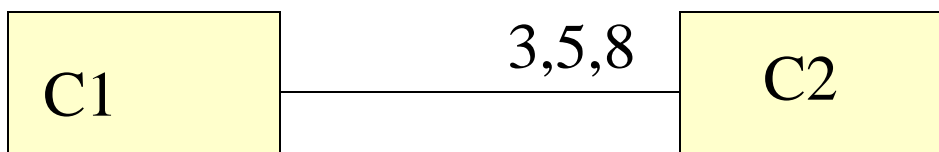
um ou mais



um a quarenta

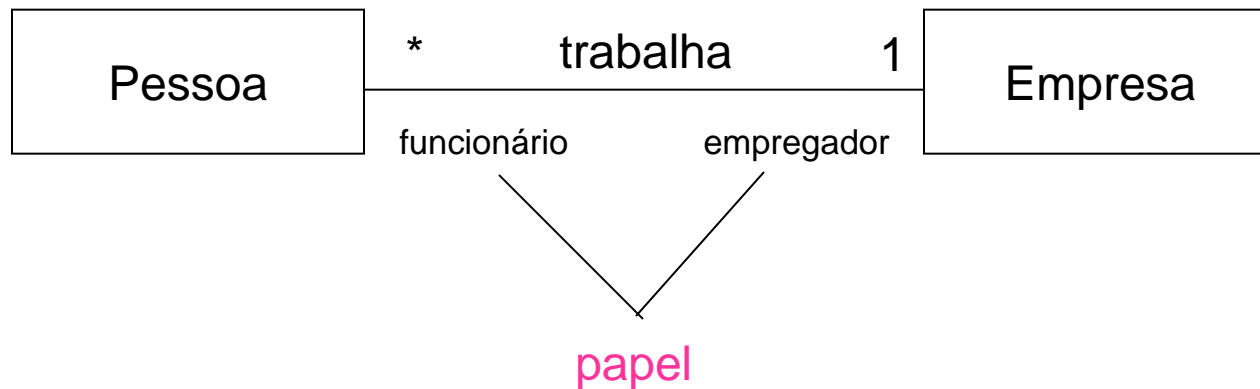


exatamente cinco



exatamente três, cinco
ou oito

Associação Reflexiva

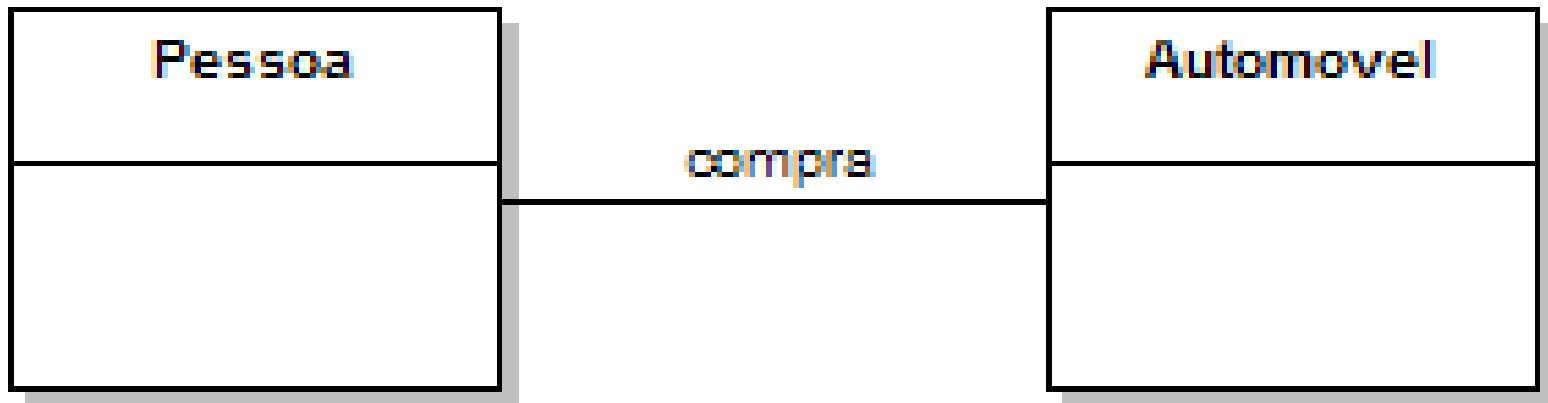


Pessoa desempenhando o papel de funcionário está associado a Empresa que desempenha o papel de empregador

