



Design Patterns

(Padrões de Projeto)

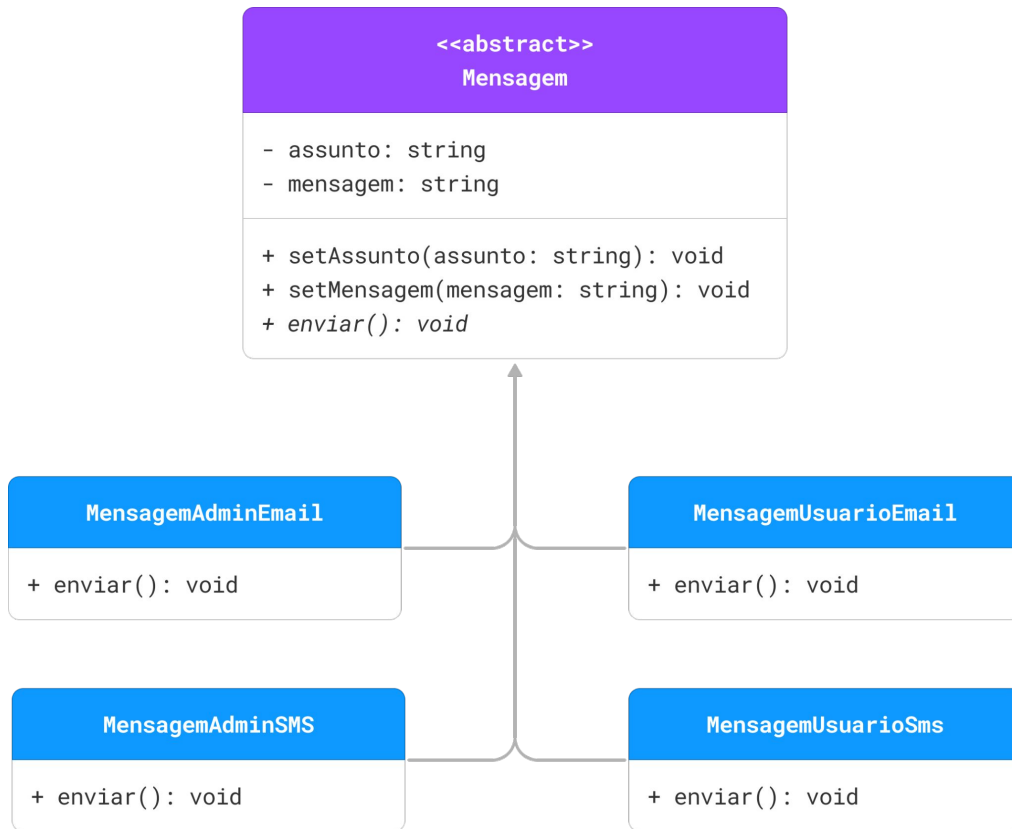


Bridge

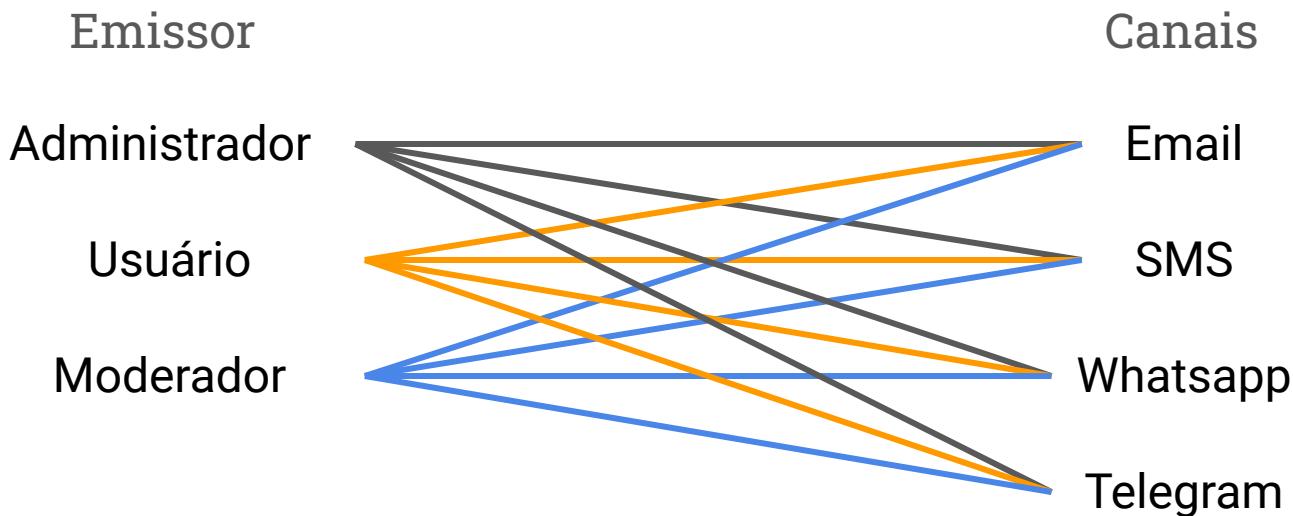
Definição

O padrão Bridge *desacopla uma abstração de sua implementação*, ou seja, permite dividir uma classe grande ou um conjunto de classes diretamente ligadas em duas hierarquias separadas, deste modo, as duas hierarquias podem variar de forma independente.

