

FACULDADE DE COMPUTAÇÃO E INFORMÁTICA
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

Projeto de Software – Aula 5 – 2º SEMESTRE/2018

Análise, Projeto e Desenvolvimento de Sistemas de Software – Mais sobre Diagrama de Classes

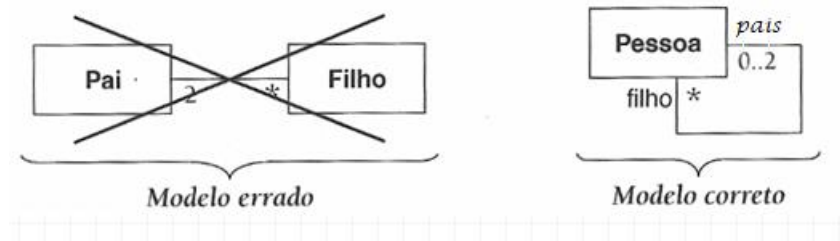
Objetivos:

- Aprender dois novos conceitos: associação reflexiva (auto-relacionamento) e classe associativa
 - Aprender sobre visibilidade e associações entre classes
 - Aprender dois novos tipos de associações: agregação e composição
 - Praticar o método para construir diagrama de classes
-

Associação Reflexiva Ou Associação Unária

A associação reflexiva ou unária é um tipo de associação que ocorre quando existe um relacionamento de uma classe para consigo mesma. No exemplo apresentado na Figura 1, a associação reflexiva representa de forma mais adequada a relação entre duas pessoas de paternidade.

Figura 1: Exemplo de associação reflexiva.

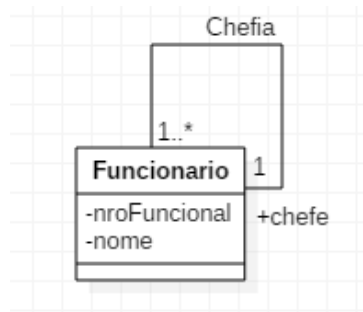


Já ao observarmos o exemplo da Figura 2, podemos perceber que a classe Funcionario apresenta como atributo o nroFuncional e o Nome, já relação de quem chefia o Funcionario é realizada através da associação reflexiva denominada Chefia.

O chefe do funcionário também é, por sua vez, um funcionário da empresa e, portanto, também constitui em uma instância da Classe Funcionário. A associação chamada Chefia indica uma possível relação entre uma ou mais instâncias da classe Funcionário com outras instâncias da própria classe Funcionário.

Quando um Funcionário Chefia um ou mais Funcionário ele desempenha o papel de Chefe.

Figura 2: Outro exemplo de associação reflexiva ou unária.

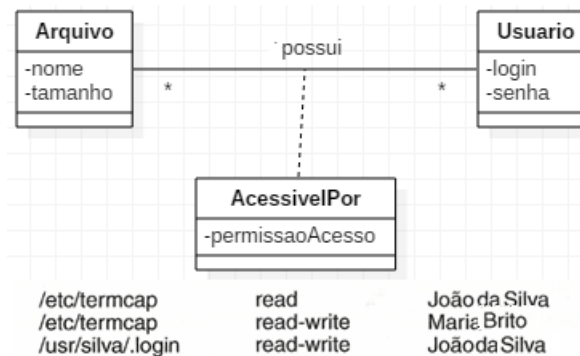


Classe Associativa

Uma classe associativa ou classe de associação é uma associação que também é uma classe. Uma classe de associação pode ter atributos e operações, e participar nas associações. Pode ser encontrada classes de associação achando advérbios em uma especificação de problema, ou abstraindo valores conhecidos.

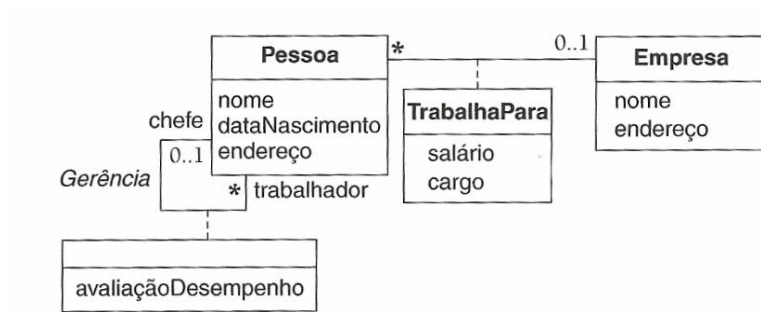
No exemplo da Figura 3, `permissaoAcesso` é um atributo de `AcessivelPor`. A notação UML, para uma classe associativa é uma caixa de classe conectada à associação por uma linha tracejada.

