

10. Шаблон Адаптер (Adapter)

Лекционен курс: Шаблини за проектирне
доц. д-р Емил Дойчев

Общи сведения

- ✓ Вид: Структурен шаблон
- ✓ Цел: Преобразува интерфейса на даден клас в друг такъв, когато има несъвместимост на интерфейсите. Позволява на класовете да работят заедно, което не би било възможно иначе поради несъвместимост на интерфейсите.
- ✓ Известен и като: Обвивка (Wrapper)

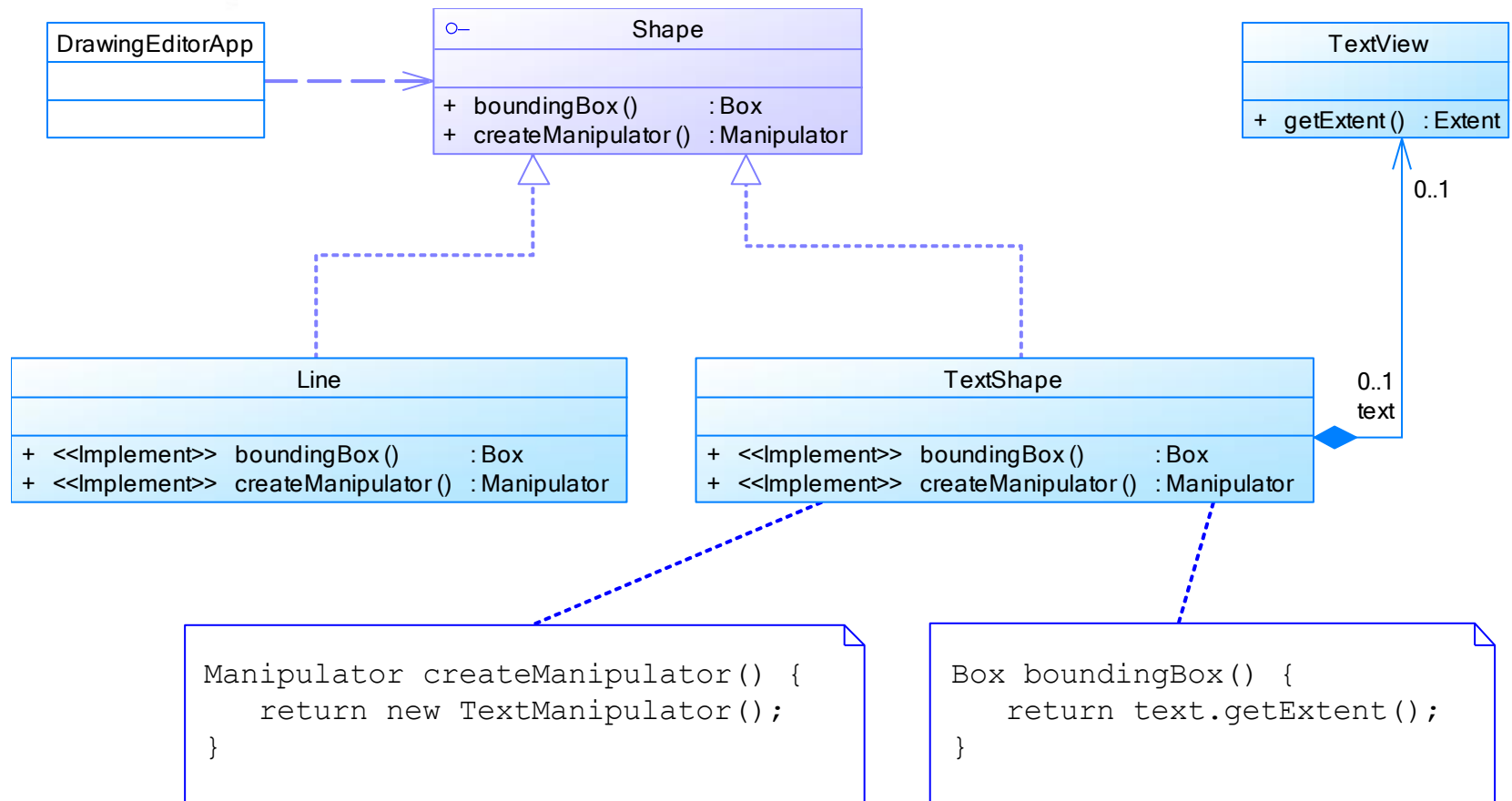
Мотивация

- ✓ Налице са 2 несъвместими класа с несъвместими интерфейси, които в даден момент трябва да работят заедно.
- ✓ Чрез шаблона Адаптер се решава този проблем без нуждата от допълнително имплементиране на функционалността на някой от създадените класове.

