



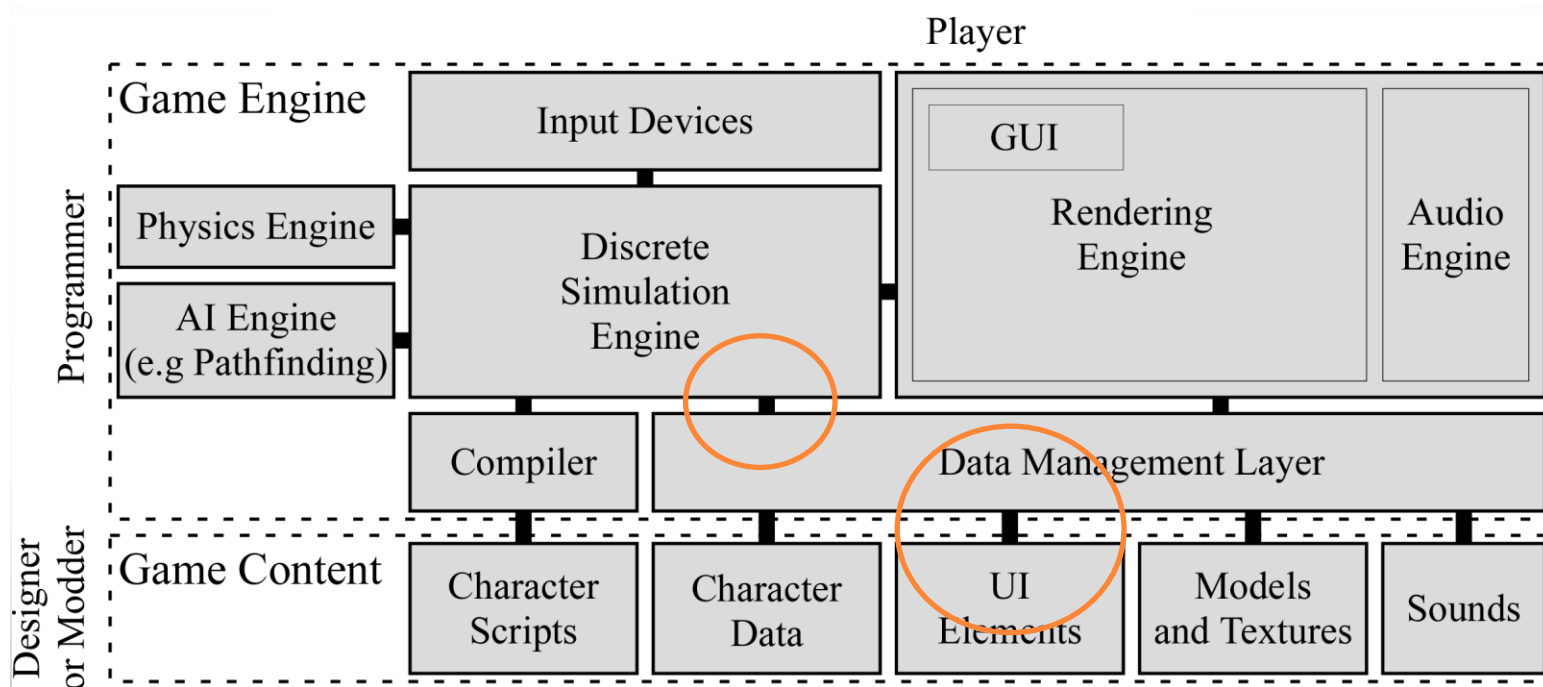
# INTRODUCTION TO COMPUTER 3D GAME DEVELOPMENT

Unity User Interface (UI)

潘茂林, [panml@mail.sysu.edu.cn](mailto:panml@mail.sysu.edu.cn)

中山大学·软件学院

# 游戏引擎架构



# 目录

- Unity UI 系统基础
  - 简介与画布
  - UI 排版
  - UI 部件
- 案例研究
  - Screen Space 菜单
  - 3D 菜单
- 面向对象的编程思考
  - 对象方法扩展
  - 对象扩展与 iTween



# UNITY UI系统 (UI)

## (1) 简介

### ○ Unity UI 系统

- IMGUI

- (<https://docs.unity3d.com/Manual/GUIScriptingGuide.html>)
- 方便的内置显示与调试工具
- 用于 Unity 编辑环境的扩展

- Unity UI

- 方便的UI



# UNITY UI系统 (UI)

## (2) 画布 (CANVAS)

### ○ 画布

- 画布是所有UI元素（对象）的容器
- 如果你创建一个UI元素，如创建 `GameObject > UI > Image` 则系统会创建一个 `Canvas` 作为它的父对象
- 如果在 `Canvas` 中创建 UI 对象，则它成为子对象

### ○ UI 元素渲染顺序

- 按“广度优先”顺序渲染
- 兄弟 UI，按在树上的自然顺序渲染
  - 放置一些相互重叠的按钮，观察顺序



# UNITY UI系统 (UI)

## (2) 画布 (CANVAS)

### ○ Render Mode

- Screen Space – Overlay

This render mode places UI elements on the screen rendered on top of the scene. If the screen is resized or changes resolution, the **Canvas will automatically change size** to match this.

