Design Patterns

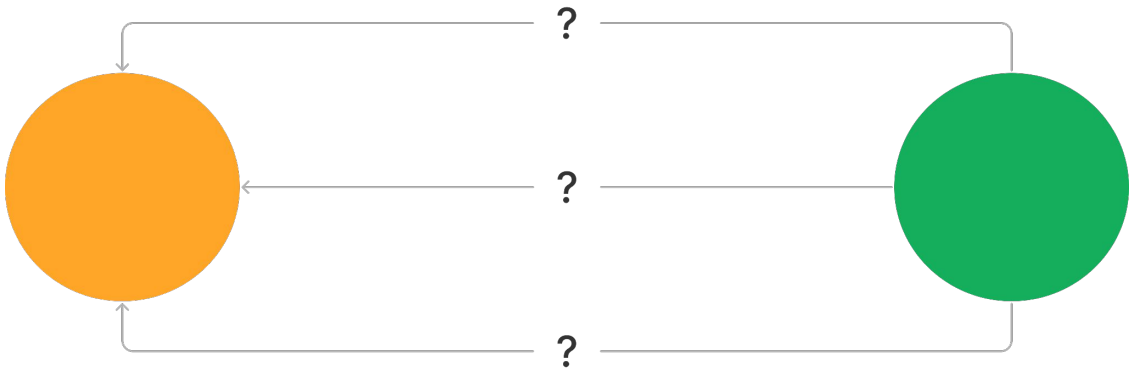
(Padrões de Projeto)

