

Introdução às Classes, Associações e Generalizações

O diagrama de classes é um tipo de diagrama de estrutura que serve para representar, visualmente, as classes (e suas relações) de um sistema. Esse diagrama é um dos mais utilizados (se não for o mais utilizado) diagramas da UML. Ele auxilia tanto a construção da aplicação como a [construção do banco de dados](#).

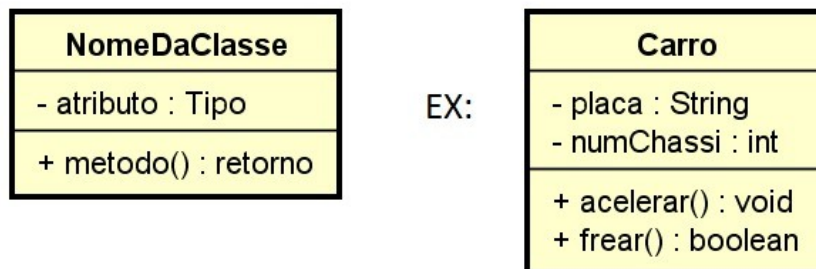
Modelando uma classe no Diagrama

Uma classe neste diagrama é modelada da seguinte forma:

1º compartimento: Nome da Classe – Identidade única

2º compartimento: Atributos – Estado

3º compartimento: Métodos – Comportamentos



A visibilidade dos atributos e métodos no sistema é representado no diagrama da seguinte forma:

- Pública (+): qualquer classe pode acessa-lo;
- Privada (-): só a classe que o possui pode acessa-lo;
- Protegida (#): somente as classes e subclasses podem acessa-lo
- Pacote (~): todas as classes que pertencem ao pacote podem acessa-lo.

Modelando relacionamentos no diagrama

Associação:

A associação representa a relação existente entre objetos. No diagrama, sua sintaxe é uma linha que liga as duas classes que representam os objetos relacionados.

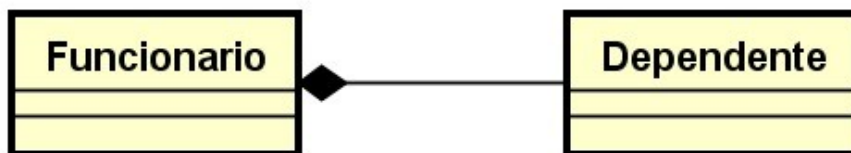


Casos especiais de Associação

Agregação: representa uma relação todo-parte entre o agregado e suas partes. Nessa relação, a perda do todo não resulta, obrigatoriamente, na perda das partes. Sua sintaxe é uma linha com um losango branco na classe que corresponde ao todo.

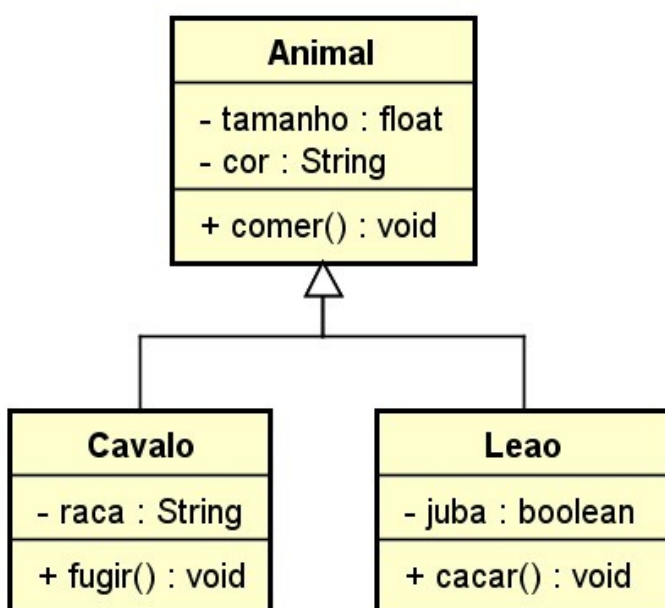


Composição: também representa uma relação todo-parte entre o agregado e suas partes. Porém a composição representa um forte relacionamento, em que as partes não sobrevivem sem o todo. Sua sintaxe é a de uma linha com um losango preto na classe que corresponde ao todo.



Generalização / Especialização:

Generalização ou Especialização representa a relação existente entre classes. Este relacionamento também é conhecido na orientação a objetos como herança. É quando existe a necessidade de criar uma classe que herde as propriedades de outra classe, isto é, os atributos, métodos e também relacionamentos (associações) da classe superior.



Nesse exemplo houve a necessidade de criar uma classe mais geral, porque todos os animais possuem tamanho e cor e se alimentam de alguma forma. Por outro lado, houve a necessidade

de especializar alguns animais por possuírem características únicas. Animal é uma generalização de Leão e Cavalo e Leão e Cavalo são especializações de Animal. Portanto, generalização e especialização são a mesma coisa, vistas de perspectivas diferentes.