- Screen Space – Camera

This is similar to **Screen Space – Overlay**

- World Space

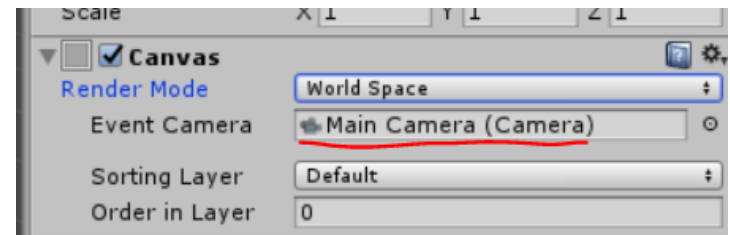
In this render mode, the Canvas will behave as any other object in the scene. **The size of the Canvas can be set manually** using its Rect Transform, and UI elements will render in front of or behind other objects in the scene based on 3D placement.



# UNITY UI系统 (UI)

## 课堂实验一：测试画布 (CANVAS)

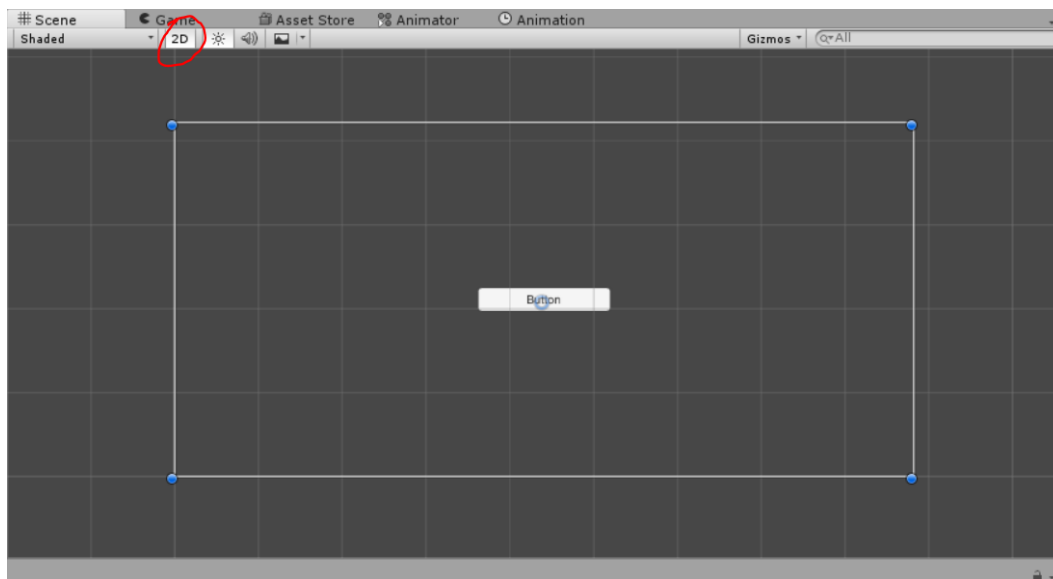
- 创建一个 Button 对象，你看到：
  - Canvas 对象，它有子对象 Button
  - EventSystem 对象
- 修改 Canvas 对象 Canvas 组件的 rendermode
  - 你应该看到一样的结果。注意检查：
    - button 位置 (0, 0, 0)
    - camera 位置 (0, 1, -10)
    - 在camera模式，拖动主相机到这里。
    - canvas 位置 (0, 1, 约90)
- 结果：
  - Screen Space – Overlay: 合适做屏幕菜单
  - World Space: 合适需要相机运动，或 canvas 运动的 2D 游戏，如 卷轴



# UNITY UI系统 (UI)

## (3) UI元素排版

- 进入 2D 编辑模式
  - 缩小放大 canvas，直到看到完整 canvas
  - 用 shift+上下左右，移动到中央，放大比例
  - **注意：** World Space 模式用鼠标拖动会改变 canvas位置
  - 建议开始用 Screen Space – Overlay 模式





