



消息传递中心

UI界面框架设计 讲师:刘国柱

生活中的消息传递

- ▶ 生活中的“消息传递”自古至今一直在快速发展中。



Unity目前提供的消息传递方式

- ▶ 基于Unity技术的游戏与项目开发，目前提供的消息传递方式种类少，且耦合性很高。
 - 1: 脚本组件公共方法、字段的相互调用。
例如: `GetComponent<Scripts>().TestMethod();`
 - 2: SendMessage 技术。
 - 3: 单例模式数据传递。

消息传递中心

- ▶ 开发一种低耦合，无需考虑被传递对象（脚本名称、组件名称）的技术非常有价值。
- ▶ “消息传递中心”：
基于观察者模式，利用委托与事件的基本机制原理，进一步封装重构的技术实现。

消息传递中心原理

- 定义MessageCenter

基本原理：

```
/// 委托：消息传递
public delegate void DelMessageDelivery(KeyValuesUpdate kv);

///<所要监听的类型，监听到以后要执行的委托>
public static Dictionary<string, DelMessageDelivery> dicMessages = new
Dictionary<string, DelMessageDelivery>();
```

```
//键值对
public class KeyValuesUpdate
{
    //键
    private string _Key;
    //值
    private object _Values;
}
```


客户程序使用演示



客户程序使用的扩展

- 窗体基类(BaseUIForms) 中对于“消息传递中心”类常用方法的封装。
 - 发送消息 SendMessage()
 - 接收消息 ReceiveMessage()
- 客户程序消息传递多组数据的演示。
- 客户程序建立“系统常量”类，方便程序复用与集中化管理

A close-up of a character wearing a red hooded cloak, looking intensely at the viewer. The character's face is partially obscured by the hood, with only their eyes and a small portion of their nose and mouth visible. They are holding a glowing blue torch in their right hand. The background is dark and atmospheric, with a bright blue light source creating a lens flare effect.

谢谢大家！