

Análise e Projeto Orientado a Objetos

Modelagem Estrutural

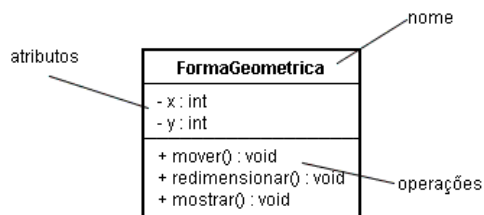
Anderson Belgamo

Linguagem UML

Modelagem Estrutural

Classes

- **Definição:** uma classe é uma descrição de um conjunto de objetos que compartilham os mesmos atributos, operações, relacionamentos e semântica.
- **Representação Gráfica:**



Classes – Termos e Conceitos

- **Nomes:**
 - O nome de uma classe deve diferenciar do nome de outras classes.
 - O nome é uma sequência de caracteres, tipicamente iniciando com maiúsculo e sem caracteres especiais.
 - Sozinho, é conhecido como nome simples.
 - O nome do caminho é o nome da classe, tendo como prefixo o nome do pacote a que essa classe pertence.

Cliente

nomes simples

SensorTemperatura

RegrasNegocio:Impostos

nomes de caminho

java:awt:Rectangle

Classes – Termos e Conceitos

■ Organização de atributos e operações:

- Pode haver a omissão dos atributos e/ou operações já que no início não tem-se a compreensão de todas essas características.

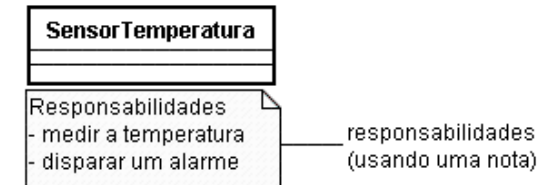
■ Responsabilidades:

- Uma responsabilidade é um contrato ou obrigações de uma determinada classe.
- Exemplo: a classe SensorTemperatura é responsável por medir a temperatura e disparar um alarme, caso a temperatura alcance determinado ponto.
 - As responsabilidades são traduzidas em um conjunto de atributos e operações, capazes de atender da melhor maneira às responsabilidades da classe.

Classes – Termos e Conceitos

■ Responsabilidades:

- As responsabilidades podem ser representadas graficamente em um compartimento separado, no final do ícone da classe.



Técnicas Básicas de Modelagem

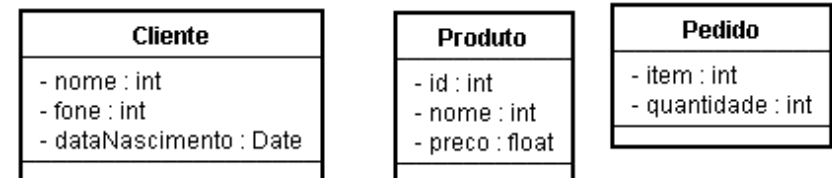
■ Modelagem do vocabulário do sistema

- Vocabulário do sistema são as abstrações definidas a partir do problema que você está tentando solucionar ou a partir da tecnologia empregada.
- Para fazer a modelagem do vocabulário de um sistema:
 - Identifique os itens que os usuários ou os implementadores usam para descrever o problema ou a solução.
 - Para cada uma das abstrações, identifique um conjunto de responsabilidades. Verifique se todas as classes estão bem-definidas e se existe um bom equilíbrio de responsabilidades entre as classes.
 - Forneça os atributos e operações necessárias para a execução dessas responsabilidades para cada uma das classes.

Técnicas Básicas de Modelagem

■ Modelagem do vocabulário do sistema

- Exemplo de sistema de vendas de varejo.



Técnicas Básicas de Modelagem

■ Modelagem da distribuição de responsabilidades em um sistema

- O objetivo é ter certeza de que suas abstrações proporcionam um conjunto equilibrado de responsabilidades.
 - Se você criar abstrações de classes muito grandes, logo descobrirá que será difícil modificar ou reutilizar seus modelos.
 - Se as suas abstrações de classes forem muito pequenas, acabará havendo um número muito maior de abstrações do que você será capaz de gerenciar ou compreender de modo razoável.

