Мост

1. Назначение и цель

Отделить абстракцию от реализации так, чтобы и то и другое можно было изменять независимо.

1. UML-диаграмма



1. Области применения: коллективная разработка проектов
2. Особенности

На все реализации заводим общий интерфейс, который они (реализации) будут имплементировать. В интерфейсе абстракции храним ссылку на интерфейс реализации. Таким образом можно совершенно независимо модифицировать абстракцию — путем уточнения интерфейса абстракции и реализацию — путем имплементации интерфейса реализации.

1. Примеры реализации

class Program

{

static void Main(string[] args)

{

Abstraction ab = new RefinedAbstraction();

// Set implementation and call

ab.Implementor = new ConcreteImplementor1();

ab.Operation();

// Change implementation and call

ab.Implementor = new ConcreteImplementor2();

ab.Operation();

// Wait for user

Console.ReadLine();

}

class Abstraction

{

public Implementor Implementor { get; set; }

public virtual void Operation()

{

Implementor.Operation();

}

}

class RefinedAbstraction:Abstraction

{

public override void Operation()

{

Implementor.Operation();

}

}

abstract class Implementor

{

public abstract void Operation();

}

class ConcreteImplementor1:Implementor

{

public override void Operation()

{

Console.WriteLine("this is ConcreteImplementor1");

}

}

{

public override void Operation()

{

Console.WriteLine("this is another implementor, ConcreteImplementor2");

}

}