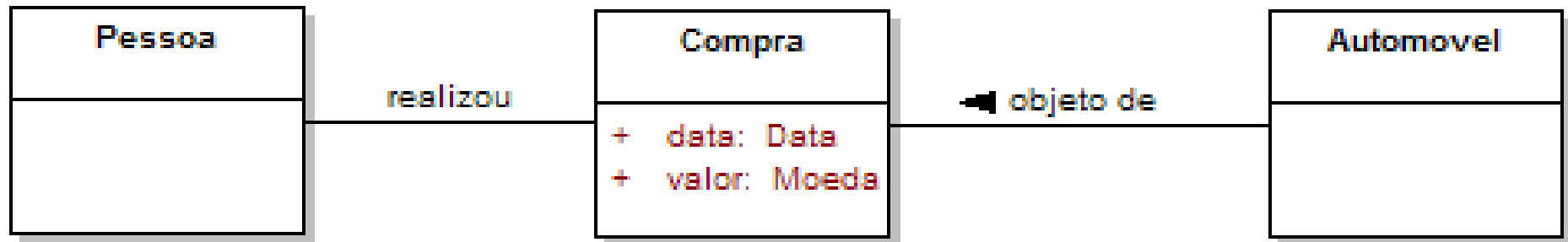
Associação Reflexiva

- Uma anotação especial é colocada juntamente com a multiplicidade, chamada de *papel*
 - que denota o papel desempenhado pelo objeto de cada lado da associação

Uma operação (transação)
não deve ser modelada
como associação



Como modelar a transação se necessário



Atributos disfarçando associações

- **Não** se deve colocar no modelo conceitual os atributos que representam “chaves estrangeiras”, como se fosse uma tabela de banco de dados relacional
 - Ex.: não colocar “nome do cliente” como atributo de “Empréstimo”