Técnicas Básicas de Modelagem

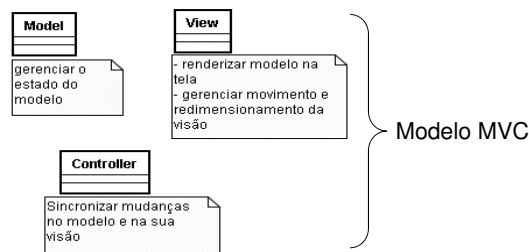
■ Modelagem da distribuição de responsabilidades em um sistema

- Para fazer a modelagem da distribuição de responsabilidades em um sistema:
 - Identifique um conjunto de classes que trabalham em conjunto com a finalidade de executar algum comportamento.
 - Identifique um conjunto de responsabilidades para cada uma dessas classes.
 - Considere esse conjunto de classes como um todo, divida em abstrações menores as classes que tenham muitas responsabilidades, reúna classes pequenas, com responsabilidades triviais, em classes maiores e redistribua as responsabilidades de forma que todas as abstrações permaneçam razoavelmente independentes.

Técnicas Básicas de Modelagem

■ Modelagem da distribuição de responsabilidades em um sistema

- Para fazer a modelagem da distribuição de responsabilidades em um sistema:
 - Leve em consideração os modos como essas classes colaboram uma com as outras e redistribua suas responsabilidades para que, em uma colaboração, nenhuma classe tenha muito ou pouco a fazer.



Técnicas Básicas de Modelagem

■ Dicas e Sugestões

- Uma classe bem-estruturada:
 - Proporciona uma abstração clara de algo definido a partir do vocabulário do domínio do problema ou do domínio da solução.
 - Incorpora um conjunto pequeno e bem-definido de responsabilidades e é capaz de executá-las todas de modo eficiente.
 - Fornece uma nítida separação da especificação da abstração e sua implementação.
 - É compreensível e simples, além de extensível e adaptável.

Técnicas Básicas de Modelagem

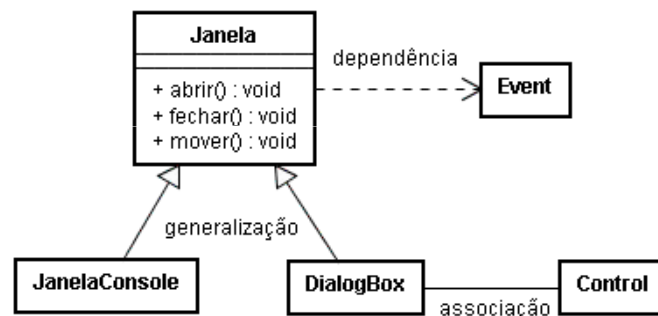
■ Dicas e Sugestões

- Ao definir uma classe na UML:
 - Mostre somente aquelas propriedades da classe que são importantes para a compreensão da abstração em seu contexto.
 - Organize extensas listas de atributos e operações, agrupando-os de acordo com suas categorias.
 - Exiba, em um mesmo diagrama de classes, classes que sejam relacionadas.

Relacionamentos

- Poucas classes trabalham sozinhas, a maioria das classes colaboram uma com as outras por meio de relacionamentos.
- Tipos de relacionamentos:
 - Dependências (relacionamentos de utilização): os canos dependem do aquecedor para fornecerem água quente.
 - Generalizações (conectam classes generalizadas a outras mais especializadas)
 - Associações (relacionamentos estruturais entre instâncias): as salas são formadas por paredes e outros itens; as próprias paredes podem ter portas e janelas embutidas.

Relacionamentos

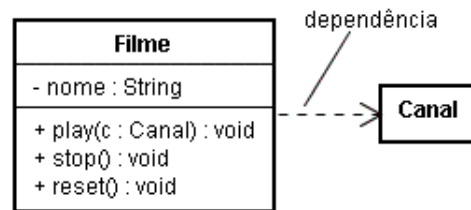


Relacionamentos – Dependência

- Uma dependência é um relacionamento de utilização, determinando as modificações na especificação de um item.
 - Exemplo: a classe Event pode afetar outro item que a utilize (por exemplo, a classe Window), mas não necessariamente o inverso.
- A dependência é representada graficamente como linhas tracejadas apontando o item do qual o outro depende.