A generalização no diagrama de classes da UML possui como sintaxe uma seta branca e fechada com direção das classes específicas para a mais geral, ou seja, das subclasses para as superclasses.

Conclusão

Neste artigo foi abordado a notação de classes e relacionamentos no diagrama da UML. O próximo passo é saber o que são e como são representados os papéis e multiplicidades de uma Associação, as classes associativas, as classes reflexivas e as associações ternárias.

Para gerar as imagens dos exemplos acima, utilizamos uma excelente ferramenta para modelagem que é o Astah. Esta ferramenta está disponível gratuitamente na [versão Community](#) e para estudantes na [versão profissional](#).

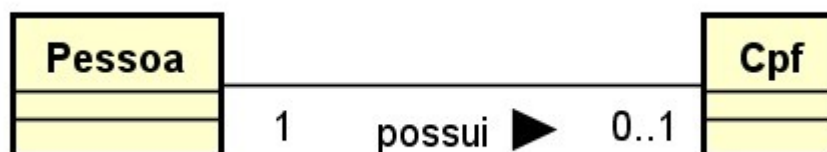
Compreendendo multiplicidade e os tipos de Associação

Entendendo o conceito de Multiplicidade

A multiplicidade nada mais é do que os limites inferior e superior da quantidade de objetos aos quais outro objeto está associado. Esses limites podem receber os valores: apenas um; zero ou muitos; um ou muitos; zero ou um; ou até um intervalo específico. Segue a tabela abaixo com a notação de cada limite no diagrama.

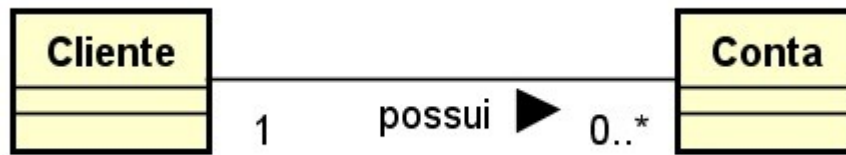
Na uml, as associações permitem representar essa informação nos extremos da sua linha.

Nos relacionamentos, os limites superiores (ou máximos) representam a sua multiplicidade. Assim, cabe as nomenclaturas para os casos a seguir:



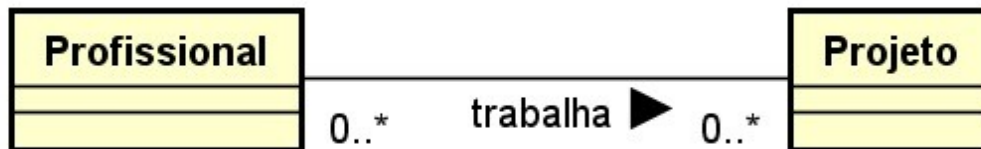
► Um para Um

Nesse primeiro exemplo, uma pessoa pode possuir 0 ou no máximo 1 cpf e um Cpf pode ser tido por somente 1 pessoa, portanto nesse caso o relacionamento é de Um para Um (observando os máximos).



► Um para Muitos

Nesse segundo exemplo um cliente pode possuir zero ou muitas contas e a Conta pode pertencer a somente 1 pessoa, portanto nesse caso o relacionamento é de Um para Muitos (observe os máximos).



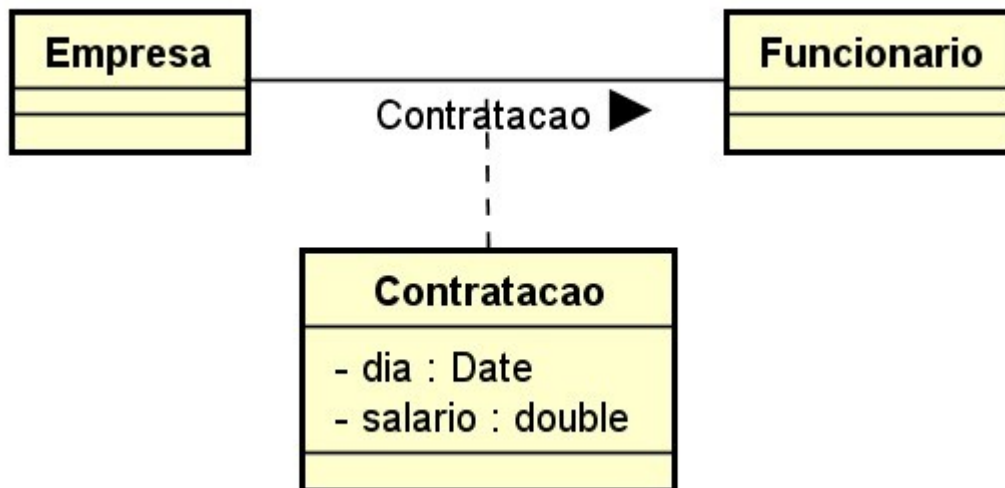
► Muitos para Muitos

Nesse último exemplo temos um Profissional que pode trabalhar em zero ou muitos projetos e um Projeto que pode ter zero ou muitos Profissionais trabalhando, portanto nesse caso o relacionamento é de Muitos para Muitos.