Herança

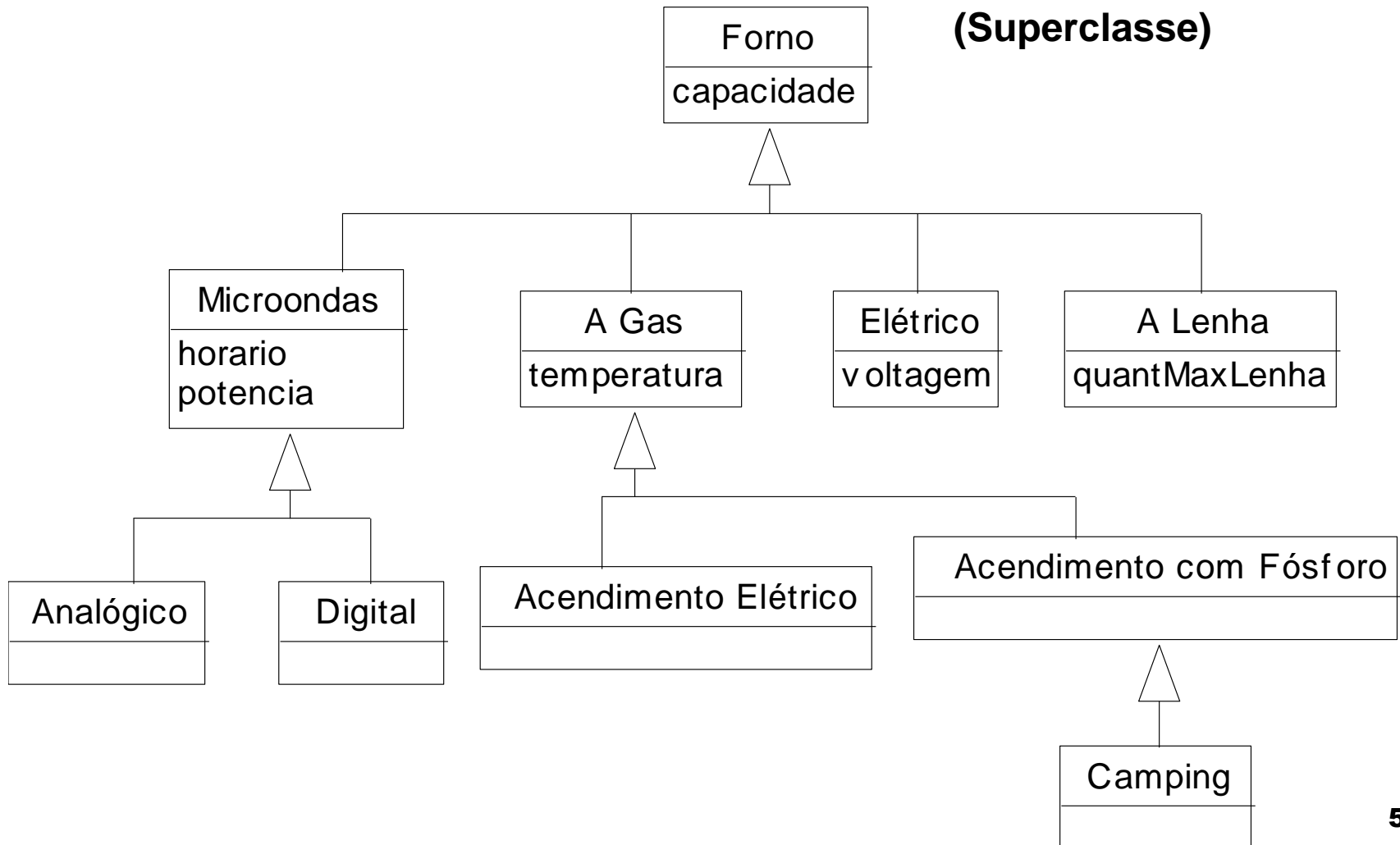
■ Herança

- é um mecanismo que permite que características comuns a diversas classes sejam colocadas em uma classe base, ou **superclasse**

Herança

- A partir de uma superclasse, outras classes podem ser especificadas (as subclasses)
- Cada subclasse apresenta as características (estrutura e métodos) da superclasse e acrescenta a elas novas características ou comportamento
- Dizemos que uma subclasse herda todas as propriedades da superclasse e acrescenta suas características próprias e exclusivas

Herança na UML



Regras da Herança

- A “regra é-um” estabelece que todos os membros do conjunto de um subtipo devem ser membros do conjunto do supertipo
 - ou seja o subtipo é um supertipo
 - Por exemplo, podemos dizer que um microondas digital é um forno, e por isso possui todos os atributos e comportamentos esperados de um forno

Regras da Herança

- A “regra dos 100%” estabelece que 100% da definição do supertipo dever ser aplicado ao subtipo
 - se a superclasse possui um dado atributo, é necessário garantir que todas as subclasses também o possuam, ou seja, ele deve fazer sentido para elas e deve ser importante conhecer seu conteúdo
 - O mesmo vale para as associações da superclasse, que devem ser aplicáveis às subclasses

