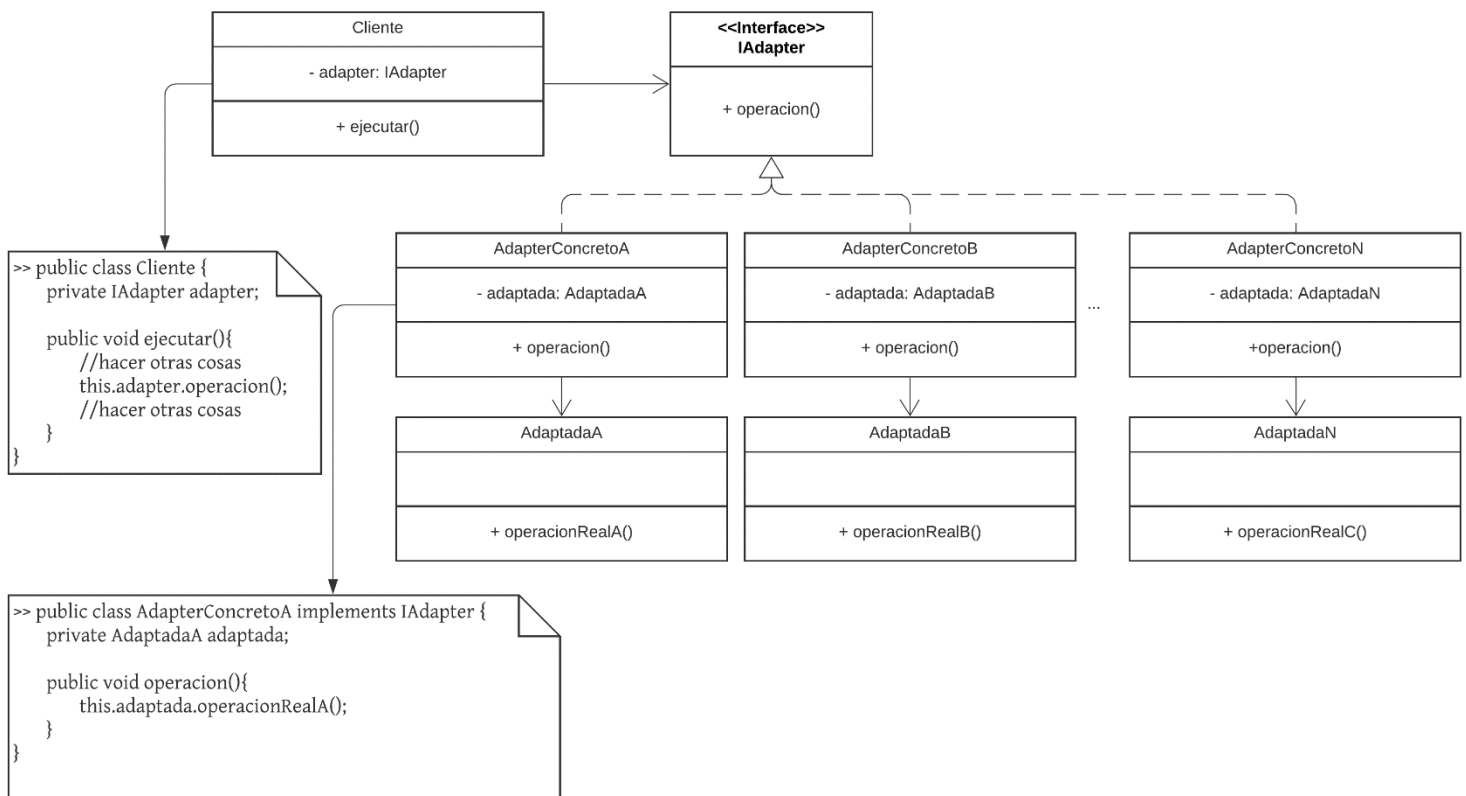


Patrón Adapter

- **Clasificación:** Patrón estructural
- **¿Qué hace?**
 - ✓ Encapsula el uso (llamadas/envío de mensajes) de la clase que se quiere adaptar en otra clase que concuerda con la interfaz requerida.
- **Se sugiere su utilización cuando:**
 - ✓ Se requiere seguir adelante con el diseño y/o implementación (código) sin conocer exactamente cómo, quién y cuándo resolverá una parte necesaria; y solamente se conoce qué responsabilidad tendrá dicha parte.
 - ✓ Se requiere usar una clase que ya existe, pero su interfaz no concuerda con la que se necesita.
- **Estructura genérica:**



Consideraciones:

- ✓ Los métodos no necesariamente son void.
- ✓ Pueden no existir N adapters concretos.

- ¿Qué proporciona su uso?
 - ✓ Mayor mantenibilidad debido al nulo acoplamiento entre el cliente y la clase adaptada.
 - ✓ Mayor cohesión de la clase cliente debido a la delegación de cierto comportamiento en el adapter.
 - ✓ Mayor facilidad de testeo ya que se podría implementar un mock de AdapterConcreto.
- Code smells que soluciona/evita de forma directa:
 - ✓ Herencia rechazada
 - ✓ Clases alternativas con diferentes interfaces
 - ✓ Clase de biblioteca incompleta