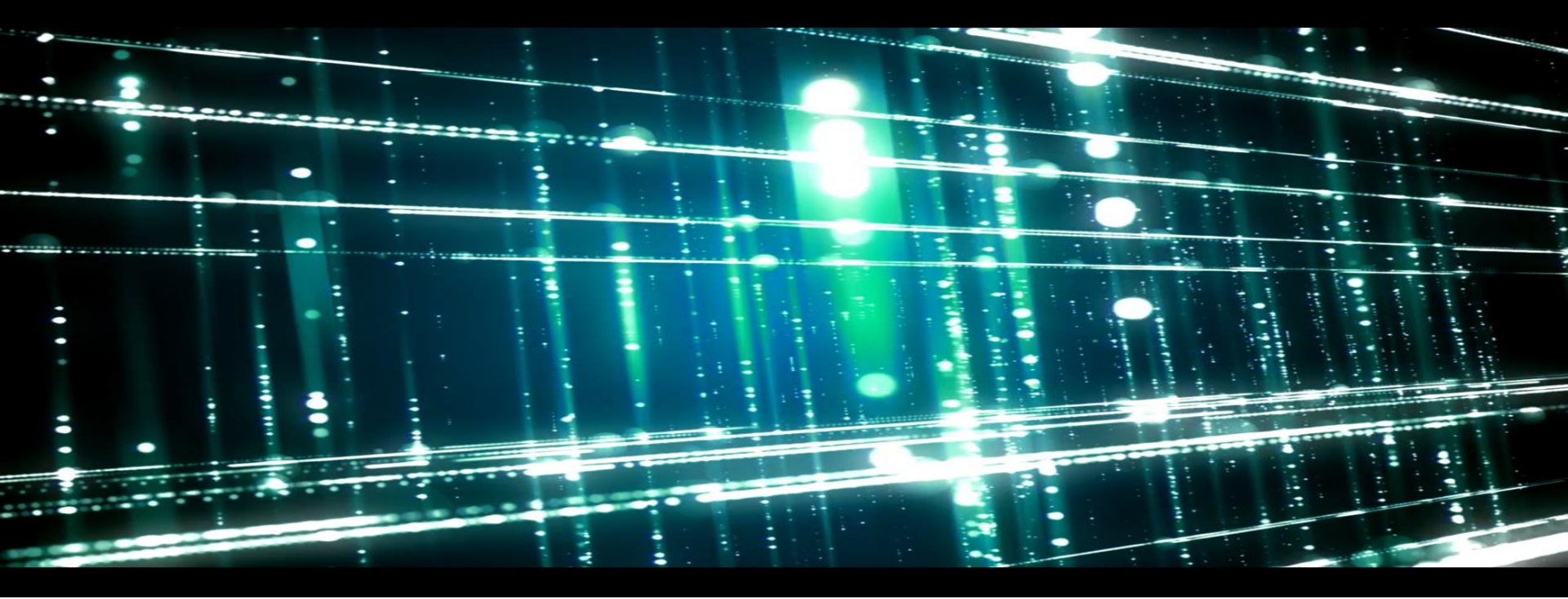

窗体层级管理

UI界面框架设计　讲师:刘国柱

窗体层级管理-栈数据结构

- 什么是“栈”数据结构？

  是一种“先进后出”的数据结构，是一种常用算法。

  生活中的“汉诺塔”游戏、“摞烧饼”、“盘子堆”都是一种典型的“栈”结构。

窗体层级管理-栈数据结构

- C#语言中提供 Stack<T> 泛型集合，来直接实现这种结构。

常用属性与方法：

- Count 属性　查询栈内元素数量
- Push()　压栈
- Pop()　出栈
- Peek()　查询栈顶元素
- GetEnumerator（）　遍历栈中所有元素

演示：典型Demo示例。

窗体层级管理-反向切换

- 开发“UI管理器”的“栈”数据结构，维护窗体的层级结构。
- 定义Stack<BaseUIForms> 类型字段。
- 显示UI窗体 ShowUIForms() 方法中

  1>“反向切换”属性的窗体，定义“压栈”方法
- 关闭(或返回上一个UI)窗体方法中

  1>“普通”显示属性的窗体，定义关闭方法。

  2> 对于“反向切换”属性的窗体，定义返回上一个窗体的方法(即：关闭)。

窗体层级管理-隐藏其他

- 显示UI窗体 ShowUIForms() 方法中

  1>“反向切换”属性窗体，定义“压栈”方法

  2> **“隐藏其他”属性窗体，定义显示业务逻辑方法**

- 关闭(或返回上一个UI)窗体方法中

  1>“普通”显示属性的窗体，定义关闭方法。

  2> 对于“反向切换”属性的窗体，定义返回上一个窗体的方法。（即：关闭）。

  3> **“隐藏其他”属性窗体，定义关闭逻辑方法**

窗体层级管理-清空栈集合(1)

- 在多个UI业务窗体中，有时候需要客户端程序主动清空“栈集合”中的当前数据，防止业务逻辑混乱。

  例如： RPG中的“商场系统”、“背包系统”、“任务系统”等。

窗体层级管理-清空栈集合(2)

- 具体代码实现：

  1：在UIType 类中，定义是否需要“清空反向切换”的字段（或者属性）。

  2：在UI管理器脚本中，关于显示UI窗体的方法中，加入判断清空栈中数据的业务逻辑即可。

窗体层级管理-客户程序调用测试

定义如下窗体编写代码测试UI框架功能：

- 登陆窗体：

  注意事项：所有窗体脚本都要继承BaseUIForms

  定义本窗体的类型（位置、显示、透明度三大属性），不写则采用默认数值。

  注册窗体按钮事件。

- 选择英雄窗体：
- 主城窗体：
- 商城窗体：
- 商品信息窗体：

重构-定义帮助类

- 程序重构发现，在UI框架内部与客户调用程序中都存在一些反复被使用的技术。
  - 对于层级视图的节点查找。
    (扩展方法)
  - 获取子节点(物体)的脚本
  - 给子节点(物体)添加脚本
  - 给子节点(物体)添加父对象

```
    _TraNormal = _TraCanvasTransfrom.Find("Normal");
    _TraFixed = _TraCanvasTransfrom.Find("Fixed");
    _TraPopUp = _TraCanvasTransfrom.Find("PopUp");
    _TraUIScripts = _TraCanvasTransfrom.Find("_ScriptMgr");
Transform searchTrans = _TraLogonFormsTransfrom.FindChild("BG/Btn_OK");
```

重构-定义窗体基类方法

- 程序重构发现，在客户程序（UI框架）调用中，会反复出现一些常用的定义方式，例如：

- 按钮的事件监听与注册方法
- 打开指定窗体
- 关闭指定窗体
- 发送消息
- 显示语言信息


谢谢大家！
刘老师讲Unity系列教程