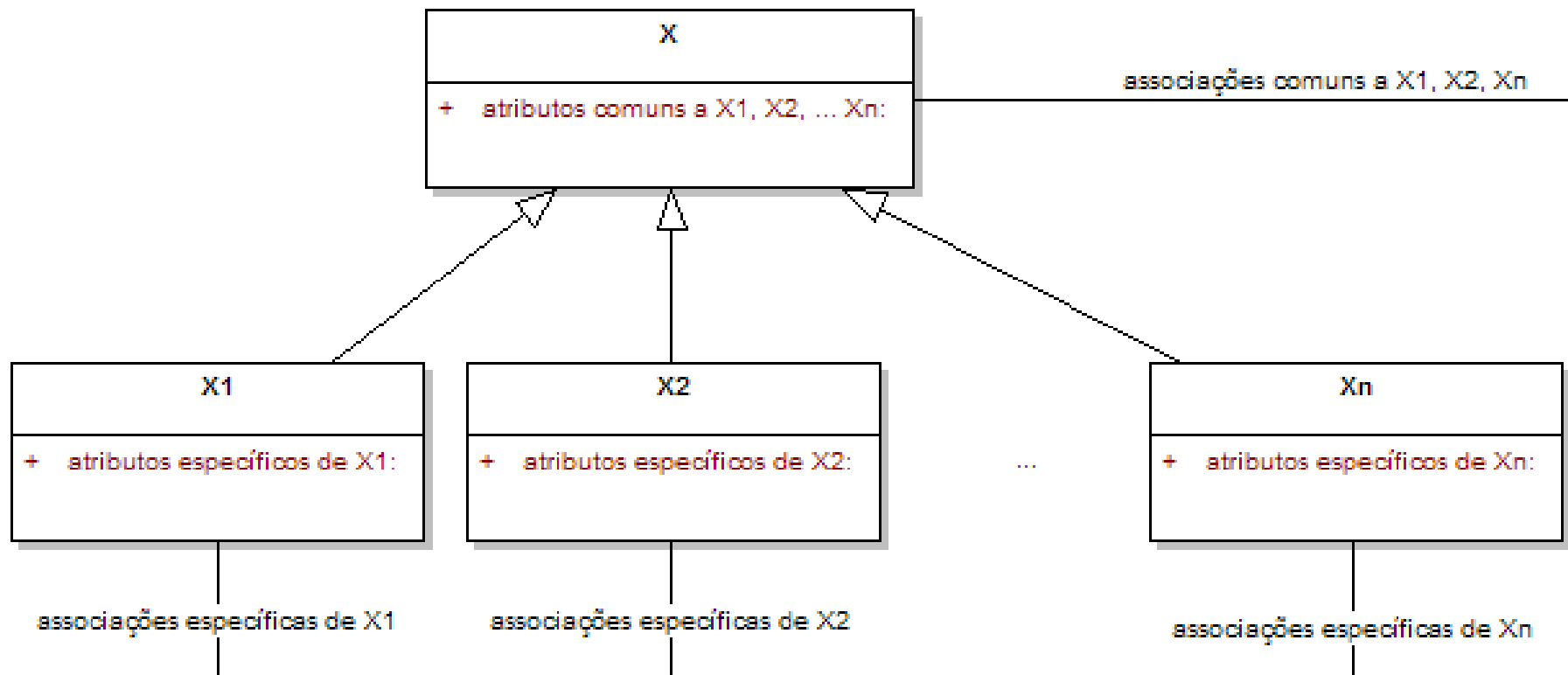
Regras da Herança

- Por exemplo

- se incluirmos a classe Cozinheiro no modelo, e associarmos essa classe à classe forno, significando que o Cozinheiro utiliza o Forno, isso deve ser verdadeiro para todas as subclasses ou seja, todos os tipos de forno devem poder ser utilizáveis por um cozinheiro

Herança: Atenção!

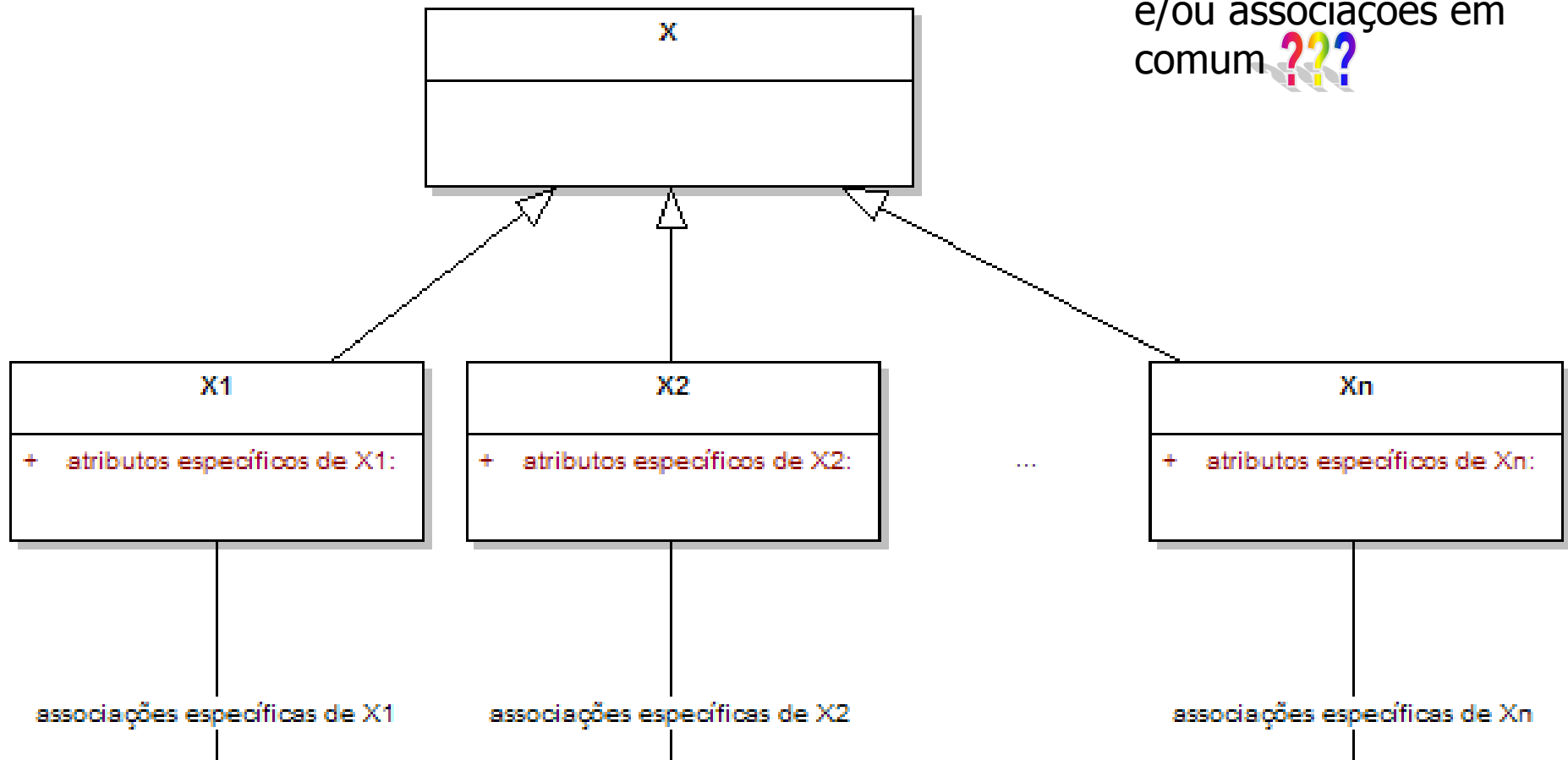
- Onde herança pode ser usada...



