



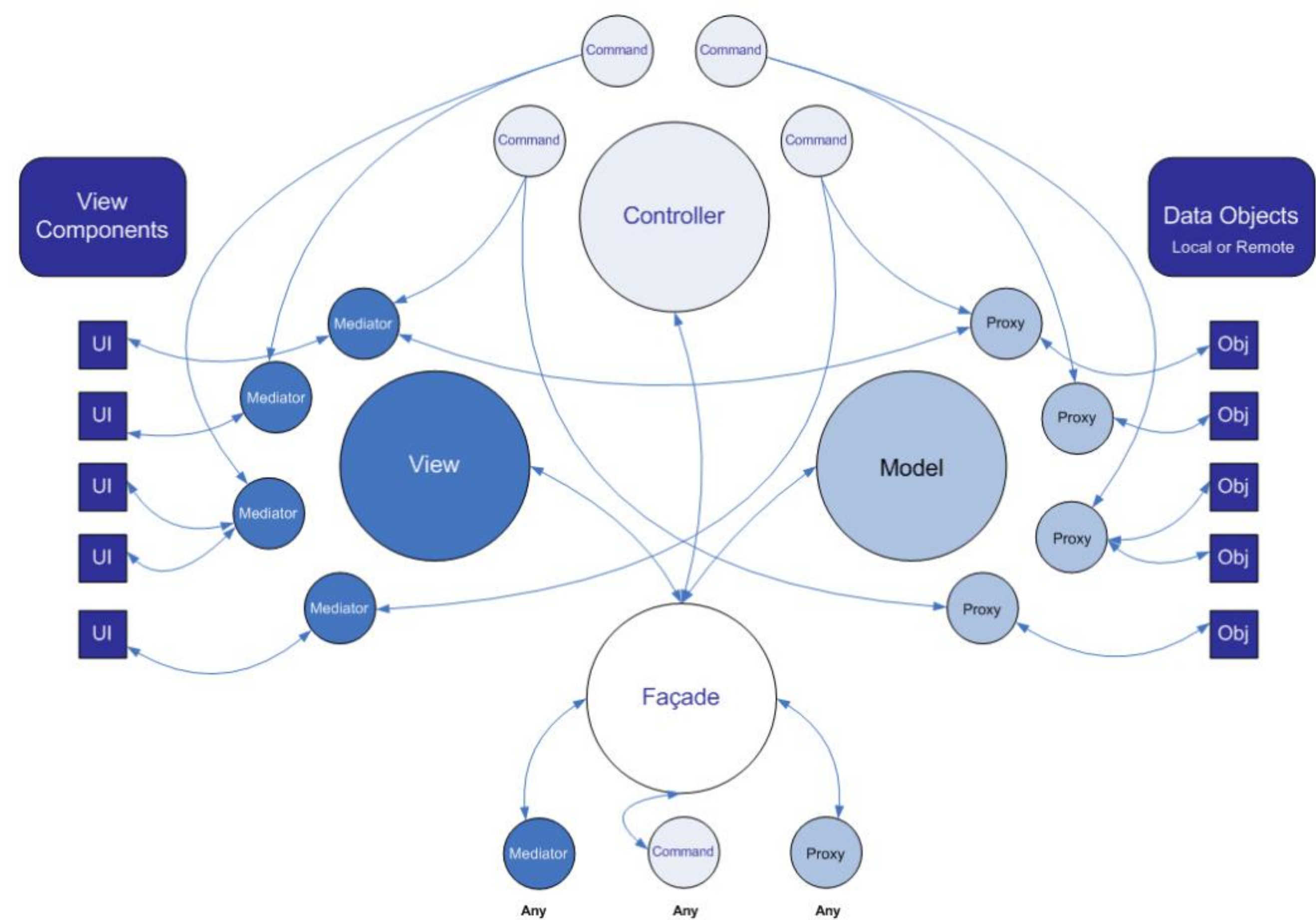
唐老狮系列教程

PureMVC总结



唐老狮系列教程-PureMVC基本概念

PureMVC的基本结构





唐老狮系列教程-PureMVC基本概念

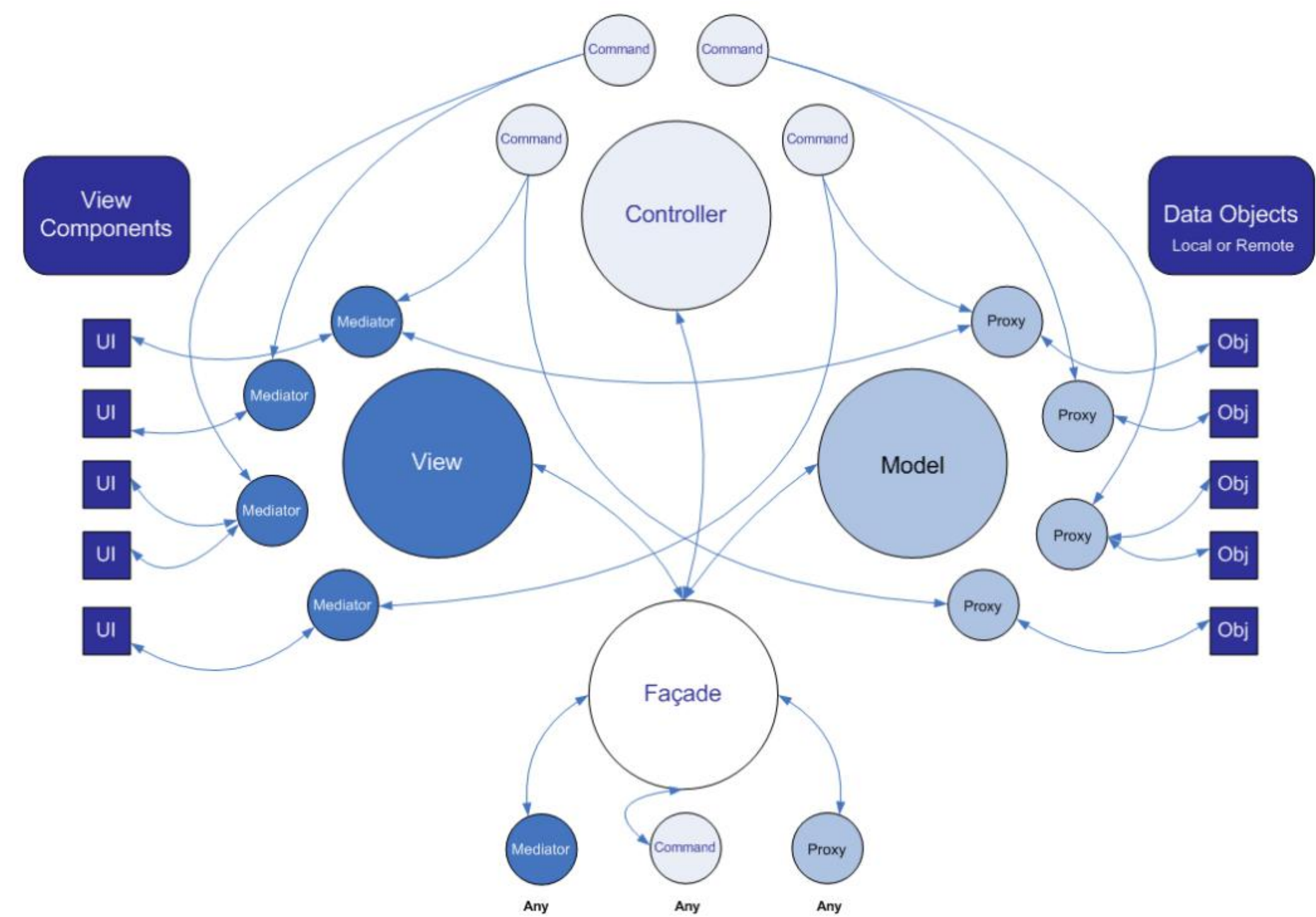
Facade

```
public class GameFacade : Facade
{
    //1.继承PureMVC中Facade脚本

    //2.为了方便我们使用Facade 需要自己写一个单例模式的属性
    4 个引用
    public static GameFacade Instance

    /// <summary>
    /// 3.初始化 控制层相关的内容
    /// </summary>
    3 个引用
    protected override void InitializeController()

    //4.一定是有一个启动函数的
    1 个引用
    public void StartUp()
}
```