Observer

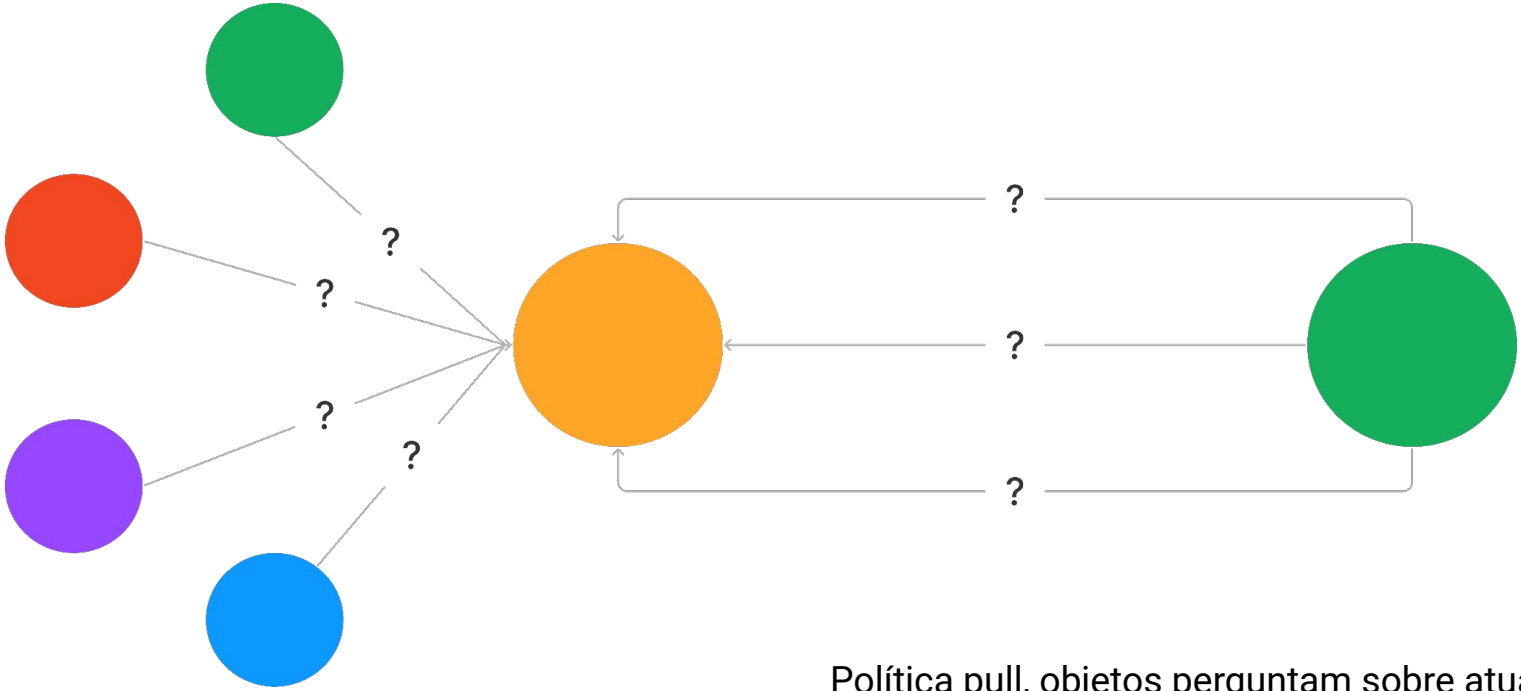
Definição

É um padrão de projeto que *define uma dependência um-para-muitos* entre objetos, de modo que *quando um objeto muda seu estado, todos seus dependentes são notificados* e atualizados automaticamente.