Herança: Atenção!

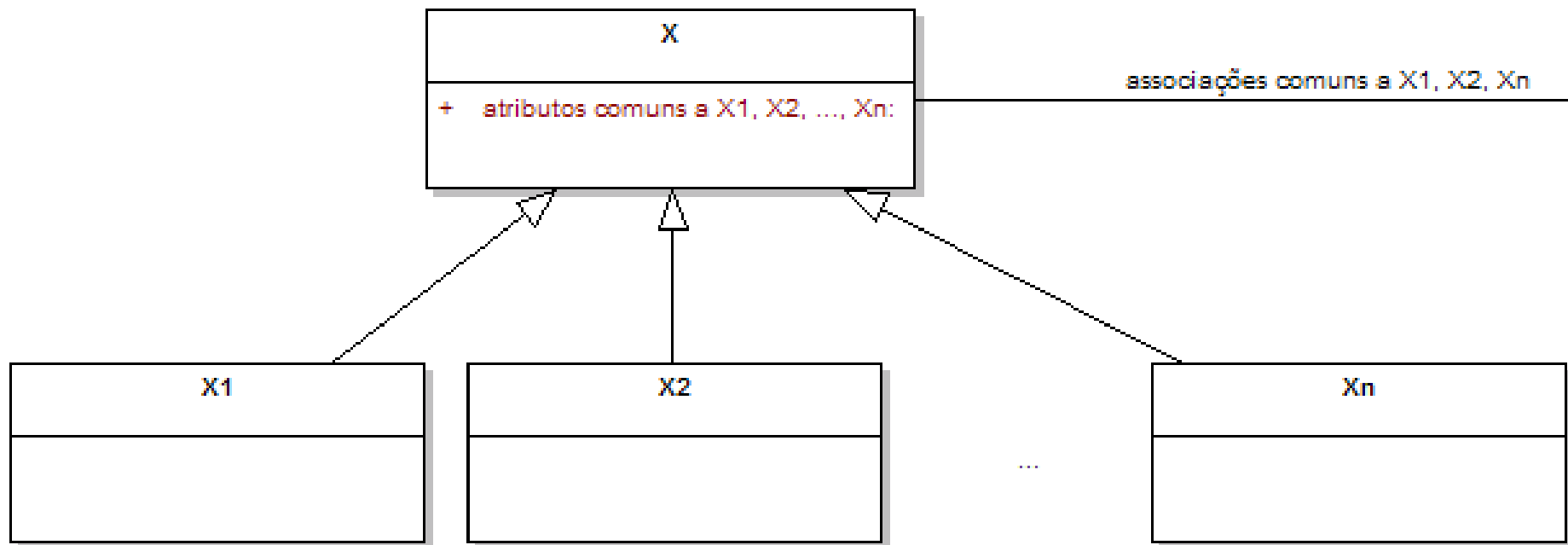
- Não use herança se...

