

# Mediator

Patrones de diseños

# Patrón de diseño

- Es una solución a un problema de diseño.
- Para que sea considerado patrón de diseño debe cumplir las siguientes características:
  - Debe haber comprobado su efectividad resolviendo problemas similares en ocasiones anteriores.
  - Debe ser reutilizable, lo que significa que es aplicable a diferentes problemas de diseños en diferentes circunstancias

# Mediator

- **Clasificación:** De comportamiento
  - Debido al hecho de que puede alterar el programa en ejecución.
- **Intención:** Definir un objeto que encapsule como interactúa un conjunto de objetos.

# Usar el patrón cuando:

- Un conjunto grande de objetos se comunica de una forma bien definida pero compleja.
- Reutilizar un objeto se hace difícil porque se relaciona con muchos objetos.
- El comportamiento de muchos objetos que está distribuido entre varias clases puede resumirse en una o varias por subclasificación.

# Participantes

- **Mediator(BackendProxy y FrontedProxy):** Implementa el comportamiento cooperativo entre los colegas (como se comunican entre ellos). Además los conoce y mantiene. Debido a las peculiaridades de DBUS, FrontedProxy recibe las peticiones de GUI y UI y se las transmite a BackendProxy, que es el encargado de enviarla al colega destinatario.
- **Colleagues:** Cada colega conoce su mediador y usa a este para comunicarse con otros colegas

# Ventajas y Desventajas

- **Desacopla a los colegas:** El patrón Mediator promueve bajar el acoplamiento entre colegas. Se puede variar y reutilizar colegas y mediadores independientemente.
- **Simplifica la comunicación entre objetos:** Los objetos que se comunican de la forma “muchos a muchos” puede ser reemplazada por una forma “uno a muchos” que es menos compleja y mas elegante. Es más fácil de entender.

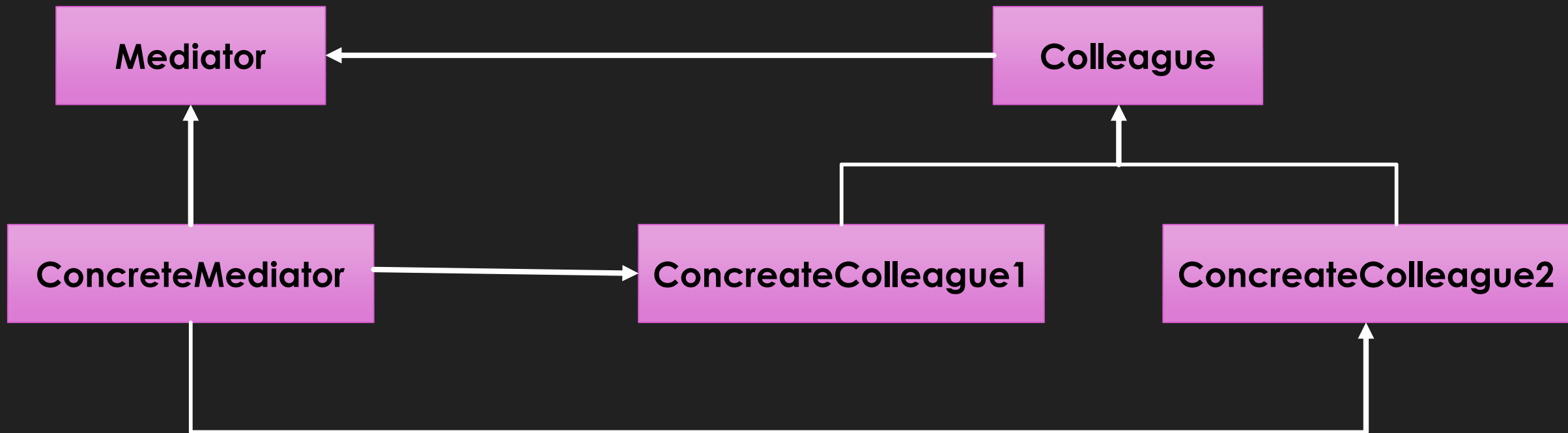
- **Abstrae como los objetos cooperan:** Haciendo a la mediación un concepto independiente y encapsulándolo en un objeto, permite enfocar como los objetos interactúan. Esto ayuda a clarificar como los objetos se relacionan en un sistema.
- **Centraliza el control:** El mediador es el que se encarga de comunicar a los colegas, este puede ser muy complejo, difícil de entender y modificar.

# Ejemplo Vida Real

- Si queremos enviar una carta no la entregamos nosotros ni la vamos pasando a los vecinos a ver si algún día llega a su destino. En lugar de eso tenemos un intermediario, en este caso es la oficina de correos, que se encarga de recorrer los mensajes y entregarlos al destino.



# Estructura



# Donde:

- **Mediator**: Establece las operaciones que pueden realizarse sobre el sistema encapsulado.
- **ConcreteMediator**: Implementa las operaciones anteriores y define la interacción de los elementos del sistema. Actúa de procesador central.
- **ConcreteColleague**: Es el integrante real del sistema que implementa todas las operaciones establecidas por la interfaz Colleague y las suyas propias.