唐老狮系列教程-PureMVC基本概念

Proxy

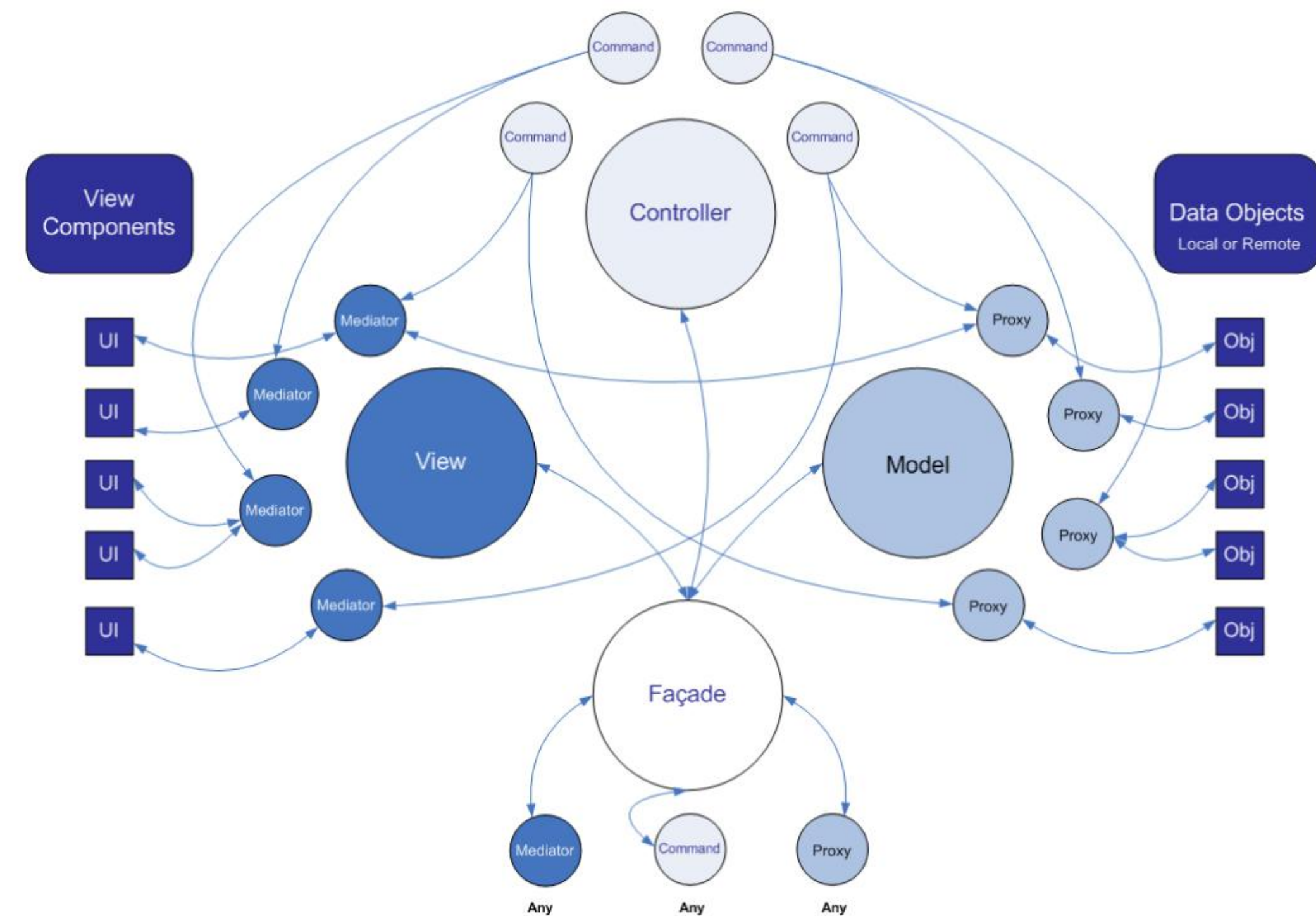
```
public class PlayerProxy : Proxy
{
    public new const string NAME = "PlayerProxy";
    //1.继承Proxy父类
    //2.写我们的构造函数

    //写构造函数
    //重要点
    //1.代理的名字!!!!
    //2.代理相关的数据!!!!

    1 个引用
    public PlayerProxy()...

    1 个引用
    public void LevUp()...

    1 个引用
    public void SaveData()...
}
```