Cadê os atributos e/ou associações em comum ???



Herança: Atenção!

- Não use herança se...



Cadê os atributos
e/ou associações
específicos ???

Agregação (ou Agregação Simples)

- é uma forma especial de associação (semântica maior)
- é um mecanismo pelo qual um objeto inclui atributos e comportamento de outros objetos a ele agregados, indicando a existência de um *todo*, composto por *partes*

Agregação na UML



Agregação

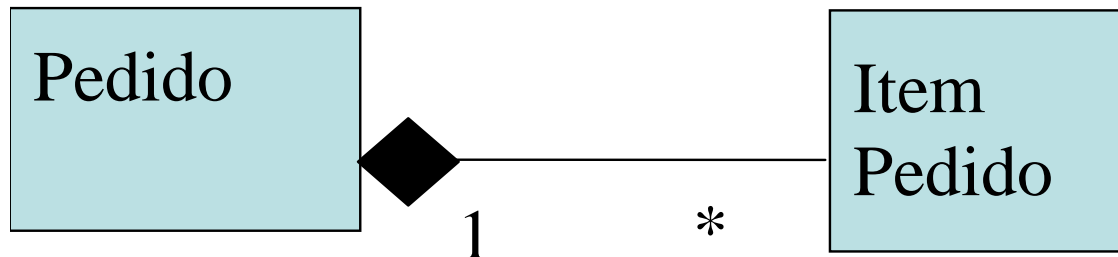
- É um conceito simples com uma semântica bem profunda
- A agregação simples é inteiramente conceitual e nada faz além de diferenciar o “todo” da parte