Figura 3: Diagrama de Classes - exemplo classe associativa.



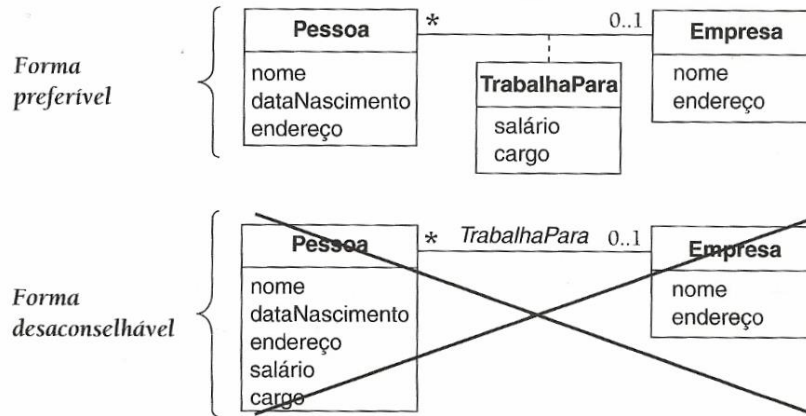
Associação muitos-para-muitos oferecem argumentos convincente para as classes de associação. A figura apresenta atributos para duas associações um-para-muitos. Cada pessoa trabalhando para uma empresa recebe um salário e possui um cargo. O chefe avalia o desempenho de cada trabalhador. Como é apresentado no exemplo da Figura 4.

Figura 4: Exemplo de Classe associativa -com associação reflexiva.



A Figura 5 mostra como é possível juntar atributos para associações um-para-um e um-para-muitos na classe oposta a uma extremidade “um”. Isso não é possível para associação muitos-para-muitos.

Figura 5: Diferença entre classe associativa e associação simples - Cuidado na troca.



Agregação e Composição

Toda associação pode ser atrelada a uma semântica. A semântica de uma associação corresponde ao seu significado, ou seja, à natureza conceitual da relação que existe entre os objetos que participam daquela associação. Como por exemplo, quando é descrito o nome a uma associação, ou também os papéis de cada classe na associação. De todos os significados diferentes que uma associação pode ter, há uma categoria especial de significados, aquelas que representam as relações **todo-parte** (BEZERRA, 2007).

Algumas associações podem ser consideradas mais fortes do que outras no sentido de que elas definem que um objeto é composto por outros. Como por exemplo, uma **casa** é composta por **cômodos**.

A UML define dois tipos de relacionamentos todo-parte, que são: agregação e a composição.

É importante ressaltar que, a agregação e a composição são casos especiais da associação, desta forma, todas as características válidas para a associação (multiplicidade, participação, papéis, etc.) são válidas para agregação e composição.

Também, deve ficar claro que, sempre que for possível utilizar agregação e composição, uma associação também pode ser utilizada.

Características particulares:

- Agregações/composições são assimétricas: se um objeto A é parte de um objeto B, B não pode ser parte de A.
- Agregações/composições propagam comportamento, no sentido de que um comportamento que se aplica a um todo automaticamente se aplica as suas partes.
- Nas agregações/composições, as partes são normalmente criadas e destruídas pelo todo. Na classe objeto todo, são definidas as operações para adicionar e remover as partes.

Regra para identificação de Agregação/Composição:

Sejam duas classes associadas, X e Y. Se uma das perguntas a seguir for respondida com um sim, provavelmente há uma agregação/composição onde X é todo e Y é parte.

1. X tem um ou mais Y?
2. Y é parte de X?

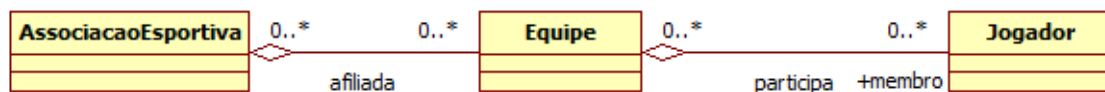
Agregação

A agregação é uma forma poderosa de associação em que um objeto agregado é composto de partes constituintes, e assim, as constituintes são partes do agregado. O objeto agregado é semanticamente um objeto estendido, tratado como uma unidade em muitas operações, embora fisicamente ele seja composto de várias unidades.

Graficamente, a UML fornece uma notação para representação da agregação. A agregação é representada como uma linha que conecta as classes relacionadas, com um diamante (losango) branco perto da classe que representa o todo.

Veja o exemplo a seguir extraído de Bezerra (2007):

Esse diagrama representa uma associação esportiva que é formada por diversas equipes, sendo que cada equipe é formada por diversos jogadores, e por outro lado, um jogador pode fazer parte de diversas equipes.



