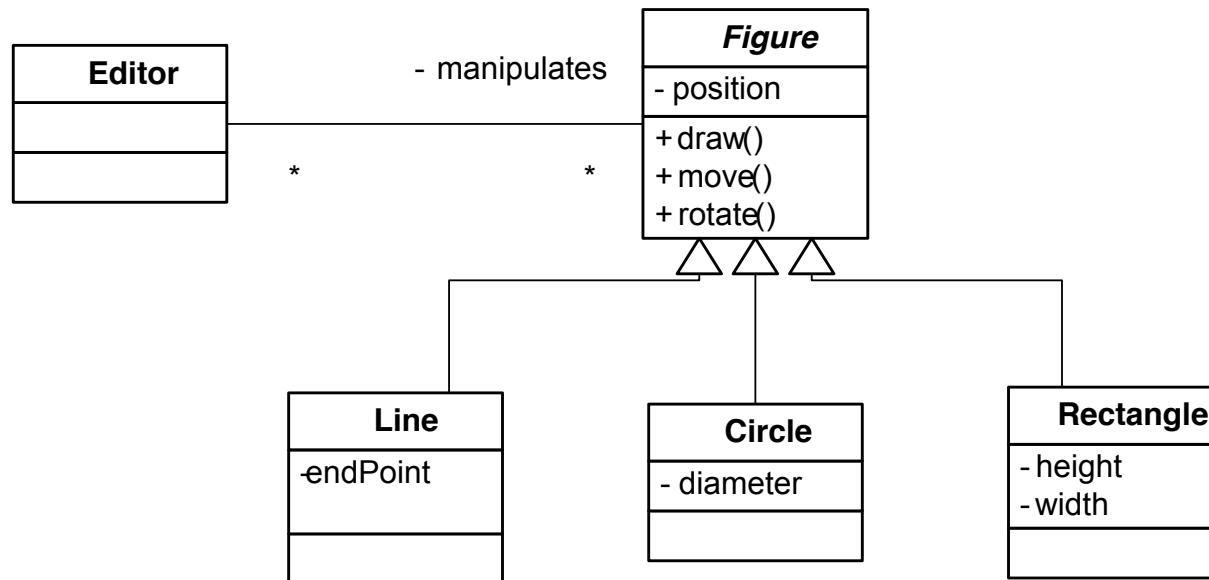


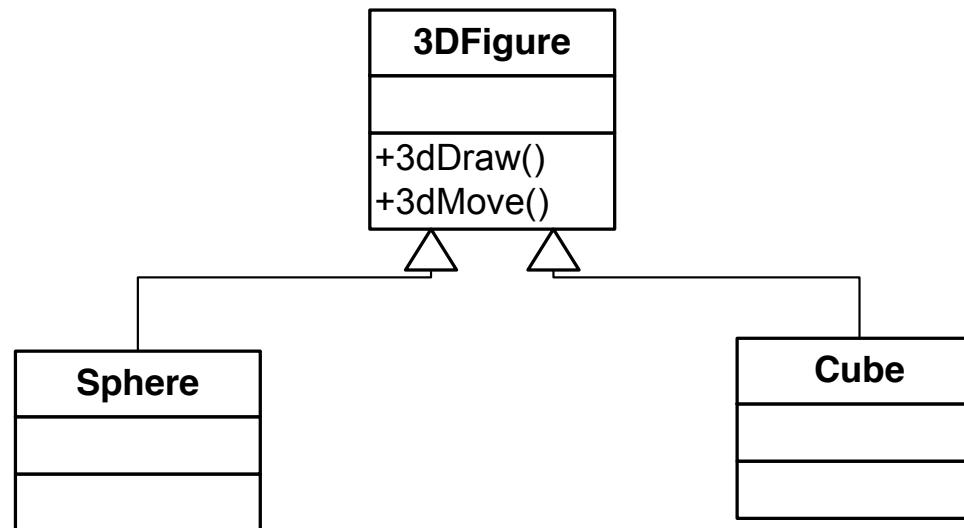
Ejemplo

- ✓ Editor grafico para manejar figuras geometricas:
- ✓ Observen polimorfismo en la relacion Editor-Figure



Ejemplo..

