

Adaptador (Adapter)

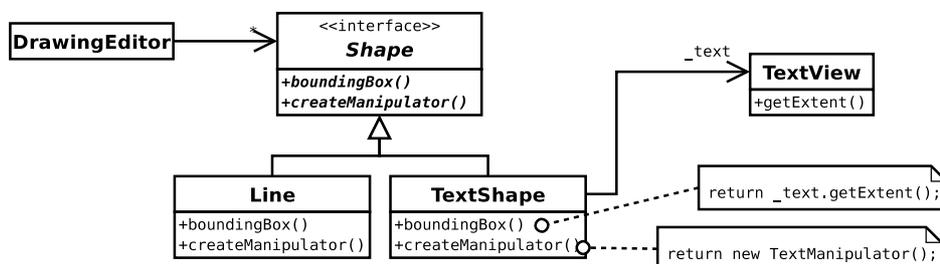
- *Patrón Estructural*

- *Propósito*

El adaptador o *wrapper* permite la colaboración entre clases con interfaces incompatibles

- *Motivación*

- Reutilizar clases con interfaces incompatibles
- Ejemplo: Editor gráfico que manipula objetos que cumplen una determinada interfaz (*Shape*)
 - Clases elementales (líneas y polígonos) sencillas
 - Edición de texto complicada: reutilizar clase existente en otra biblioteca (*TextView*)
- Problema: La biblioteca de *TextView* no fue diseñada pensando en la interfaz *Shape*
- Solución (¿?): Modificar la clase *TextView* para que cumpla la interfaz *Shape* (inaceptable y en ocasiones imposible)
- Solución: Utilizar un *adaptador*.



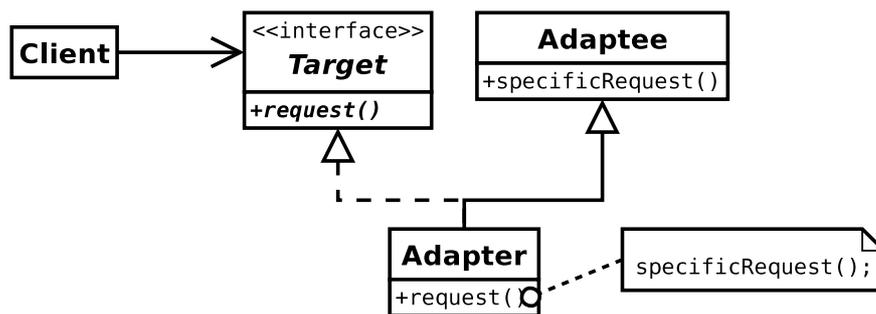
■ Aplicabilidad

- Se desea utilizar una clase existente y la interfaz no es compatible
- Se desea crear clases reusables que cooperan con clases no relacionadas
- Se desea utilizar diversas subclases existentes pero resulta impráctico adaptar sus interfaces mediante la extensión de todas ellas (utilizar un objeto adaptador de la clase padre)

■ Estructura

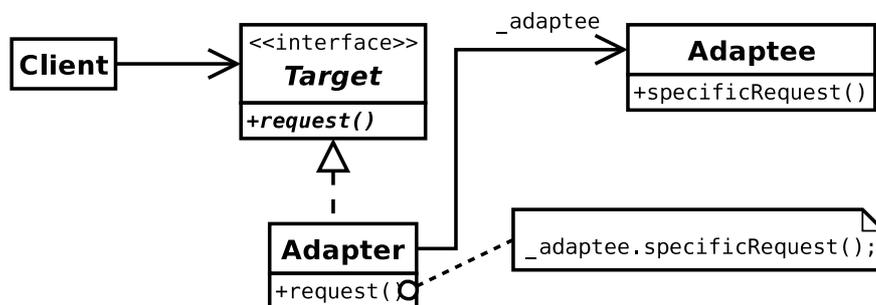
- Opción 1: Clase adaptadora (herencia múltiple)

Heredar interfaz pública (*Target*) e implementación (*Adaptee*)



- Opción 2: Objeto adaptador (composición)

Composición para implementar un objeto compatible con *Target* a partir de *Adaptee*



■ *Participantes*

- Objetivo (Target)
Define interfaz dependiente del dominio usada por el cliente
- Cliente (Client)
Colabora con los objetos de acuerdo con interfaz Objetivo
- Adaptado (Adaptee)
Define una interfaz existente que necesita ser adaptada
- Adaptador (Adapter)
Adapta la interfaz del adaptado a la interfaz Objetivo

■ *Colaboraciones*

- El cliente envía mensajes al adaptador y éste, en respuesta, los envía al objeto adaptado.

■ *Consecuencias*

- En ocasiones, permite la incorporación de funcionalidades no disponibles en el adaptado
- Clase adaptadora
 - No permite adaptar una clase y todas sus subclases
 - Permite que el adaptador redefina parte del comportamiento del adaptado
 - No hay una nueva indirección (sólo un objeto)
- Objeto adaptador
 - Un adaptador permite adaptar a diversos objetos (extensiones de una clase común)
 - Difícil cambiar el comportamiento del adaptado (extender adaptado y adaptar la nueva subclase)