



2、具体内容

UML是统一的建模语言，本质就是利用图形化的形式来实现程序类关系的描述，在之前已经画出了大量的图形，这些图形都是按照UML标准进行的，并且最早并且最广泛的设计工具：Rational Rose，而后出现越来越多的设计工具，而像PowerDesigner也可以实现这些类图的定义。

■类图描述

一般情况如果想要进行类结构的描述，往往可以使用三层的结构来示：

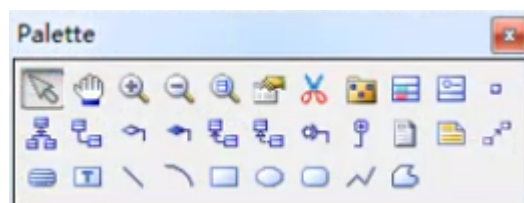
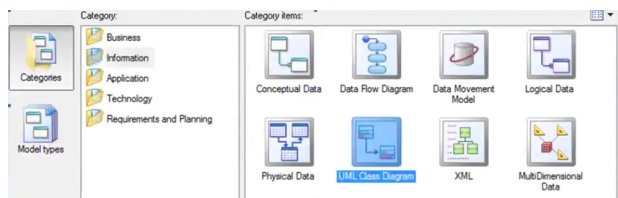
类名称
属性
方法

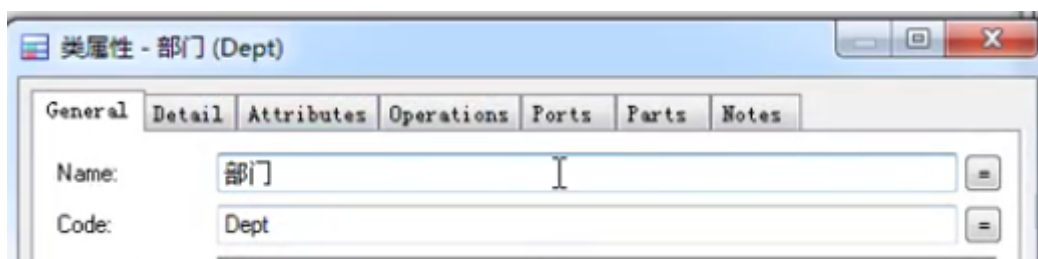
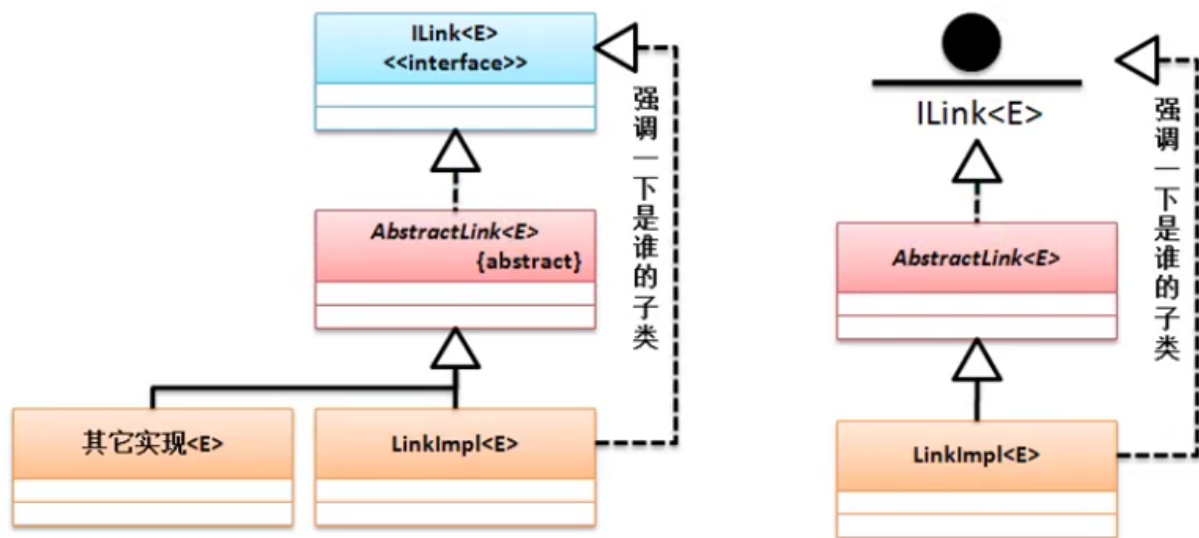
如果要是一个普通类的名称，往往直接编写即可，而如果是抽象类，往往使用斜体描述。“类名称”，所以为了更加清楚的描述，往往在抽象类上再加一个“abstract”。

