

Patrons de Conception

-

Adapter

Simon Urli
urli@i3s.unice.fr

Master I MIAGE
2014-2015

Objectif

“ Permettre l’adaptation d’une classe à une autre interface qui est attendue par le client : autoriser ainsi des classes ayant des interfaces incompatibles à collaborer. ”

Synonyme : wrapper

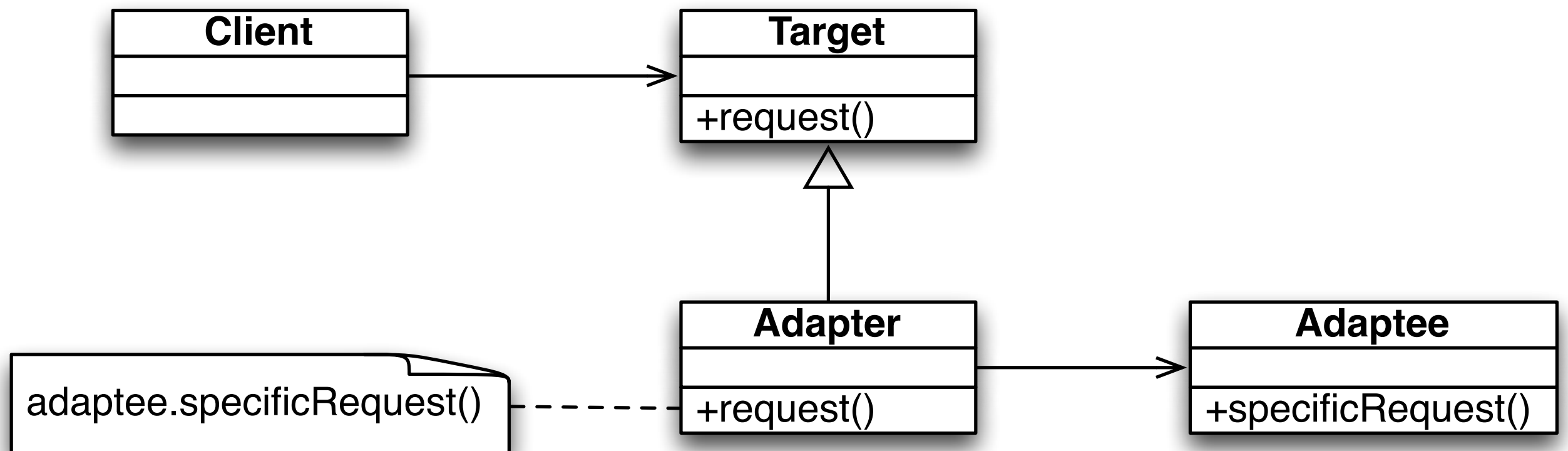
Exemple



Application

- Volonté d'utiliser une classe alors que l'interface ne convient pas.
- Réutilisation d'un code fournit.
- Intégration d'une bibliothèque de code.

Structure



Implémentation

- Client utilise des objets de type Target.
- Target définit l'interface spécifique à l'application que le client utilise.
- Adaptee est un objet existant que l'on veut utiliser dans Client.
- Adapter est conforme à l'interface Target et encapsule les méthodes de Adaptee.

Exemple 1/3

```

public interface FrenchPlug extends Plug {
    public FemaleConnector getGround();
    public MaleConnector getNeutral();
    public MaleConnector getPhase();
    public void plug(FrenchSocket fs);
}

public interface EnglishPlug extends Plug {
    public MaleConnector getGround();
    public MaleConnector getPhase();
    public MaleConnector getNeutral();
    public void plug(EnglishSocket es);
}

public interface FrenchSocket extends Socket {
    public void connectGround(FemaleConnector c);
    public void connectNeutral(MaleConnector c);
    public void connectPhase(MaleConnector c);
}

public interface EnglishSocket extends Socket {
    public void connectGround(MaleConnector c);
    public void connectNeutral(MaleConnector c);
    public void connectPhase(MaleConnector c);
}
  
```



Exemple 2/3

```
public class ConcreteEnglishPlug implements EnglishPlug {  
    private MaleConnector ground;  
    private MaleConnector phase;  
    private MaleConnector neutral;  
  
    public void plug(EnglishSocket s) {  
        s.connectGround(this.ground);  
        s.connectNeutral(this.neutral);  
        s.connectPhase(this.phase);  
    }  
  
    public MaleConnector getGround() { return this.ground; }  
    public MaleConnector getPhase() { return this.phase; }  
    public MaleConnector getNeutral() { return this.neutral; }  
}
```


Exemple 3/3

```
public class FrenchAdapterEnglishPlug implements FrenchPlug {  
    private EnglishPlug plug;  
    private EnglishSocket internal;  
    private FemaleConnector ground;  
    public FrenchAdapterEnglishPlug(EnglishPlug ep) {  
        this.plug = ep;  
        this.internal = new EnglishSocket();  
        this.ground = new FemaleConnector();  
    }  
    public void plug(FrenchSocket s) {  
        this.plug.plug(this.internal);  
        this.ground.connectedTo(this.plug.getGround());  
        s.connectGround(this.getGround());  
        s.connectNeutral(this.getNeutral());  
        s.connectPhase(this.getPhase());  
    }  
    public MaleConnector getGround() { return this.ground; }  
    public MaleConnector getPhase() { return this.plug.getPhase(); }  
    public MaleConnector getNeutral() { return this.plug.getNeutral(); }  
}
```

Exercice

- Implémenter une interface pour les mouvements d'une souris et une autre pour les actions reconnues sur un touchpad. Implémenter une souris concrète. Implémenter finalement un adapter pour manipuler cette souris à partir d'un objet conforme à l'interface du touchpad.