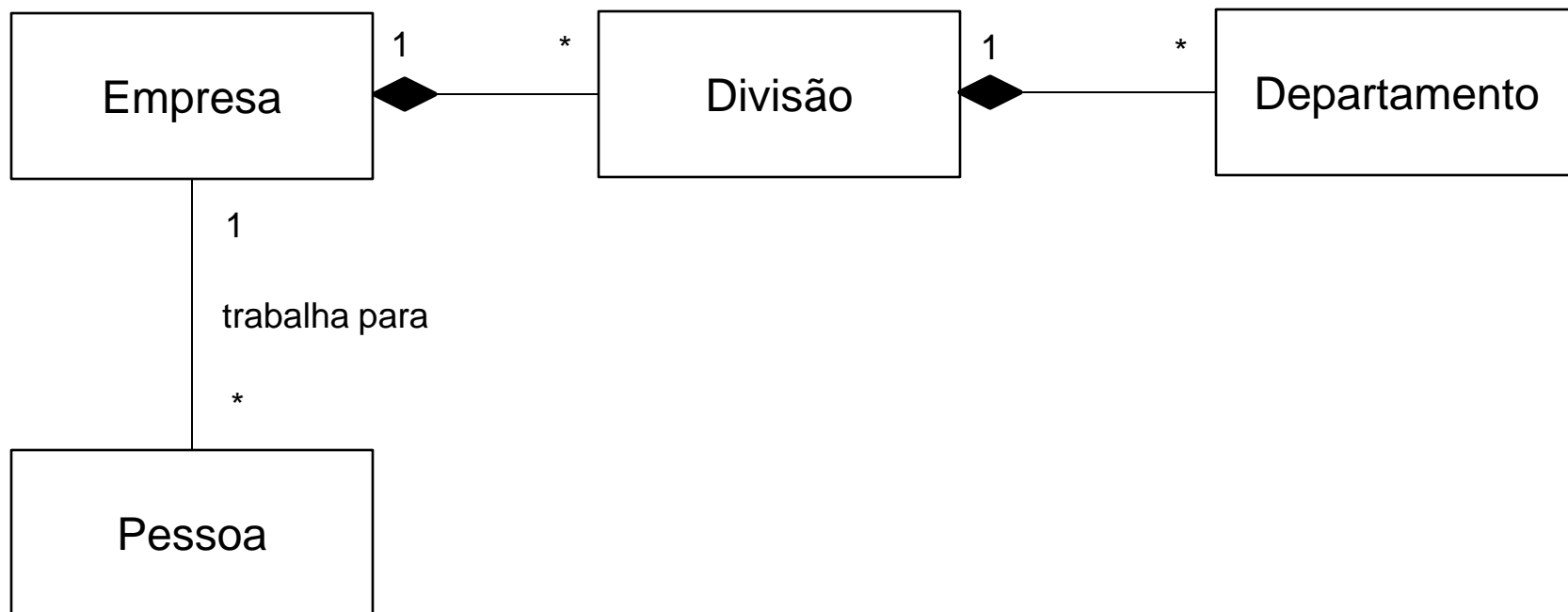
Agregação versus Associação

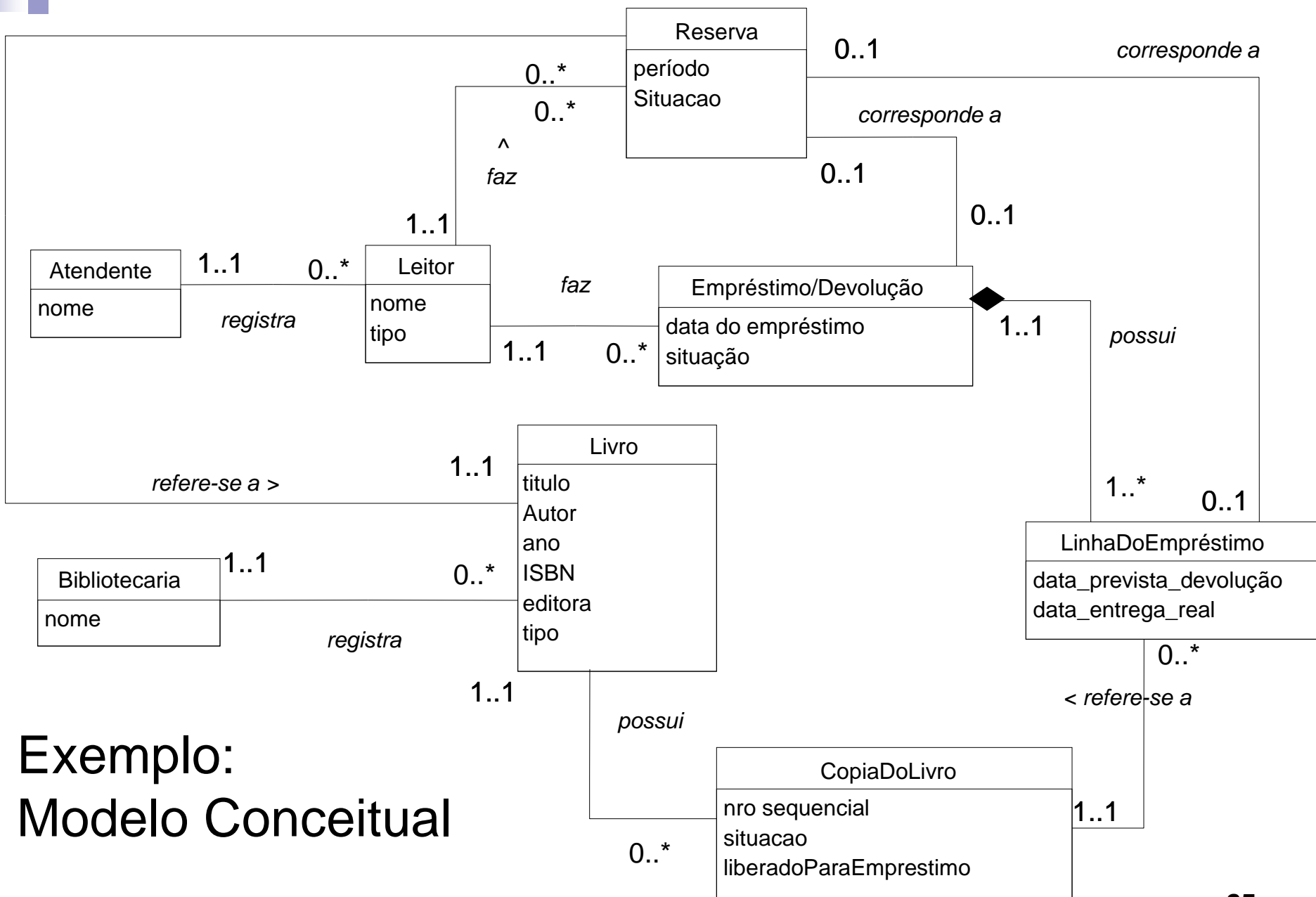
- Se dois objetos estiverem fortemente ligados por um relacionamento todo-parte, isso é uma agregação
 - Questão: O objeto é parte do todo?
- Se os dois objetos forem em geral considerados independentes, embora frequentemente possam ser vinculados, isso é uma associação

Composição

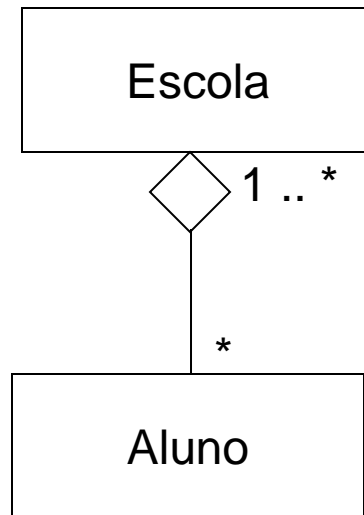
- É uma variação da agregação simples (mais restritiva)
- A composição ou agregação composta ocorre quando a multiplicidade na extremidade do “todo” pode ser no máximo 1
- O tempo de vida da “parte” é coincidente com o do “todo”







Outro exemplo:



Uma escola tem zero ou mais alunos, cada aluno poderá estar registrado em uma ou mais escolas.

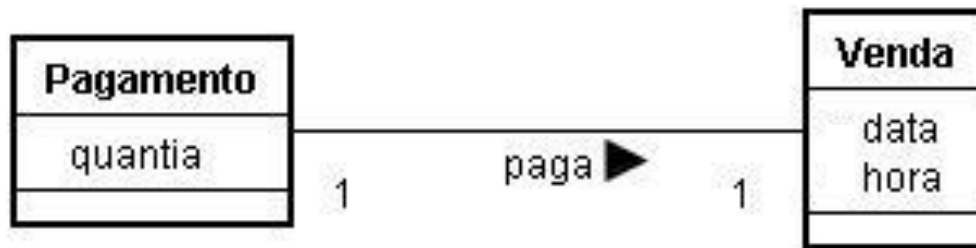
Considerações...

- No exemplo anterior, você poderia não usar a agregação e usar associações. No entanto, ao especificar que “escola” é um todo e que “aluno” é uma de suas partes, fica claro qual deles é organizacionalmente superior ao outro
- Assim, as escolas são definidas, de alguma forma, pelos seus alunos
- De forma semelhante, os alunos não existem fora da escola a que pertencem

Importante...

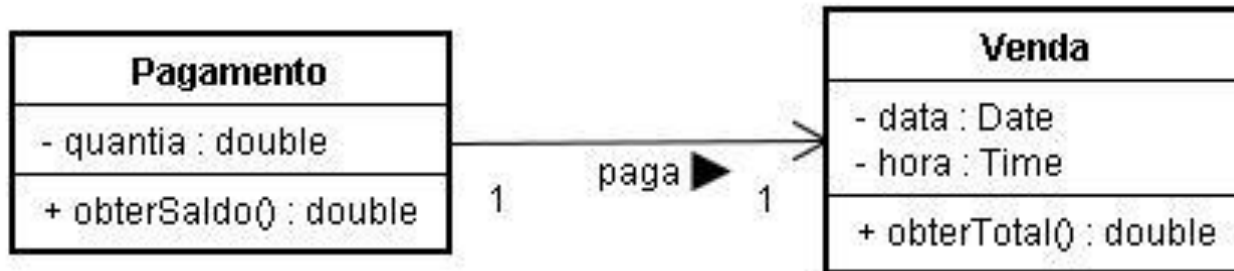
- PU utiliza o diagrama de classes para representar o “Modelo Conceitual”
 - Não há um diagrama “Modelo de Domínio” na documentação oficial da UML
- Assim, metodologias e processos existentes (como é o caso do PU) aplica os diagramas puros da UML no contexto de modelos definidos por seus autores

Importante...



Modelo Conceitual do PU

A notação pura de diagrama de classes/objetos da UML é usada em um modelo essencial visualizando conceitos do mundo real



Modelo de Projeto do PU

A notação pura de diagrama de classes da UML é usada em um modelo de implementação visualizando componentes de software