对于类中的属性可以使用“访问权限 属性名称：属性类型”的格式来进行定义，而对于访问权限基本上重点值考虑三个：public (+)、protected (#)、private (-)。

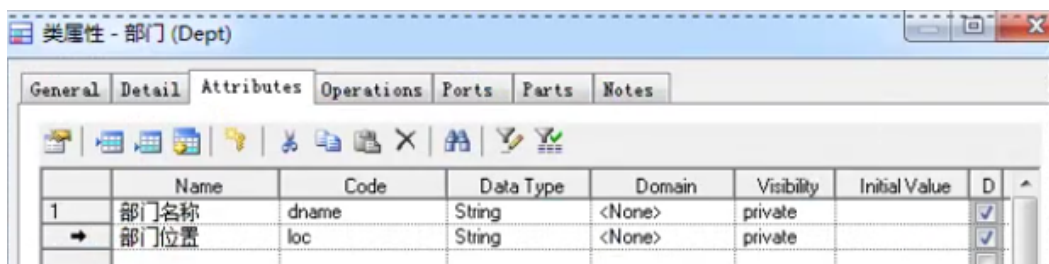
类中的方法采用的格式“访问权限 方法名称()：返回值”结构来描述，一般的方法都是public声明的。

开发者可以随意找到一些小的免费的开发工具，本次使用的是PowerDesigner工具进行设计。

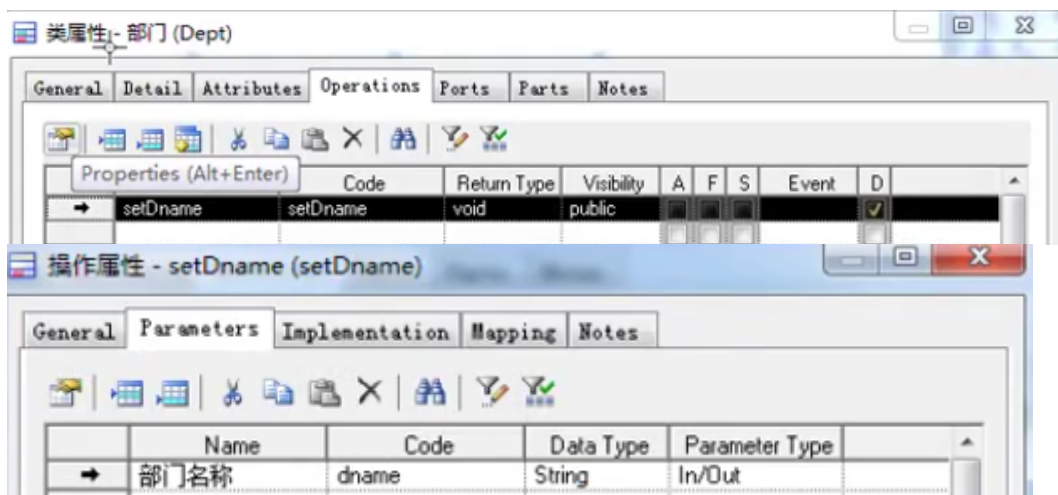




在进行类图定义的时候可以设置类中的属性（Attributes）、类中支持的方法（Operations）。



在类图之中进行方法创建的时候必须先创建方法，而后通过属性才可以设置它的参数



General Detail Attributes Operations Ports Parts Notes

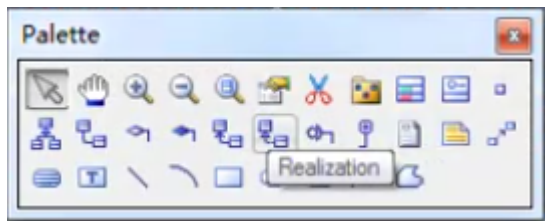
	Name	Code	Return Type	Visibility	A	F	S	Event	D
1	设置部门名称	setName	void	public					✓
2	设置部门位置	setLoc	void	public					✓
3	获取部门名称	getName	String	public					✓
4	获取部门位置	getLoc	String	public					✓
→	Dept	Dept		public			✓		✓
6	Dept	Dept		public					✓