Мотивация

- ✓ Например дадена библиотека с класове не може да бъде използвана тъй като нейният интерфейс е несъвместим с изискванията от приложението.
- ✓ Интерфейсът на библиотеката не може да бъде променен защото не разполагаме със сорс кодът ѝ.
- ✓ Дори и да разполагахме с него най-вероятно не бихме променяли интерфейса на библиотеката за всяко специфично приложение.

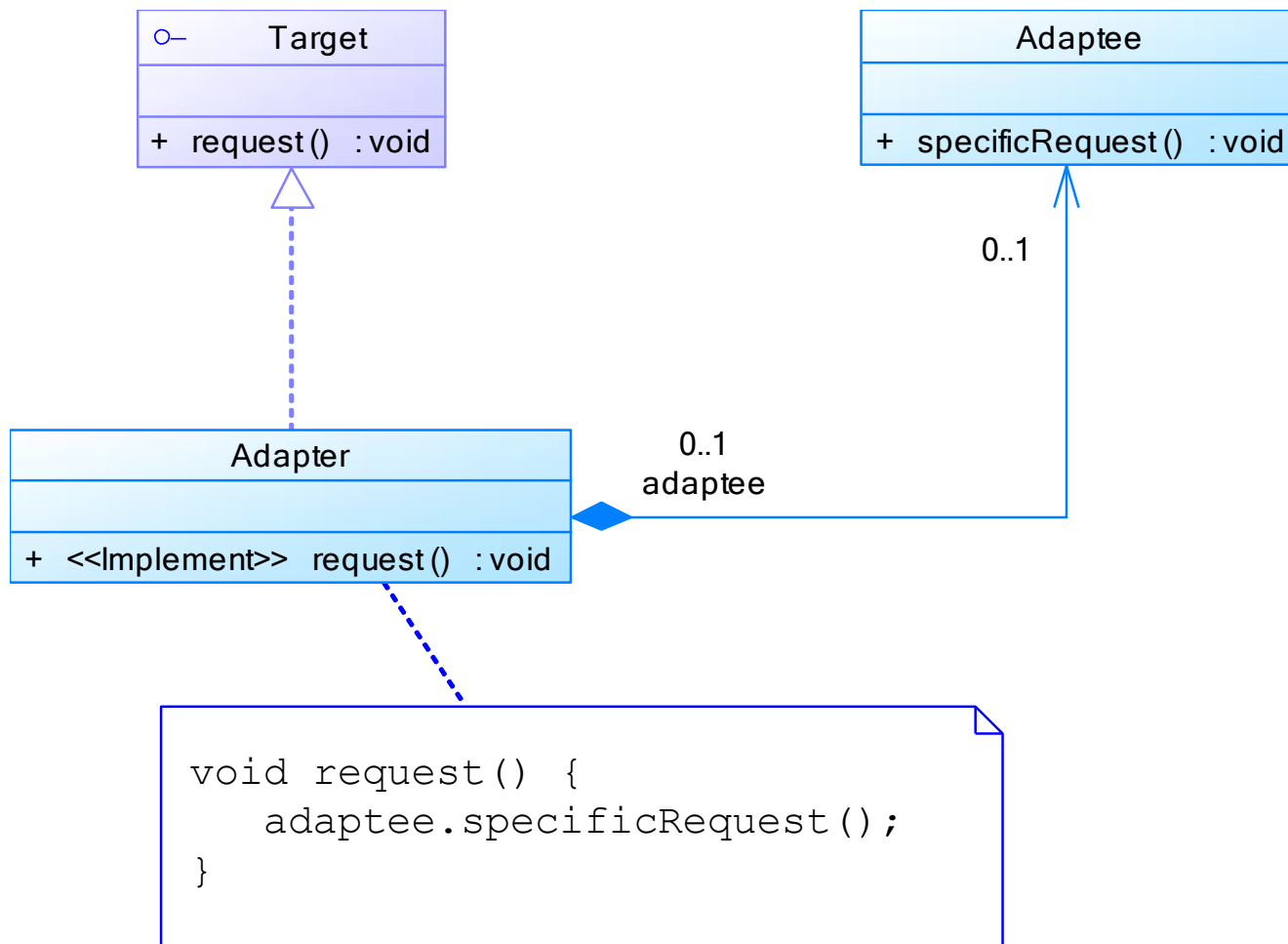
Мотивация



Употреба

- ✓ Когато трябва да се използва клас, чийто интерфейс не е съвместим с целта, която трябва да се постигне.
- ✓ Многократна употреба на клас с класове чиито интерфейси нямат обща съвместимост
- ✓ Използване на библиотеки от трети страни