Observer

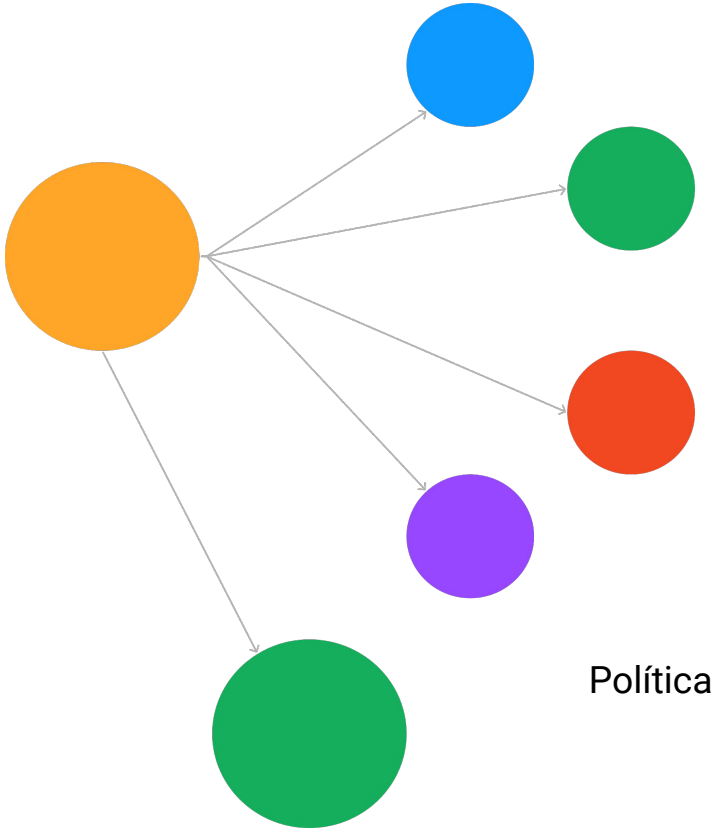


Observer



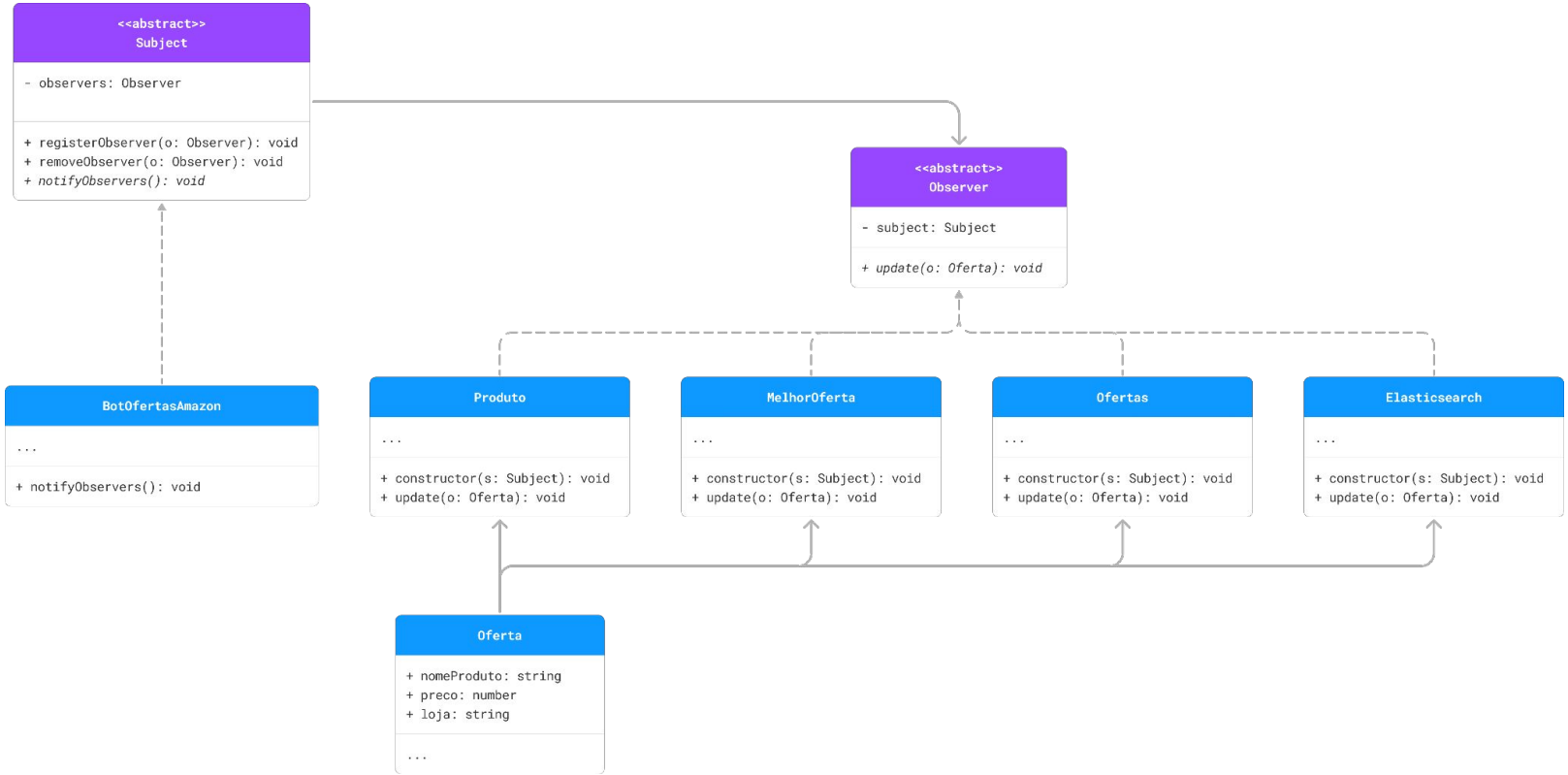
Política pull, objetos perguntam sobre atualizações do objeto observado.

Observer



Política push, objeto observado vai informar sobre uma atualização.

Observer


<<abstract>>
Subject
- observers: Observer
+ registerObserver(o: Observer): void
+ removeObserver(o: Observer): void
+ notifyObservers(): void
<<abstract>>
Observer
- subject: Subject
+ update(o: Oferta): void
BotOfertasAmazon
...
+ notifyObservers(): void
Produto
...
+ constructor(s: Subject): void
+ update(o: Oferta): void
MelhorOferta
...
+ constructor(s: Subject): void
+ update(o: Oferta): void
Ofertas
...
+ constructor(s: Subject): void
+ update(o: Oferta): void
Elasticsearch
...
+ constructor(s: Subject): void
+ update(o: Oferta): void
Oferta
+ nomeProduto: string
+ preco: number
+ loja: string
...

Observer

Aplicabilidade

- Quando uma abstração tem dois aspectos, um depende do outro, é necessário que eles possam variar e serem reutilizados independentemente.
- Quando uma alteração em um objeto requer a alteração de outros e não se conhece quantos objetos precisam ser alterados.
- Quando um objeto deve ser capaz de notificar outros objetos sem os conhecer, ou seja, tais objetos não podem ser fortemente acoplados.

Observer

Consequências

- O padrão Observer permite variar e reutilizar assuntos e observadores de forma independente.
- Acoplamento abstrato entre assunto e observador.
- Suporte para comunicação via broadcast.
- Pode causar atualização inesperadas e cíclicas.

Observer

Exercício

Usando o padrão de projeto Observer, crie um sistema de monitoramento de temperatura que deverá ter:

1. Uma classe SensorDeTemperatura, que armazena a temperatura e notifica os observadores quando a temperatura muda.
2. A interface Observer, que define o método de atualização dos observadores.
3. Observadores DisplayCelular e DisplayPainel, que reagem às mudanças de temperatura e mostram a nova temperatura.