Compreendendo as Classes Associativas

Suponhamos a situação de contratação de um funcionário por uma empresa. Existiriam duas classes, a classe “Empresa” e a classe “Funcionário”. Entre essas classes existiria um relacionamento chamado, por exemplo, “contrata”. Suponhamos também que seja interessante para o sistema guardar as informações do dia da contratação e do salário acordado. Onde essas informações ficariam? No funcionário? Na empresa? O mais correto, no diagrama da UML, é que se crie uma classe para guardar as informações do relacionamento, da

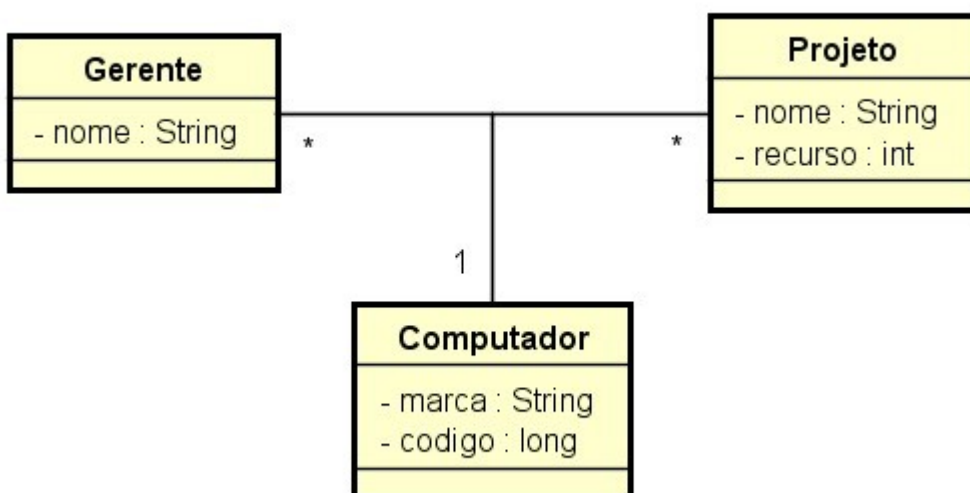
associação. Essa classe é chamada Classe Associativa ou classe de associação. Veja no exemplo abaixo.



As classes associativas no diagrama da UML são representadas igualmente as classes comuns, assim, podem se relacionar livremente com outras classes. O que as diferenciam é somente uma linha, que é tracejada, ligada ao relacionamento que a mesma representa.

Entendendo os relacionamentos Ternários

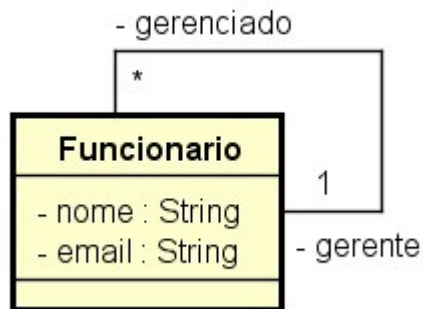
Associação ternária, quaternária ou n-ária é uma nomenclatura que se dá ao grau de uma associação. Quando binária (mais comum), a relação acontece entre somente duas classes. Quando ternária, acontece entre três. Quando quaternária entre quatro e quando n-ária (menos comum) de cinco em diante.



No exemplo acima, o relacionamento ternário está representando que o gerente pode trabalhar em muitos projetos, o projeto pode possuir muitos gerentes trabalhando, e cada gerente que trabalha em um projeto irá utilizar um computador.

Compreendendo o Auto relacionamento / Relacionamento Reflexivo

As associações reflexivas, também conhecidas como auto associações, relacionam objetos da mesma classe. Esse tipo de associação indica que um objeto de uma classe se associa a outros objetos dessa mesma classe (não a ele mesmo). Vejamos no exemplo que segue.



Nesse exemplo, não houve a necessidade de criar uma classe gerente, pois gerente também é um funcionário. Portanto, o auto relacionamento sugere que um funcionário (gerente) gerencia outros funcionários, ou seja, um relacionamento entre objetos da mesma classe.

No exemplo, os nomes gerente e gerenciados, que aparecem em cima da linha de associação, são denominados papéis. Eles servem justamente para informar quais são as funções dos objetos das classes associadas.

Conclusão

Nesse post vimos os conceitos de multiplicidade, classes associativas, associações ternárias e associações Reflexivas. Com isso damos fim aos conceitos base do Diagrama de Classes da UML.