Структура



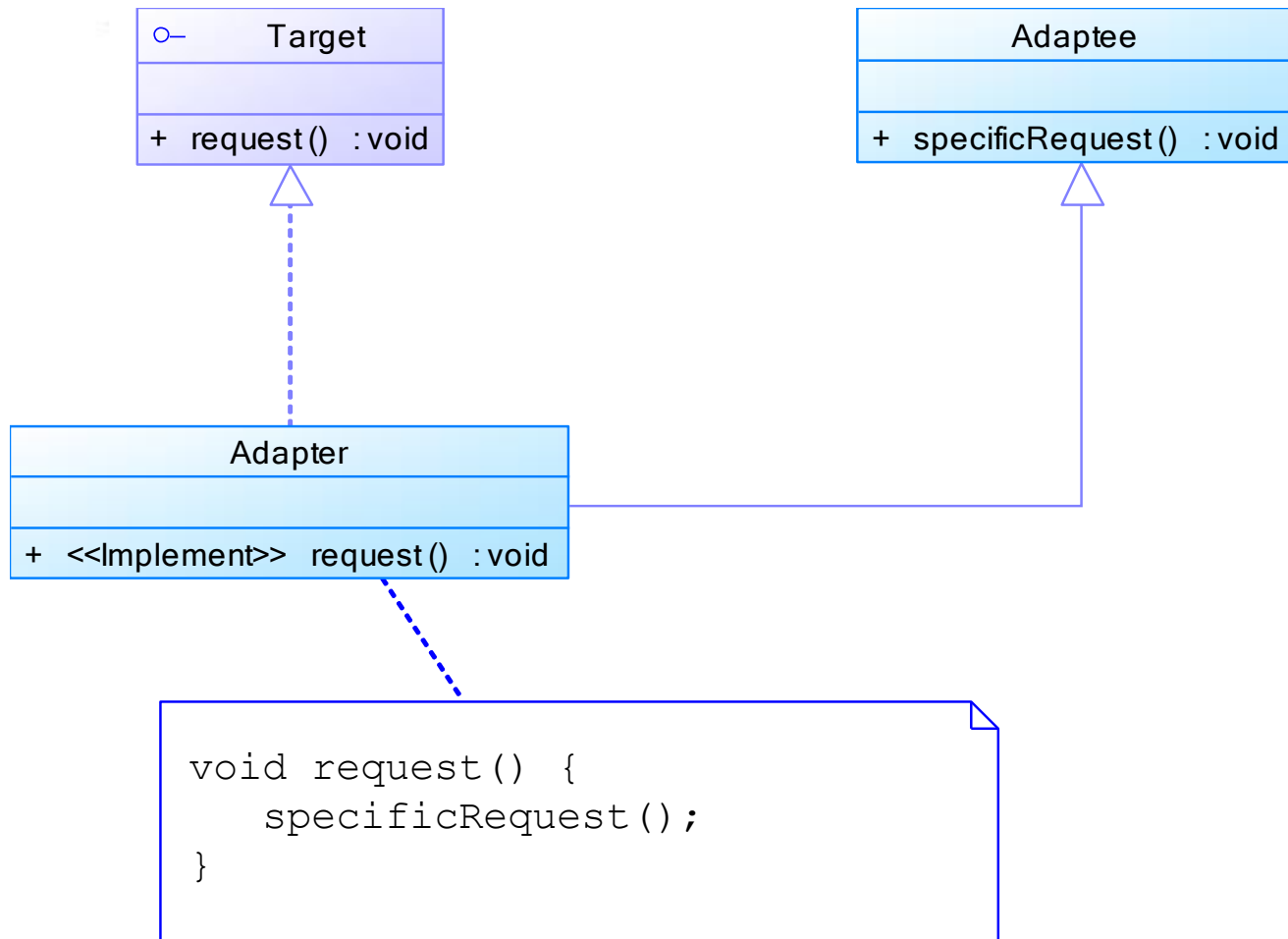
Участници

- ✓ **Target** – интерфейс, който **Client** използва директно
- ✓ **Adapter** – адаптира интерфейса **Adaptee** към **Target** интерфейса
- ✓ **Adaptee** – интерфейс, който трябва да бъде адаптиран
- ✓ **Client** – обект ползващ **Target** интерфейса