部门

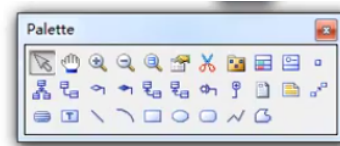
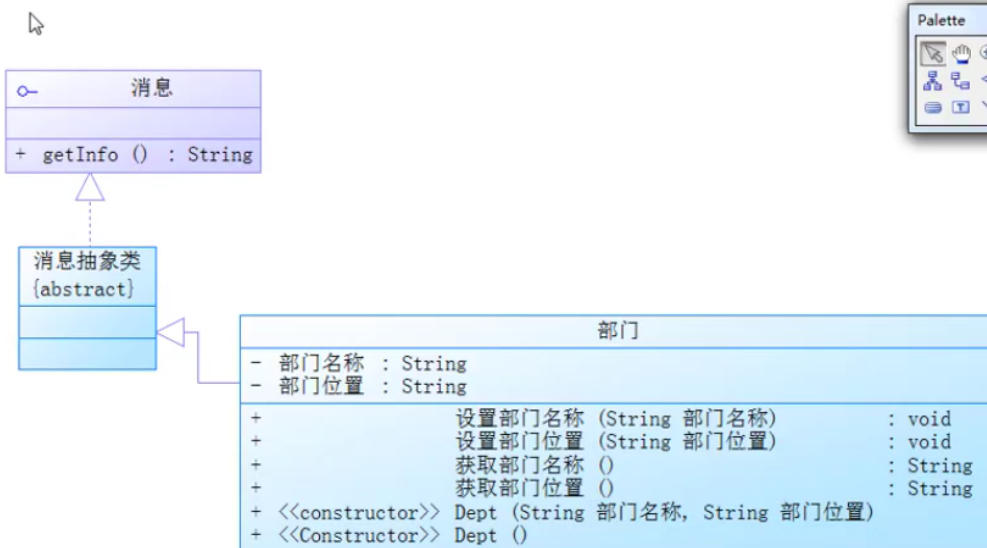
- 部门名称 : String
- 部门位置 : String
- + 设置部门名称 (String 部门名称) : void
- + 设置部门位置 (String 部门位置) : void
- + 获取部门名称 () : String
- + 获取部门位置 () : String
- + <<constructor>> Dept (String 部门名称, String 部门位置)
- + <<Constructor>> Dept ()