Relacionamentos – Dependência

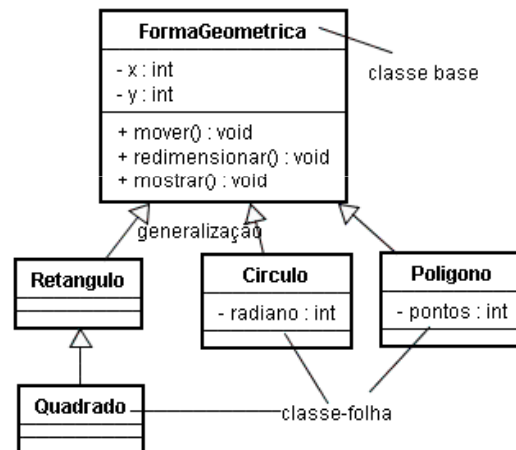
- Com muita frequência você usará as dependências no contexto das classes para mostrar que uma classe usa outra como argumento na assinatura de uma operação.



Relacionamentos – Generalização

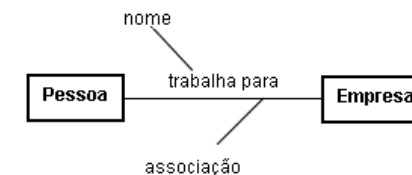
- Uma generalização é um relacionamento entre itens gerais (chamados super-classes ou classes-mãe) e tipos mais específicos desses itens (chamados subclasses ou classes-filha).
 - Também chamado de relacionamento "é um tipo de".
 - Exemplo JanelaOval é um tipo de Janela.
- A filha herda as propriedades da mãe, ou seja, atributos e operações.
- A operação de uma filha, que tenha a mesma assinatura de uma operação da mãe, prevalecerá em relação à operação da mãe (**polimorfismo**).

Relacionamentos – Generalização



Relacionamentos – Aprimoramentos

- **Nome:** uma associação pode ter um nome, que pode ser utilizado para descrever a natureza do relacionamento.



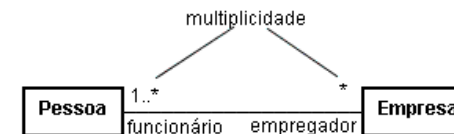
Relacionamentos – Aprimoramentos

- **Papel:** quando uma classe participa de uma associação, ela tem um papel específico a executar nesse relacionamento; o papel é apenas a face que a classe próxima a uma das extremidades apresenta à classe encontrada na outra extremidade da associação.
 - Exemplo: Pessoa, desempenhando o papel de funcionário está associado a Empresa que desempenha o papel de empregador.



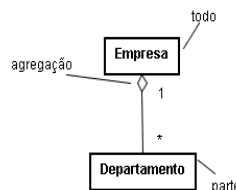
Relacionamentos – Aprimoramentos

- **Multiplicidade:** em muitas situações, é importante determinar a quantidade de objetos que podem ser conectados pela instância de uma associação.
 - Essa quantidade é chamada de multiplicidade do papel de uma associação e é escrita como uma expressão equivalente a um intervalo de valores ou a um valor explícito.



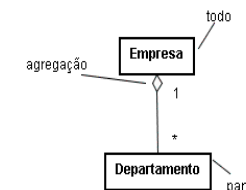
Relacionamentos – Aprimoramentos

- **Agregação:** uma pura associação (um tipos especial de associação) entre duas classes representa um relacionamento estrutural entre pares, significando que essas duas classes estão conceitualmente em um mesmo nível, sem que uma seja mais importante do que a outra.
 - Também chamado de relacionamento “tem-um” ou “todo-parte”.



Relacionamentos – Aprimoramentos

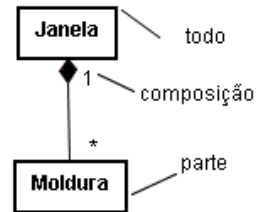
- **Agregação:** uma pura associação (um tipos especial de associação) entre duas classes representa um relacionamento estrutural entre pares, significando que essas duas classes estão conceitualmente em um mesmo nível, sem que uma seja mais importante do que a outra.
 - Também chamado de relacionamento “tem-um” ou “todo-parte”.



Relacionamentos – Aprimoramentos

■ Agregação:

- A agregação vista até o momento é inteiramente conceitual e nada faz além de diferenciar o “todo” da “parte”. Ela não modifica o significado da navegação pela associação entre o todo e suas partes, nem vincula o tempo de vida do todo e suas partes.
- Já a **COMPOSIÇÃO** adiciona alguma semântica importante, com tempo de vida coincidente como parte do todo.
 - As partes sem multiplicidade fixada podem ser criadas após a composição, mas, uma vez criadas, vivem e morrem com ela.



Relacionamentos – Aprimoramentos

■ Classes de Associação:

- Em uma associação entre duas classes, a própria associação poderá ter propriedades. Por exemplo:

