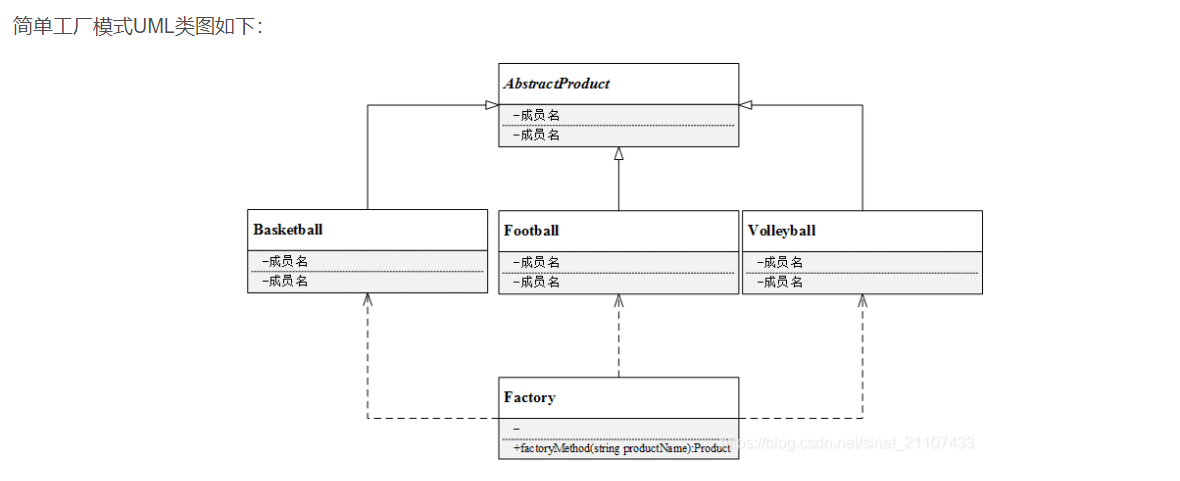
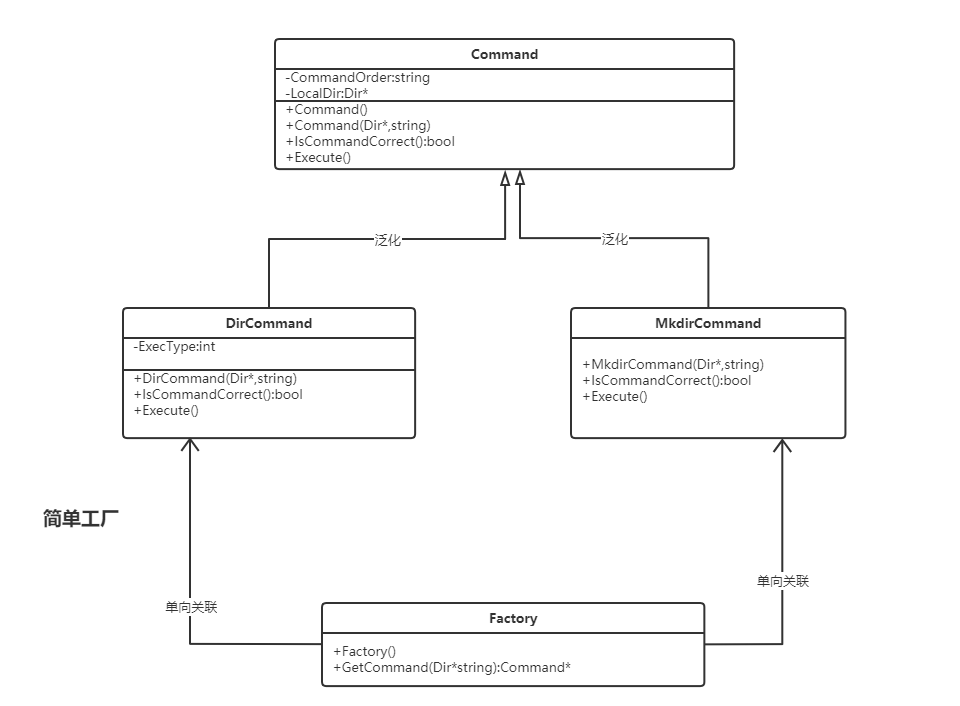


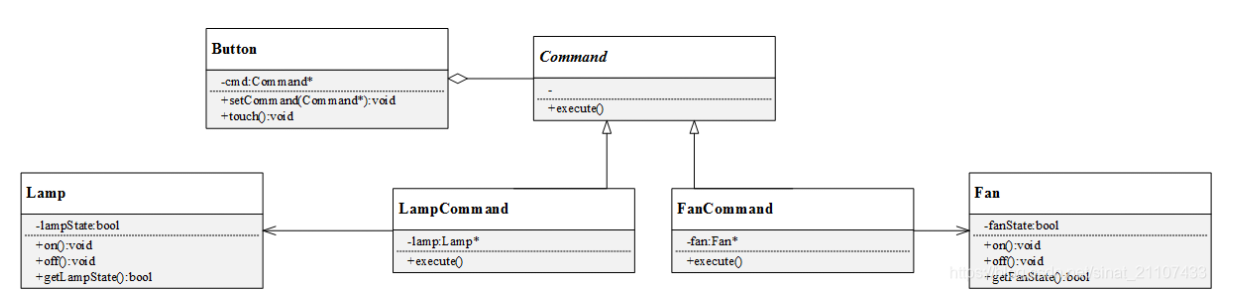
简单工厂模式



使用简单工厂模式处理命令，工厂解析出用户使用的命令类型，根据命令类型生产出具体的产品（命令类）去处理该条命令



命令模式



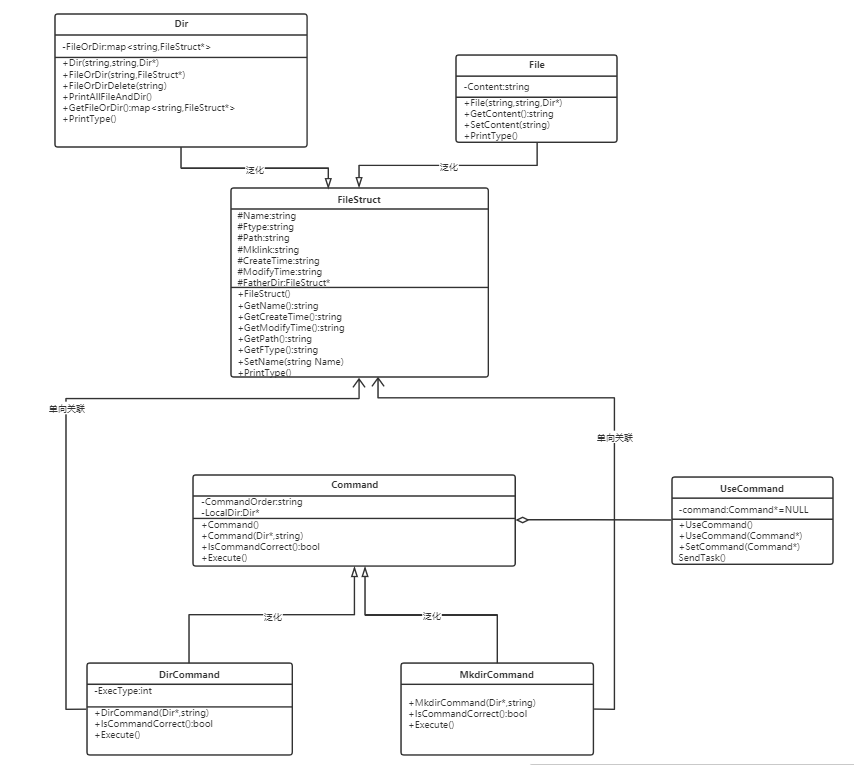
命令模式共有以下几种角色：

Command(抽象命令类):是一个抽象类，声明了用于执行命令的接口execute()

ConcreteCommand(具体命令类):具体的命令类，实现了执行命令的接口execute(),它对应具体的接收者对象，将接收者的动作action()绑定其中,在execute()方法中将调用接收者的动作action()

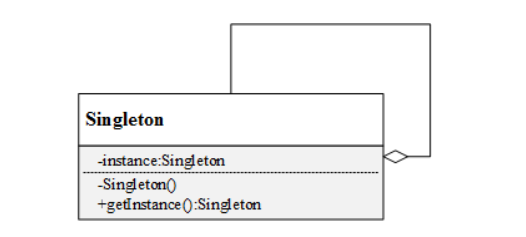
Invoker(调用者):请求的发送者，通过命令对象来执行请求，一个调用者不需要在设计时确定其接收者，所以调用者通过聚合，与命令类产生关联。具体实现中，可以将一个具体命令对象注入调用者中，再通过调用具体命令对象的execute()方法，实现简介请求命令执行者（接收者）的操作。

Receiver(接收者):实现处理请求的具体操作（action）。

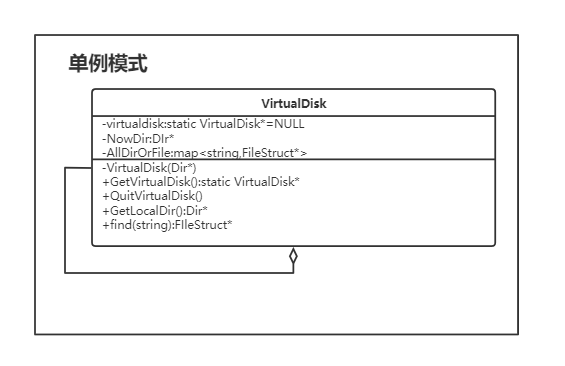


在本系统中，调用者为UseCommand,抽象命令类为Command,具体命令类为DirCommand和MkdirCommand，接收者为FileStruct执行动作。

单例模式

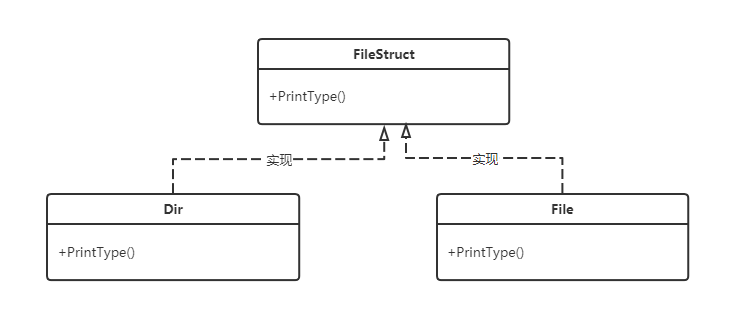


单例模式结构非常简单，其UML图如下所示，只包含一个类，即单例类。为防止创建多个对象，其构造函数必须是私有的（外界不能访问）。另一方面，为了提供一个全局访问点来访问该唯一实例，单例类提供了一个公有方法getInstance来返回该实例

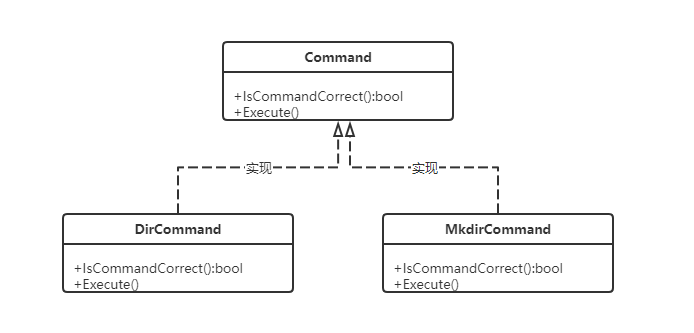


在本系统中，单例模式运用在虚拟磁盘上，虚拟磁盘运行过程中只能有一个，Load操作执行时，正在运行的虚拟磁盘将被释放

接口：



类型转换时必须有虚函数才能进行类型转换dynamic\_cast



IsCommandCorrect()判断该条命令是否正确

Execute()执行该条命令