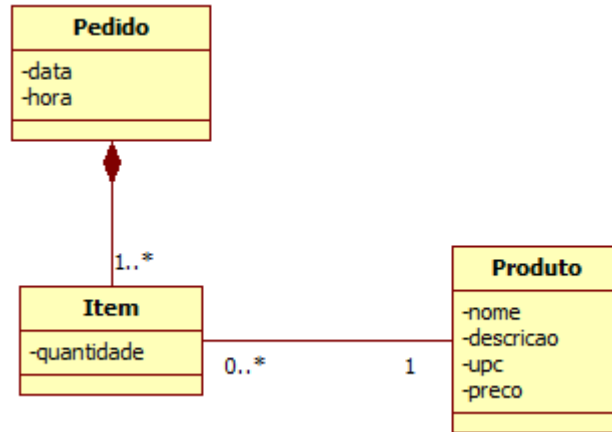
Composição

A composição é uma forma de agregação com duas restrições adicionais. Uma parte constituinte pode pertencer a no máximo uma montagem. E, além disso, quando uma parte constituinte tiver sido atribuída a uma montagem, ela terá o tempo de vida coincidente com a montagem.

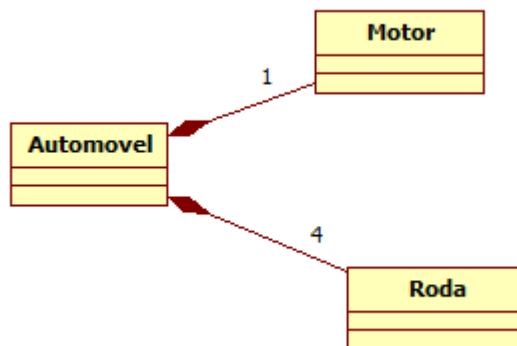
Graficamente, a UML fornece uma notação para representação da composição. A composição é representada como uma linha que conecta as classes relacionadas, com um diamante (losango)

negro (ou preenchido – para contrapor com o diamante branco da agregação.) perto da classe que representa o todo.

Veja este exemplo de composição extraído de Bezerra (2007). Considere que os itens de um pedido de compra, assim, é comum um pedido de compra incluir vários itens. Cada item diz respeito a um produto faturado. Os itens têm identidade própria (é possível distinguir um item de outro no mesmo pedido). Observe o diagrama UML a seguir.



Ou, este outro exemplo, um automóvel possui várias partes, a saber, um motor e quatro rodas.



Como construir um Diagrama de Classes

Procedimento Simplificado para a Construção do Diagrama de Classes

1. Leia atentamente a descrição do problema (ou descrição do cenário do problema)
2. Identificar os candidatos a classes, para isso, grife os substantivos na descrição do problema;
 - Os substantivos podem ser geralmente, candidatos a classes;
3. Faça uma lista dos substantivos, esses substantivos são candidatos a classes.
4. Analise a lista de classes candidatas e elimine os redundantes, similares ou não relevantes

- Agrupe as classes que descrevem o mesmo conceito (sinônimos e similares).
Deixe o que descreve melhor o conceito;
 - Eliminar classes não relevantes (por exemplo, substantivos que aparecem no texto, mas não está no escopo do sistema);
 - Especificar melhor as classes vagas;
 - Eliminar as construções não relacionadas com o mundo real; as construções relacionadas com a implementação devem ser utilizadas na fase de projeto (por exemplo, processador, subrotina, algoritmo, interrupção).
5. Preparar o dicionário de classe, ou seja, descreva brevemente o significado de cada uma da lista.
 6. Identificar e nomear as associações entre as classes, ou seja, identifique as classes que conceitualmente tem uma relação no mundo real. Nomeie a associação com um verbo que descreva a relação entre as classes.
 7. Para cada classe identifique as propriedades que são relevantes no contexto do cenário descrito, ou seja, identifique as informações que são relevantes e que descrevem (que a qualificam) a classe no contexto do sistema.
 8. Analise cada relacionamento entre as classes e verifique a quantidade de ocorrências que um objeto das classes pode ter de outro objeto das classes na outra extremidade da associação. Faça o mesmo procedimento na outra extremidade.
 9. Identificar os relacionamentos de agregação/composição. Utilize as regras da semântica deste tipo de associação.
 10. Refinar o modelo.

Referências bibliográficas

- BLAHA, M., RUMBAUGH, J. Modelagem e projetos baseados em objetos com UML 2. Rio de Janeiro:Elsevier-Campus, 2006.
- BOOCH, G. Object-oriented analysis and design with applications. 3ª.ed. Addison-Wesley, 2007.