- ✓ Queremos extender el editor a figuras 3D

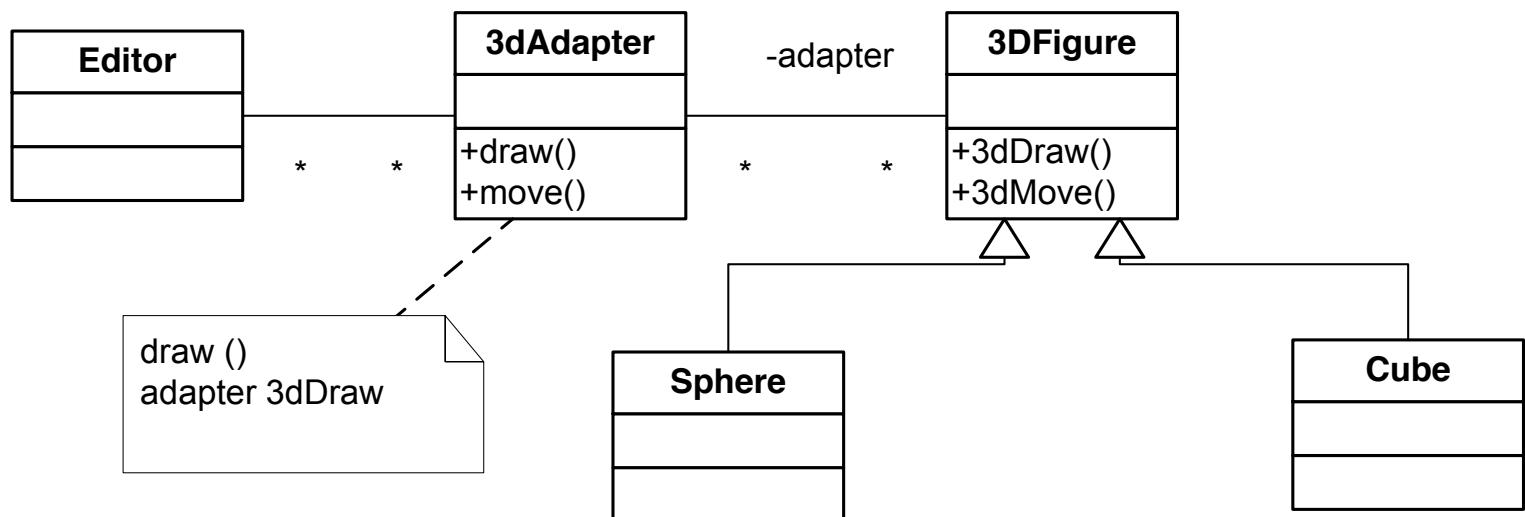


Problema

- ✓ Como integramos esta jerarquia?
- ✓ Que problemas tenemos?
- ✓ Sacrificamos polimorfismo?
- ✓ Editamos el codigo de la nueva jerarquia?

Solution

- ✓ Cuando tratamos con interfaces incompatibles, intentar adaptarlas.



3dAdapter is sub-clase de?

Adapter

✓ **Intencion:**

“Convertir” la interfaz de una clase en otra que el cliente espera
EL Adapter permite que ciertas clases trabajen en conjunto cuando no podrian por tener interfaces incompatibles