Class 部门

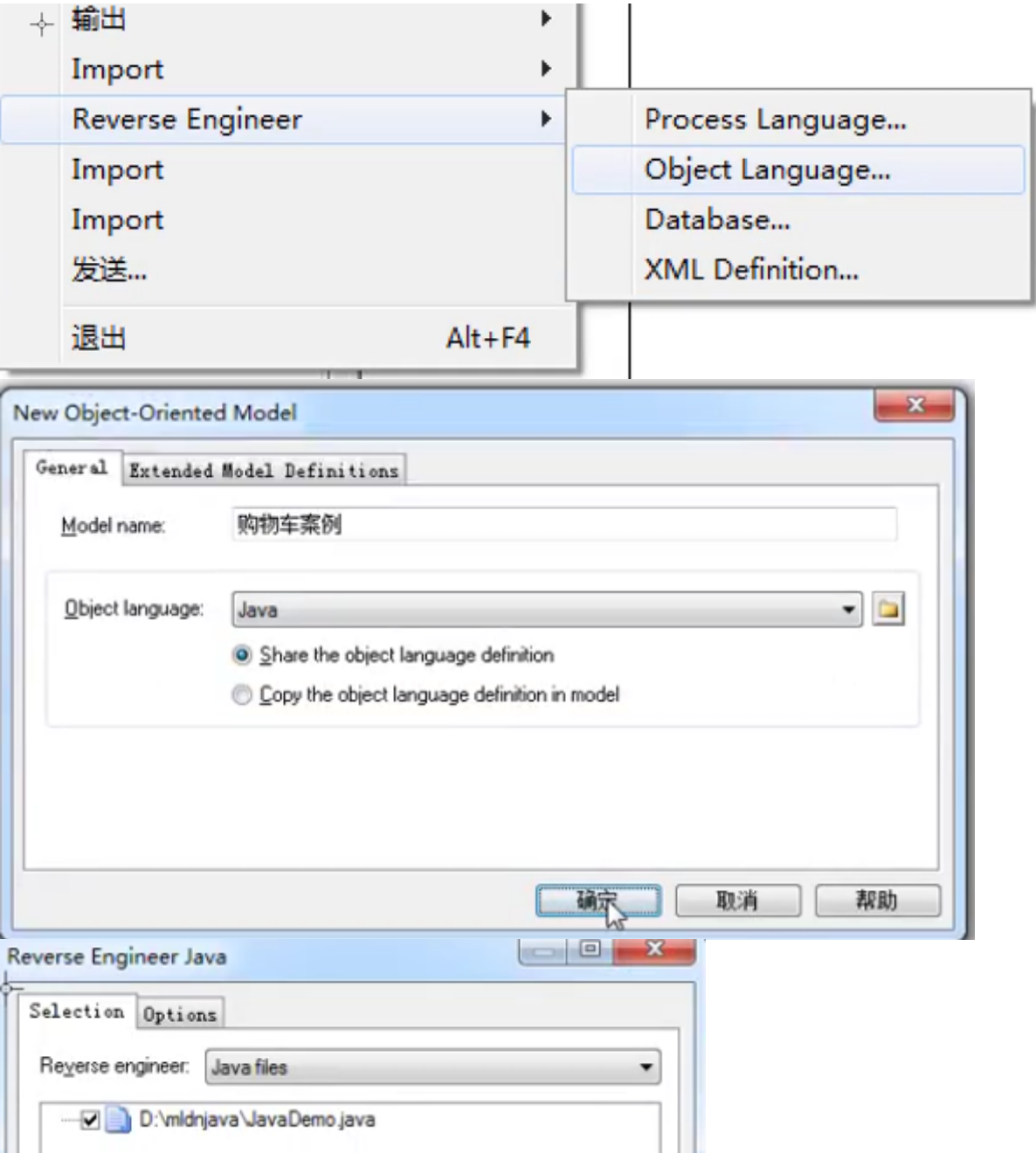
子类实现接口使用的是“三角和虚线”；



类的继承使用的是“三角和实现”；



实际的项目开发之中，如果花费大量的人力进行这样的设计一定是没有意义的（一定是无用功），所以一般会将程序的代码通过转换引擎变为图形显示。



■时序图

时序图主要描述的是你的代码的执行流程，例如，现在有如下一段代码：

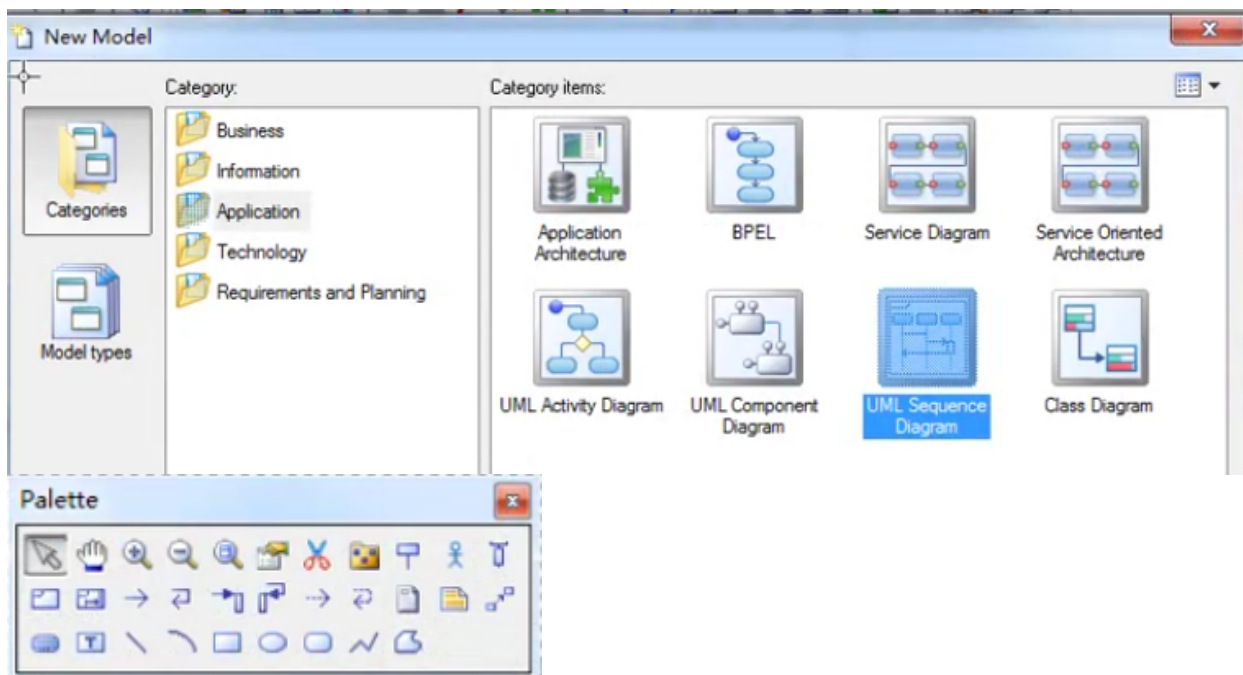
```
interface IMessage{
}
class MessageImpl implements IMessage{
    public MessageImpl(){
        this.connect();//本类的自己方法调用
    }
    public void send(){
```