Bridge

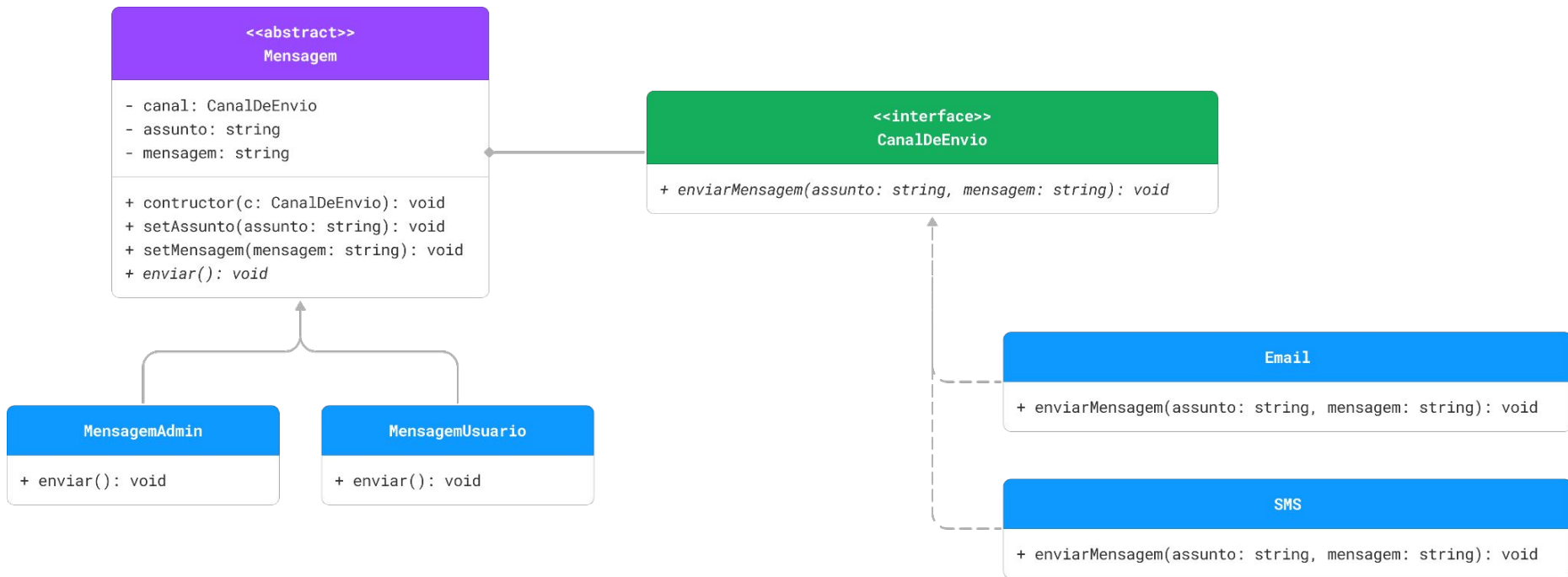


Bridge



Nossas classes crescem em duas dimensões: Emissor e Canal, pois a implementação do envio da mensagem por canal está diretamente ligada ao emissor que está enviando a mensagem.

Bridge





Bridge

Aplicabilidade

- Quando se deseja evitar uma ligação permanente entre uma abstração e sua implementação.
- Quando tanto as abstrações quanto suas implementações devem ser extensíveis por subclassificação.
- Quando existe uma proliferação de classes.



Bridge

Consequências

- Desacopla a interface da implementação.
- Extensibilidade aprimorada.
- Ocultação de detalhes de implementação dos clientes.



Bridge

Exercício

Usando o padrão de projeto Bridge, implemente um sistema de desenho onde existem diferentes formas geométricas (como "Círculo" e "Quadrado") e diferentes tipos de renderização (como "Renderizador 2D" e "Renderizador 3D").

Instruções:

- Crie uma interface **Renderizador** com métodos como **renderizarForma()**.
- Implemente duas classes concretas de renderização: **Renderizador2D** e **Renderizador3D**.
- Crie uma classe abstrata **Forma**, que possui um **Renderizador**.
- Crie as classes concretas **Circulo** e **Quadrado**, que herdam de **Forma** e utilizam o **Renderizador**.