唐老狮系列教程-PureMVC基本概念

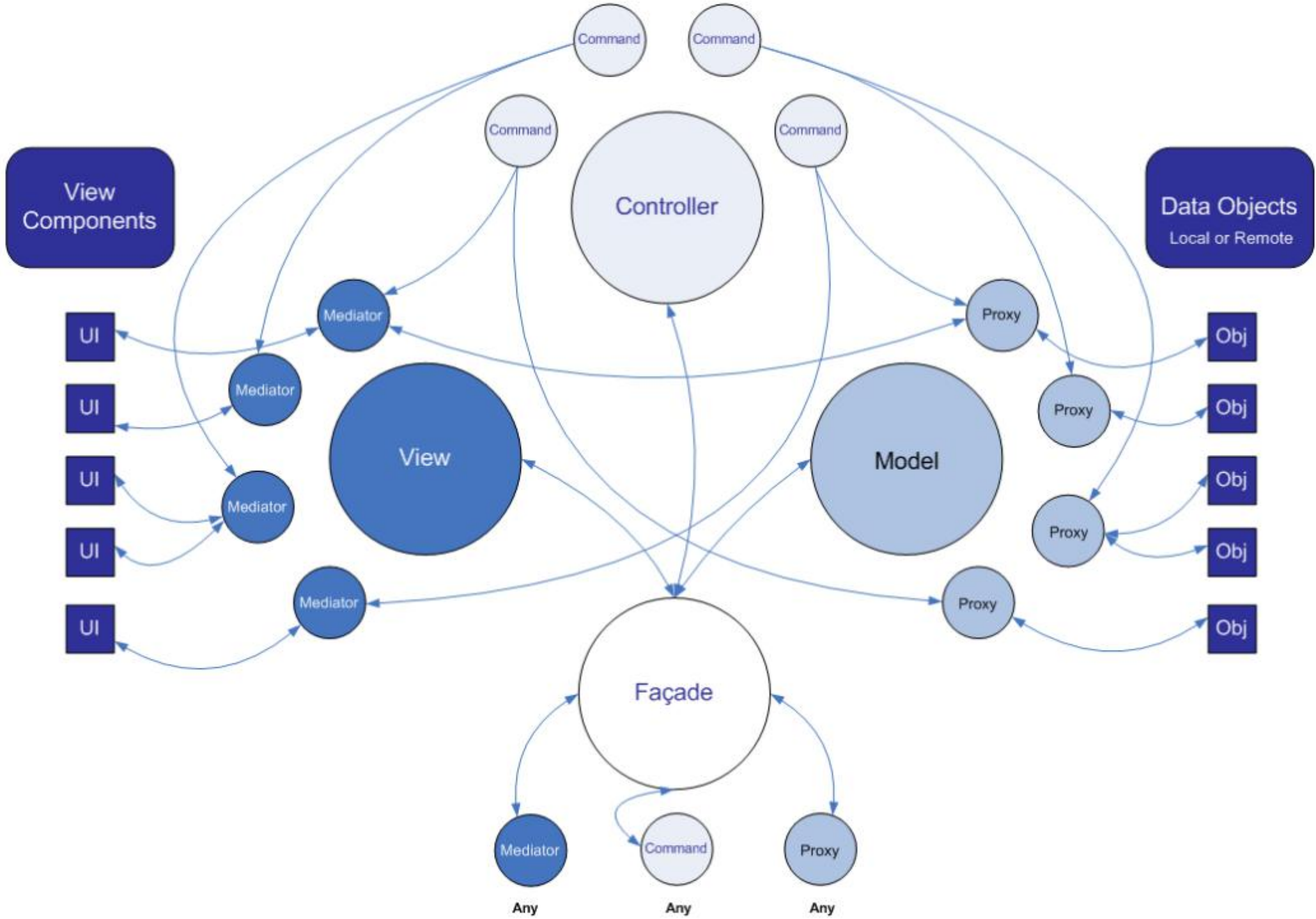
Mediator

```
public class NewRoleViewMediator : Mediator
{
    public static new string NAME = "NewRoleViewMediator";
    //套路写法
    //1.继承PureMVC中的Mediator脚本
    //2.写构造函数
    1 个引用
    public NewRoleViewMediator()...

    //3.重写监听通知的方法
    5 个引用
    public override string[] ListNotificationInterests()...

    //4.重写处理通知的方法
    4 个引用
    public override void HandleNotifcation(INotification notification)...

    1 个引用
    public void SetView(NewRoleView view)...
```