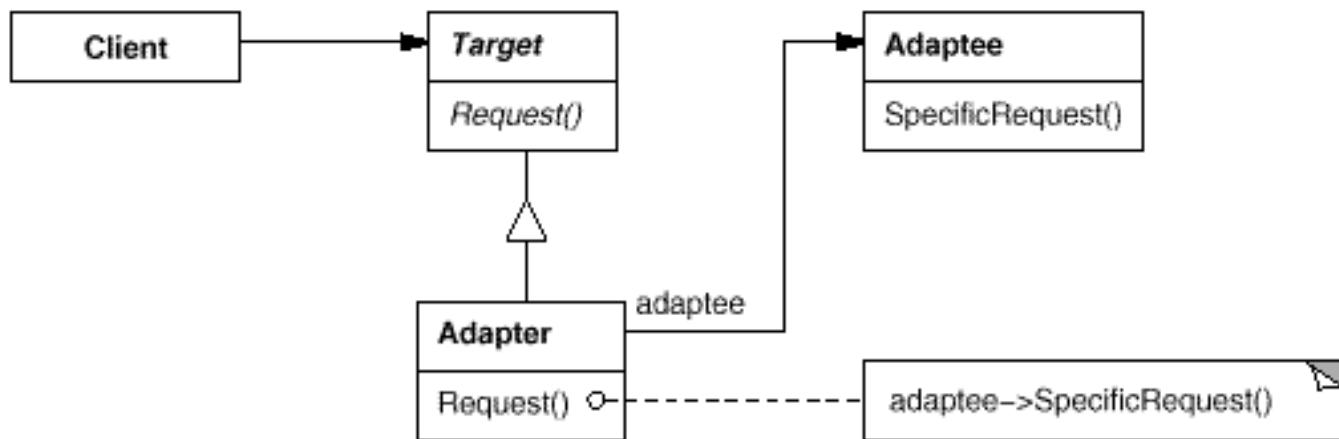
✓ **Applicabilidad:**

Use el adapter cuando:

- ✓ Ud quiere usar una clase existente y su interfaz no es compatible con lo que precisa

Adapter

✓ Estructura



Adapter. Participants

✓ **Participantes:**

✓ **Target (Figure)**

✓ defines the domain-specific interface that Client uses.

✓ **Client (Editor)**

✓ collaborates with objects conforming to the Target interface.

✓ **Adaptee (3DFigure)**

✓ defines an existing interface that needs adapting.

✓ **Adapter (3DAdapter)**

✓ adapts the interface of Adaptee to the Target interface.

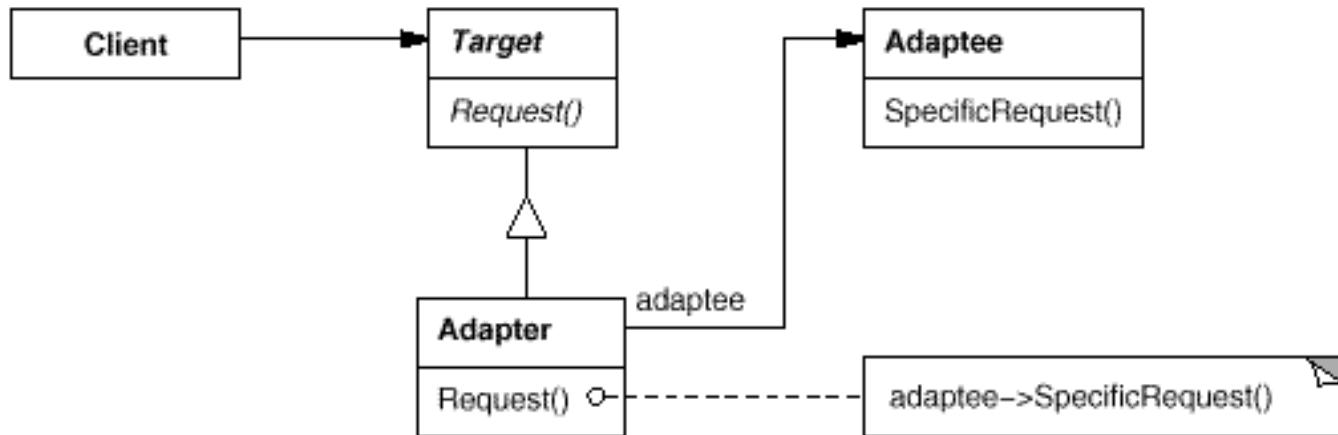
Descubriendo Patrones

- ✓ Como es el proceso de descubrimiento?
- ✓ Que tipo de observacion/abstraccion realizamos?

Usando Patrones

- ✓ Supongamos que conocemos patrones(e.g. Adapter).
- ✓ Como mapeamos un patron a un diseño especifico?
- ✓ Como aplicamos el principio de Alexander (“use the patterns millions of times without doing the same thing twice”)?

Adapter



**Como usamos esta informacion? Es suficiente?
Que mas necesitamos?**