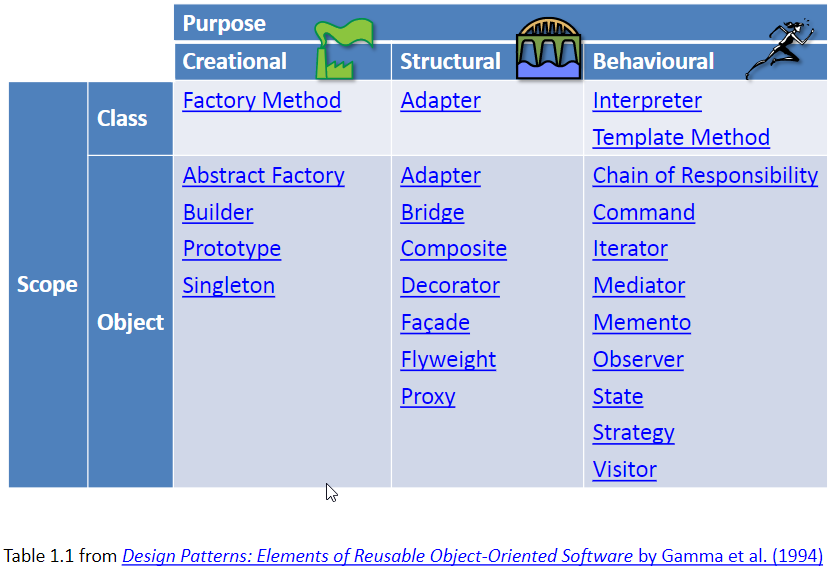
# Module 3 – Design Patterns

## patterns

Types

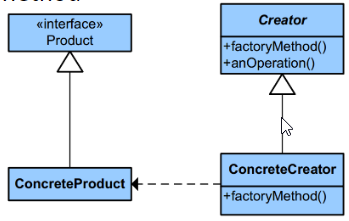
* Creational Patterns
* Structural Patterns
* Behaviour Patterns

## GoF’s 23 Design Patterns

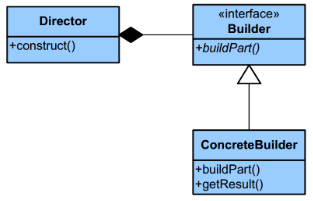


## Creational Patterns

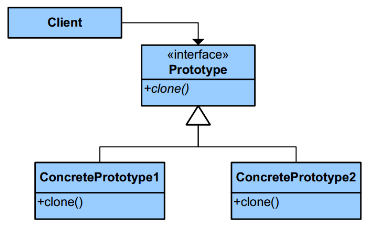
### Abstract Factory: 启用接口



### Builder: 使用builder在不同个数参数使用不同的构造函数

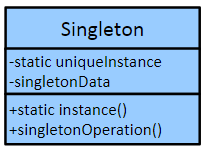


### Prototype: 类似模板，根据prototype复制出不同的实例



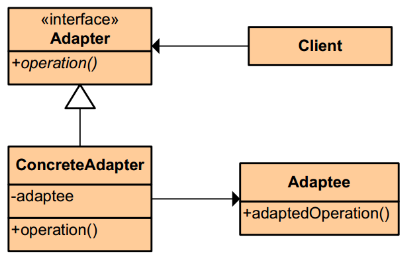
### Singleton: 单例模式的类只能创建一个对象

这个经常用到的，比如解析配置文件的类就是单例的

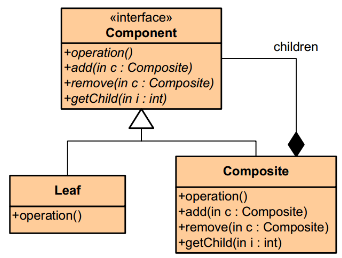


## Structural Patterns

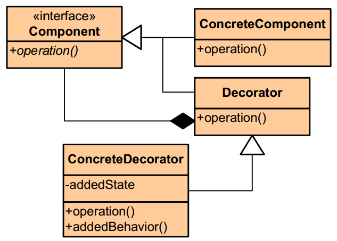
### Adapter: 适配器 用于接入旧系统



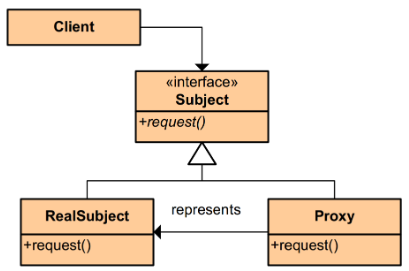
### Composite: 树形结构表示 部分-整体 的层次结构



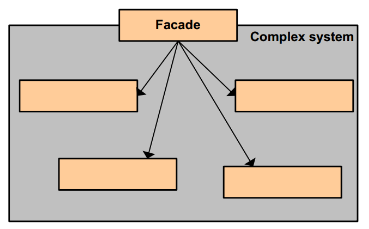
### Decorator=添加新特性



### Proxy 作为接入真正类的中间件



### Façade 单点接入系统

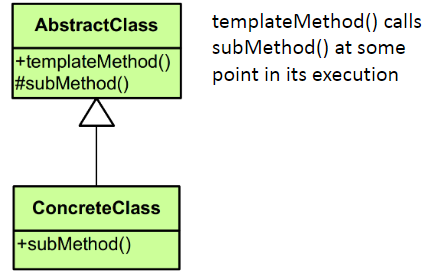


### Bridge 脱离真正实现

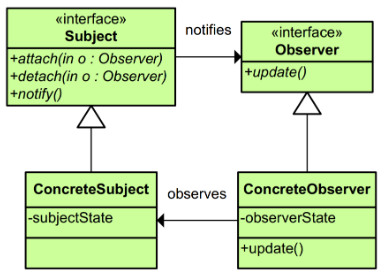
### Flyweight 分享共同数据

## Behavioural Patterns 相同代码应对用户的不同行为

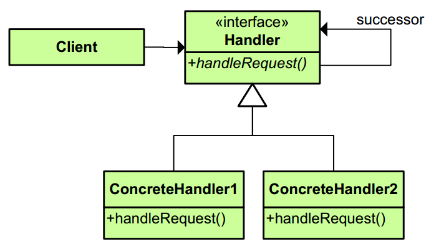
### Template Method= 一个模板根据不同需求而customize



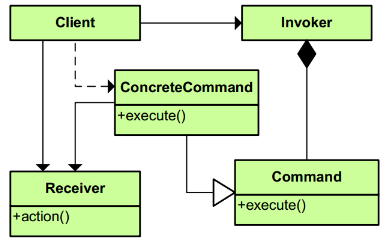
### Observer: 监听调用



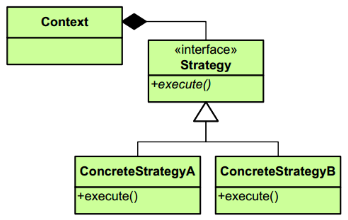
## Chain of Responsibility 不同handler应对不同的请求



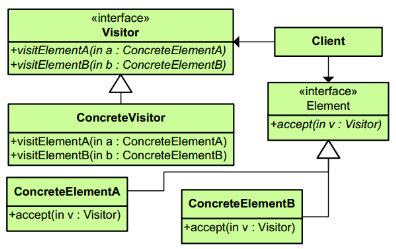
## Command: 命令接入实际执行



## Strategy: 动态选择不同的执行



## Visitor 根据不同的参数访问不同的程序



## Interpreter= 解释器

## Iterator= 循环执行

## Mediator= 通过中接器到不同的执行

## Memento=保存实例的中间状态

## State=根据状态做出不同的执行