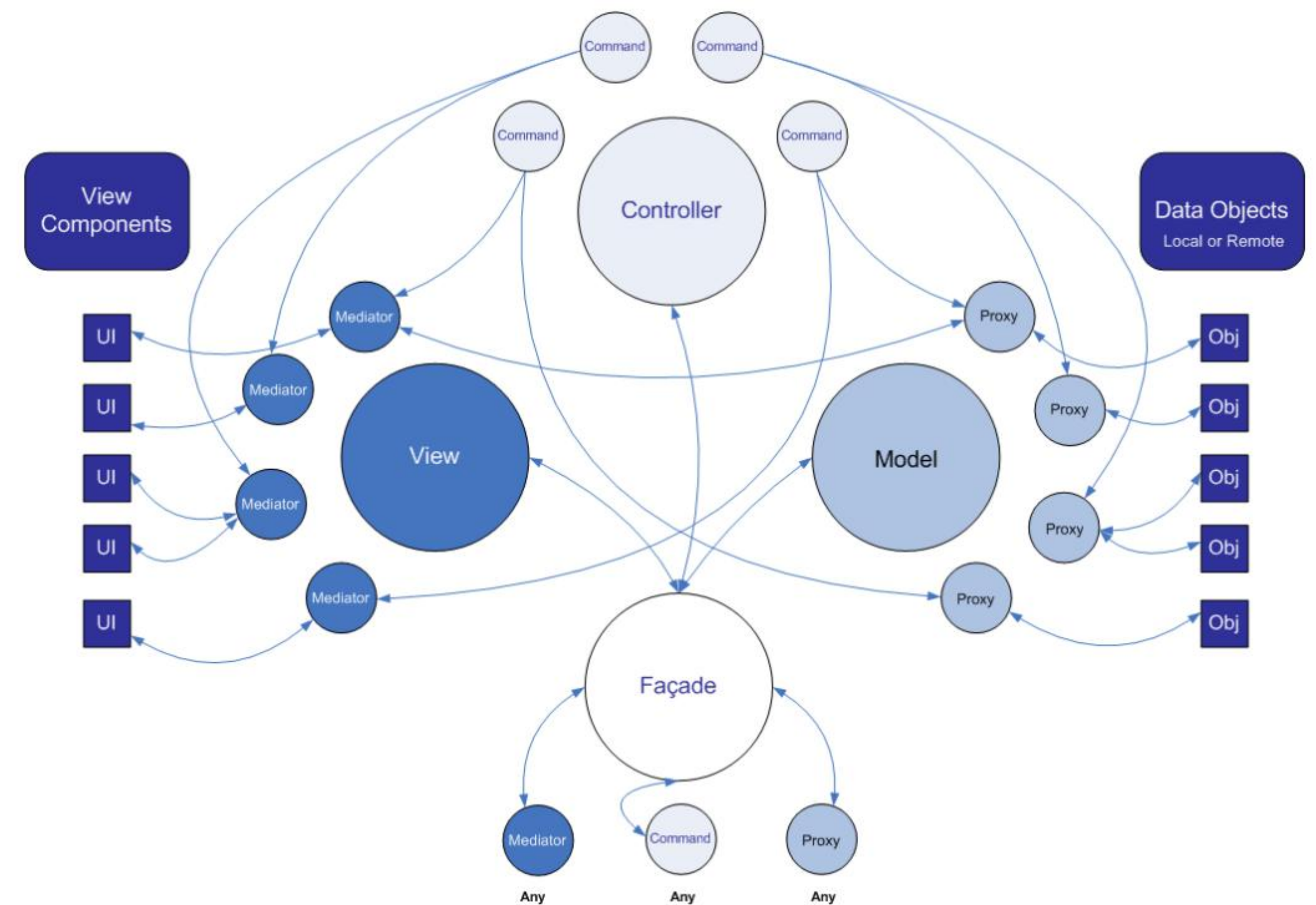
唐老狮系列教程-PureMVC基本概念

Command

1 个引用

```
public class ShowPanelCommand : SimpleCommand
{
    12 个引用
    public override void Execute(INotification notification) ...
}
```

//INotification 对象 里面包含两个对我们来说 重要的参数
//1.通知名 我们根据这个名字 来做对应的处理
//2.通知包含的信息





唐老狮系列教程-PureMVC基本概念

总结

套路写法

先数据

后界面

再用命令做串联

Facade 判断、注册和获取



唐老狮系列教程

Thank
感谢您的聆听

WELCOME
TO THE
UNITY
SPECIALTY COURSE
STUDY

版权所有：唐老狮 tpandme@163.com