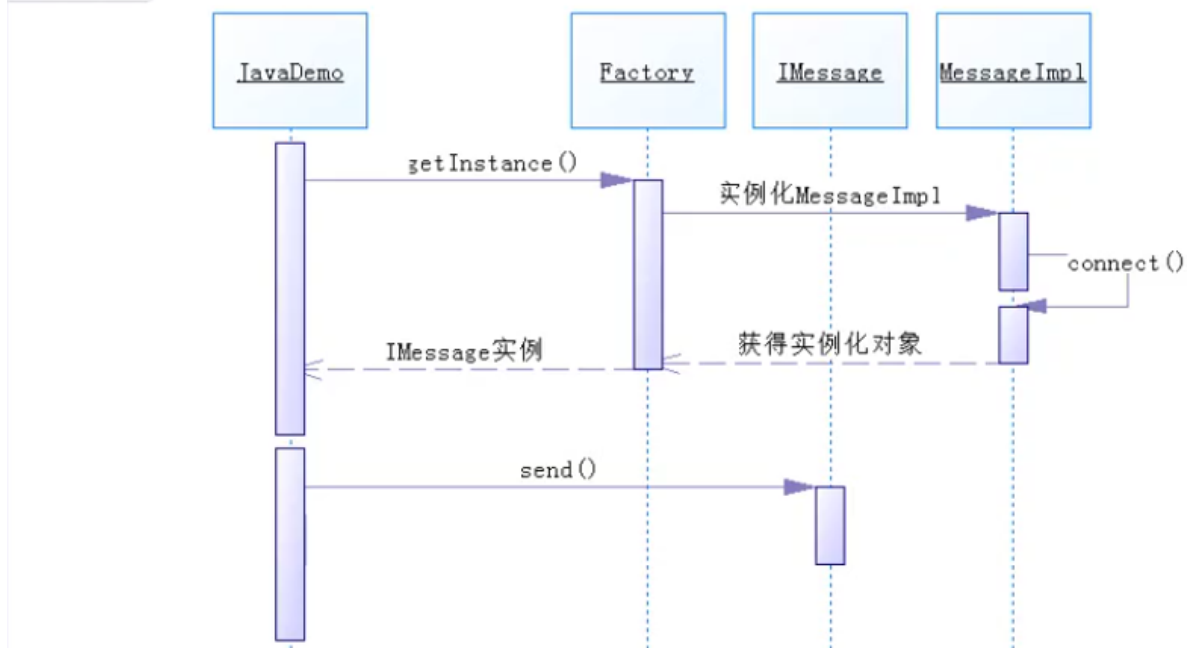
```

        System.out.println("www.mldn.cn");
    }
    public void connect(){
        System.out.println("进行消息发送通道连接");
    }
}
class Factory{
    public static IMessage getInstance(){
        return new MessageImple;
    }
}
public class JavaDemo {
    public static void main(String args[]) {
        IMessage msg = Factory.getInstance();
        msg.send();
    }
}

