

Mediator

Propósito

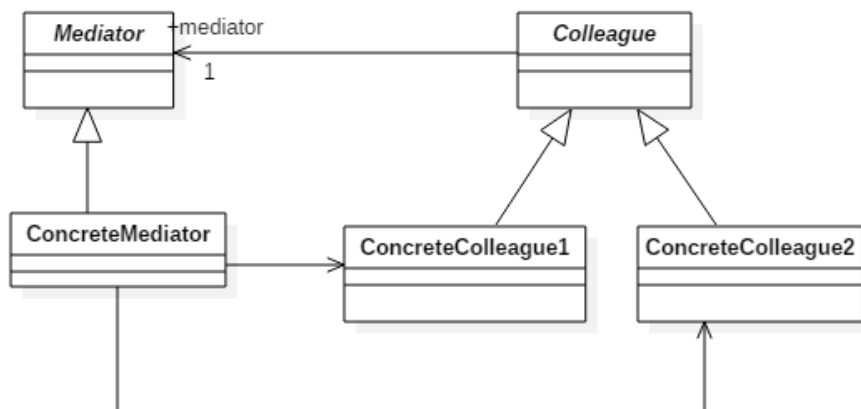
Define un objeto que encapsula la interacción entre un conjunto de objetos. El Mediator promueve un desacoplamiento débil al evitar que los objetos tengan referencia al resto de objetos de forma explícita, lo cual permite variar su interacción de forma independiente.

Aplicabilidad

El patrón Mediator se utiliza cuando:

- Un conjunto de objetos se comunica entre sí de un modo bien definido pero complejo. Las interdependencias resultantes no están estructuradas y son difíciles de entender
- Reutilizar un objeto es difícil porque tiene referencias y se comunica con muchos otros objetos
- Se desea configurar un comportamiento que se distribuye entre varias clases pero se quiere evitar realizar muchas subclases

Estructura



Participantes

- **Mediator**: define una interface para comunicar con objetos **Colleague**
- **ConcreteMediator**: Implementa el comportamiento cooperativo mediante la coordinación entre objetos **Colleague**. Conoce y mantiene a sus colegas
- **Colleague**: cada clase **Colleague** conoce a su objeto **Mediator**. Cada colega se comunica con su mediador siempre que quiere comunicarse con algún otro colega

Colaboraciones

- Los colegas envían y reciben peticiones al objeto **Mediator**. El mediador implementa el comportamiento cooperativo enviando las peticiones a los colegas que corresponda

Consecuencias

El patrón Mediator tiene los siguientes beneficios y desventajas:

- Limita las subclases. Un mediador centraliza el comportamiento que de otro modo estaría distribuido entre varios objetos. Cambiar este comportamiento implica únicamente subclasificar el Mediator. Las clases Colleague se pueden reutilizar
- Desacopla a los colegas. Un mediador promueve el desacoplamiento débil entre colegas. Puedes variar y reutilizar las clases Colleague y Mediator de forma independiente.
- Simplifica los protocolos de los objetos. El mediador reemplaza las interacciones muchos-a-muchos por interacciones uno-a-muchos entre el mediador y sus colegas. Las relaciones un-a-muchos son más fáciles de comprender, mantener y extender
- Abstrae el modo en que cooperan los objetos. Al encapsular la mediación en un objeto, permite centrarse en la interacción entre los objetos como algo independiente de su comportamiento individual. Esto puede ayudar a clarificar cómo interactúan los objetos en un sistema
- Centraliza el control. El patrón Mediator cambia la complejidad en la interacción por complejidad en el mediador. Como el mediador encapsula los protocolos, puede convertirse en un objeto más complejo que un colega. Esto puede tener como consecuencia que el mediador sea monolítico y difícil de mantener