



链滴

U3D 界面框架新旧 UI 系统

作者: [xu365082218](#)

原文链接: <https://ld246.com/article/1507563934738>

来源网站: 链滴

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

界面是所有游戏工程不可缺少的部分，而大部分项目必须会含有一套控制界面的基础类

NGUI的UICamera和EventSystem应该是2个不同的玩意，但是他们都会使用raycast得到一个点击的表，然后由近到远决定处理方式，

一般UICamera挂在UI相机上，然后设置其事件响应层以及相机的剔除遮罩，raycast得到碰撞列表后像最近的物件发送事件和参数，如果该物件上有其他不同的方法实现

则会报异常，就是通常新UI系统和NGUI系统一起用的时候，会出现一些乱七八糟说参数不匹配的异常

类似NGUI的UICamera向一个对象发送信息 `gameObject.SendMessage("OnSelect", true)`

假如这个gameObject上挂了一个Button这个是系统内置的，那么这个Button应该是实现了一个非Bool类型参数的OnSelect(BaseEventData data)

这个时候，就会出错，说函数参数不匹配，这样来看UICamera的做法，`gameObject.SendMessage`如此类的实际上弊病很大，特别是一些Component具有相同函数名，但是参数原型不匹配时

实际上2者大部分应该是能转换的

系统UI内的图像原件，都会有raycastTarget勾选框，也就是说，贴一张图，加一个Button原件，就在点到该图像时，进入到Button原件的自定义事件内，也就是OnClick

要想有更详细的事件类似UICamera发送的onpress ondrag onselect等，则需要自己仿造UICamera逻辑，来处理IPointerDownHandler, IPointerUpHandler, IPointerExitHandler接口

类似

```
public class GameButton : MonoBehaviour, IPointerDownHandler, IPointerUpHandler, IPointerExitHandler
{
    void Start(){}
    void Update(){}
    public void OnPointerDown(PointerEventData eventData){}
    public void OnPointerUp(PointerEventData eventData){}
    public void OnPointerExit(PointerEventData eventData){}
}
```

有一点需要注意，要让含有raycastTarget勾选框的原件响应点击事件，还需要查看其与UI相机的朝向，有次测试无论如何点一个Image都无法触发事件，原来是他的y轴旋转填了180，后来置0后就可以发了

点击鼠标后raycast触发碰撞的列表，可以在场景树组织中选中 EventSystem物件查看

关于# NGUI的 (UICamera) 事件处理 有篇文章

<https://www.zhuzhehao.cn/?p=184>