```

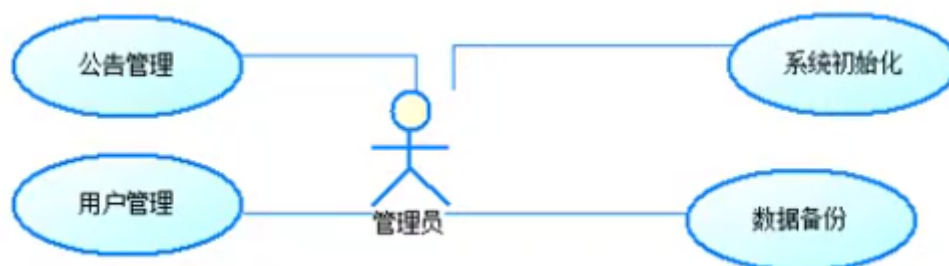
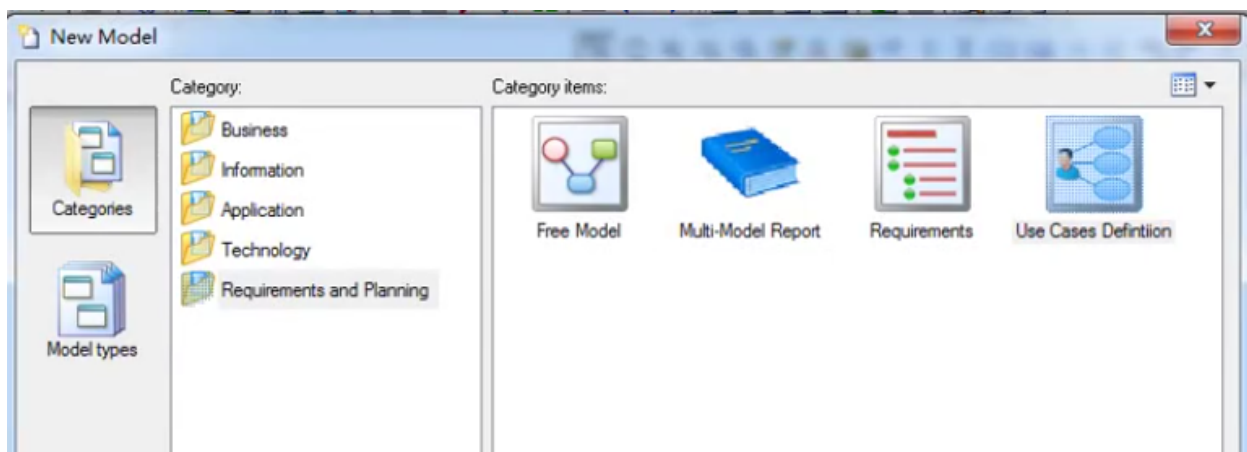
这个时候的程序执行流程可以直接通过时序图来描述





■用例图

用例图描述的是程序的执行分配，例如：现在如果是系统管理员可能拥有系统初始化、系统备份、公告发布的功能，而普通的公告管理员只是负责公告的管理，不应该负责系统的管理，所以就可以在设计书上写上一些用例图加以说明。





一般出现在项目的设计过程之中比较多。