Разновидности

- ✓ Клас Адаптери (**Class Adapters**)– Налице са в езиците, които поддържат множествено наследяване.
Класа Adapter наследява Target и Adaptee

Разновидности



Имплементация на шаблона

✓ Налице са:

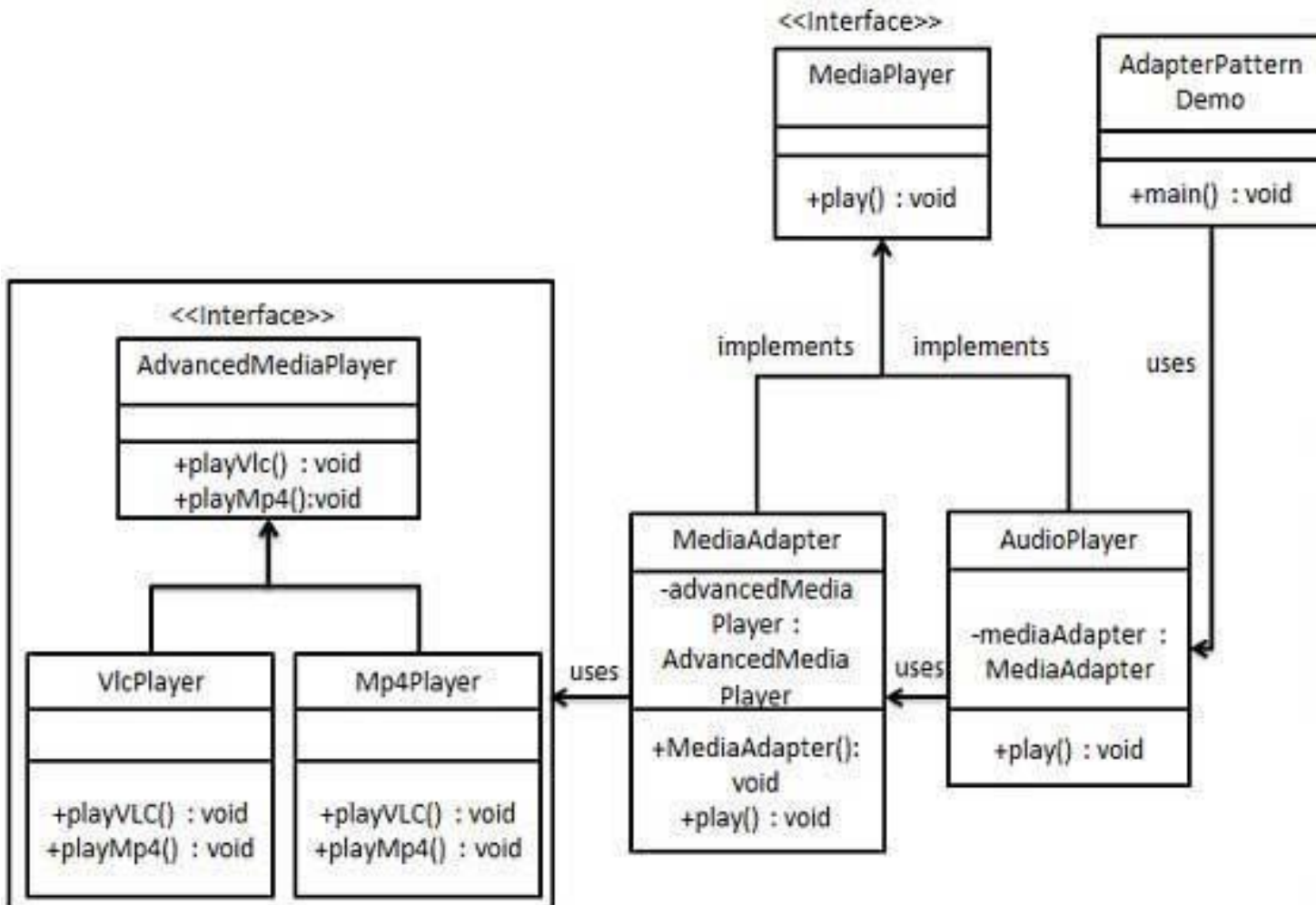
- интерфейс **MediaPlayer** и клас **AudioPlayer**, който имплементира **MediaPlayer**. **AudioPlayer** може да изпълнява mp3 файлове.
- интерфейс **AdvancedMediaPlayer** и класове имплементиращи **AdvancedMediaPlayer**, които изпълнява vlc и mp4 файлове.

✓ Задача: **AudioPlayer** трябва да изпълнява както mp3 файлове така и vlc, mp4 файлове

Имплементация на шаблона

- ✓ Създава се клас **MediaAdapter**, който имплементира **MediaPlayer** и ползва **AdvancedMediaPlayer**
- ✓ **AudioPlayer** използва **MediaAdapter** и имплементира интерфейса **MediaPlayer**. По този начин обектите от тип **AudioPlayer** вече могат да изпълняват желаните аудио формати.

Имплементация на шаблона



Имплементация на шаблона

MediaPlayer.java

```
public interface MediaPlayer {  
    public void play(String audioType, String fileName);  
}
```

AdvancedMediaPlayer.java

```
public interface AdvancedMediaPlayer {  
    public void playVlc(String fileName);  
    public void playMp4(String fileName);  
}
```

Имплементация на шаблона

VlcPlayer.java

```
public class VlcPlayer implements AdvancedMediaPlayer{
    @Override
    public void playVlc(String fileName) {
        System.out.println("Playing vlc file. Name: "+ fileName);
    }

    @Override
    public void playMp4(String fileName) {
        //do nothing
    }
}
```

Mp4Player.java

```
public class Mp4Player implements AdvancedMediaPlayer{

    @Override
    public void playVlc(String fileName) {
        //do nothing
    }

    @Override
    public void playMp4(String fileName) {
        System.out.println("Playing mp4 file. Name: "+ fileName);
    }
}
```

Имплементация на шаблона

MediaAdapter.java

```
public class MediaAdapter implements MediaPlayer {

    AdvancedMediaPlayer advancedMusicPlayer;

    public MediaAdapter(String audioType) {
        if(audioType.equalsIgnoreCase("vlc") ){
            advancedMusicPlayer = new VlcPlayer();
        } else if (audioType.equalsIgnoreCase("mp4")) {
            advancedMusicPlayer = new Mp4Player();
        }
    }

    @Override
    public void play(String audioType, String fileName) {
        if(audioType.equalsIgnoreCase("vlc")){
            advancedMusicPlayer.playVlc(fileName);
        } else if(audioType.equalsIgnoreCase("mp4")){
            advancedMusicPlayer.playMp4(fileName);
        }
    }
}
```


Имплементация на шаблона

AudioPlayer.java

```
public class AudioPlayer implements MediaPlayer {
    MediaAdapter mediaAdapter;

    @Override
    public void play(String audioType, String fileName) {

        //inbuilt support to play mp3 music files
        if(audioType.equalsIgnoreCase("mp3")){
            System.out.println("Playing mp3 file. Name: "+ fileName);
        }
        //mediaAdapter is providing support to play other file formats
        else if(audioType.equalsIgnoreCase("vlc")
            || audioType.equalsIgnoreCase("mp4")){
            mediaAdapter = new MediaAdapter(audioType);
            mediaAdapter.play(audioType, fileName);
        }
        else{
            System.out.println("Invalid media. "+
                audioType + " format not supported");
        }
    }
}
```

Имплементация на шаблона

AdapterPatternDemo.java

```
public class AdapterPatternDemo {  
    public static void main(String[] args) {  
        AudioPlayer audioPlayer = new AudioPlayer();  
  
        audioPlayer.play("mp3", "beyond the horizon.mp3");  
        audioPlayer.play("mp4", "alone.mp4");  
        audioPlayer.play("vlc", "far far away.vlc");  
        audioPlayer.play("avi", "mind me.avi");  
    }  
}
```

Резултат

```
Playing mp3 file. Name: beyond the horizon.mp3  
Playing mp4 file. Name: alone.mp4  
Playing vlc file. Name: far far away.vlc  
Invalid media. avi format not supported
```