# UNITY UI系统 (UI)

## (3) UI元素排版

### UI元素排版

- 组件 RectTransform
- The Rect Tool

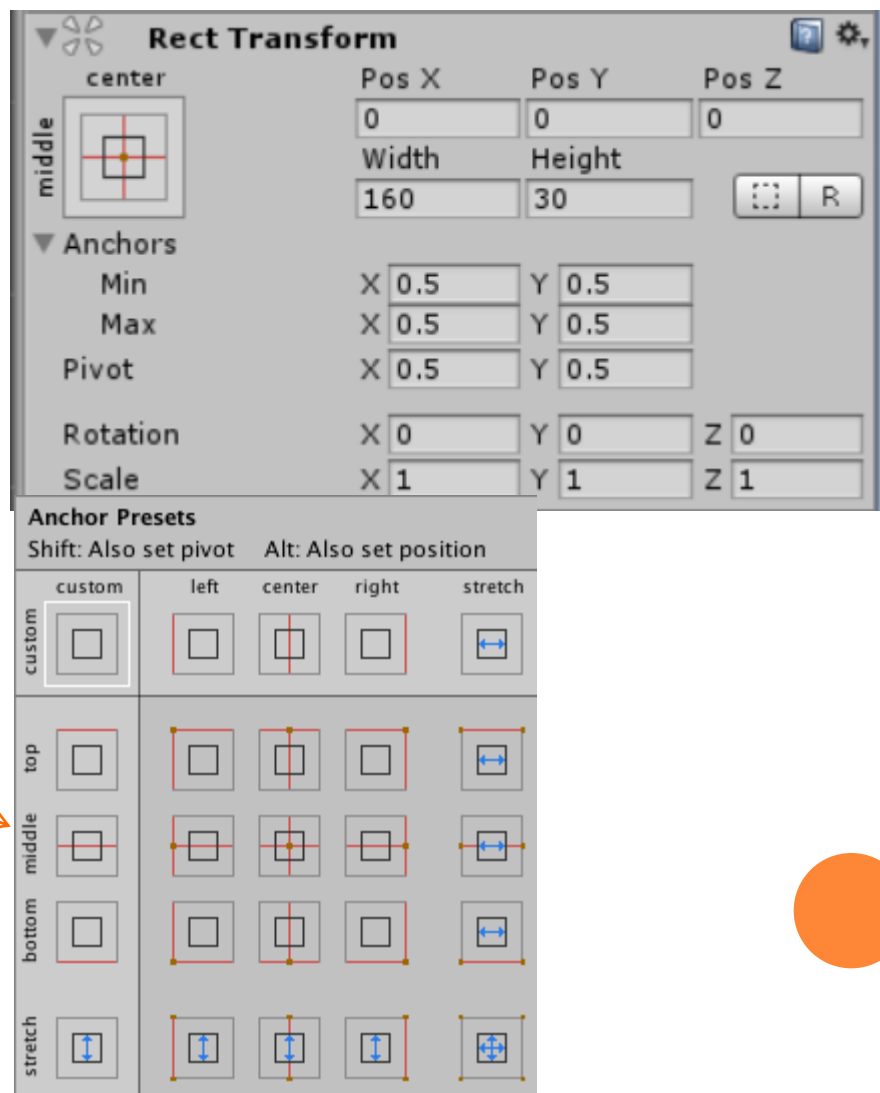


- 改变大小，旋转等



- 改变旋转点

- 锚点预设工具



# UNITY UI系统 (UI)

## (3) UI元素排版- 大小与旋转

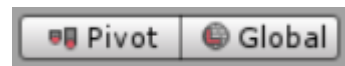
### ○ 控制 button大小

- 使用 Size 工具
- 使用 Scale 工具
- 比较它们的不同



### ○ 控制 button 旋转

- 控制 pivot (原点) 位置
- 然后旋转 button



# UNITY UI系统 (UI)

## (3) UI元素排版- ANCHORS

### ○ 锚点

- 停靠位置或比例
- 讲述了父对象和子对象的关系

### ○ 实验

- 创建按钮子对象,叫 sub button
- 修改 sub button 的位置
- 使用工具修改锚点
- 放大缩小Button ... ..
- 更多



<https://docs.unity3d.com/Manual/UIBasicLayout.html>

# UNITY UI系统 (UI)

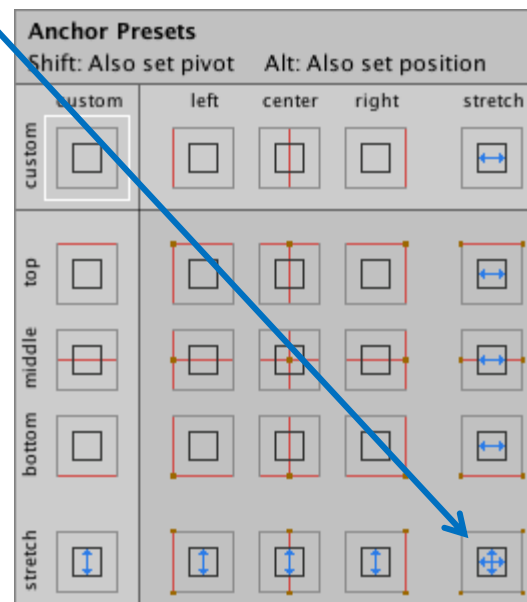
## (3) UI元素排版- ANCHORS

### ○ 单点停靠与比例停靠

- 当子对象顶靠父对象范围内则产生同时缩放

### ○ 实验:

- 如何实现父对象与子对象等比缩放?



# UNITY UI系统 (UI)

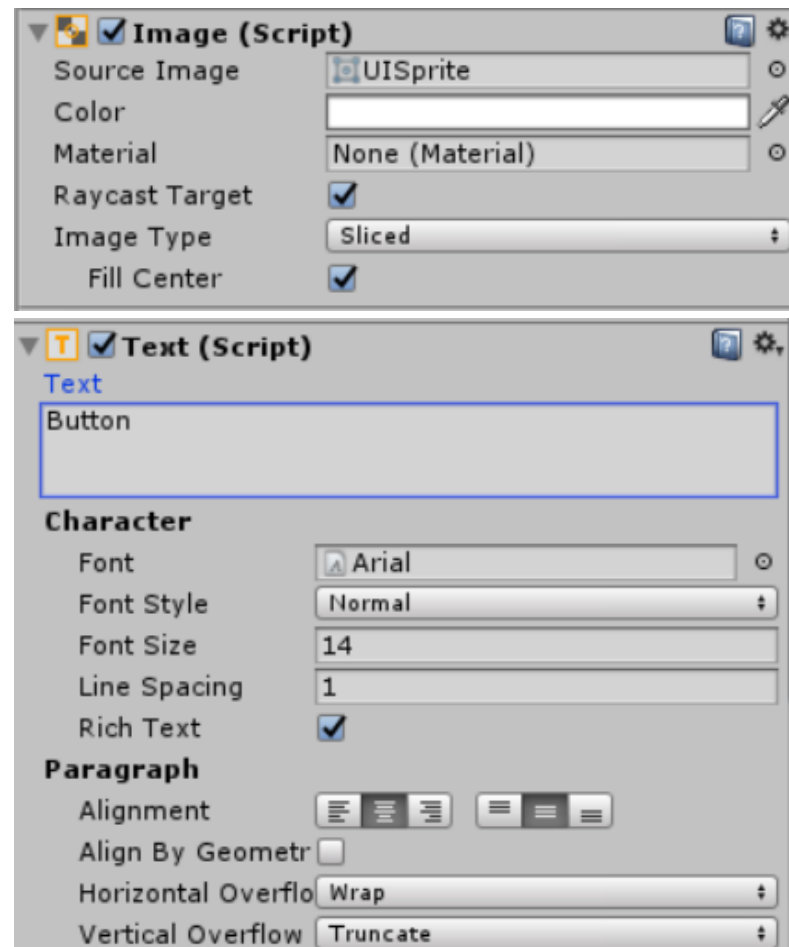
## (4) 可视化部件

### ○ Image

- 选择其他 Source Image 观察效果
- 选择 Image Type 观察效果

### ○ Text

- 修改 TEXT, 汉字支持吗?



# UNITY UI系统 (UI)

## (4) 可视化部件

### ○ Raw Image

- The Image component takes a sprite but Raw Image takes a texture (no borders etc). Raw Image should only be used if necessary otherwise Image will be suitable in the majority of cases.

### ○ Mask

- A Mask is not a visible UI control but rather a way to modify the appearance of a control's child elements. The mask restricts (ie, “masks”) the child elements to the shape of the parent. So, if the child is larger than the parent then only the part of the child that fits within the parent will be visible.

### ○ Effects

- Visual components can also have various simple effects applied, such as a simple drop shadow or outline



# UNITY UI系统 (UI)

## (4) 交互部件

- Button
- Toggle
- Toggle Group
- Slider
- Scrollbar
- Dropdown
- Input Field
- Scroll Rect (Scroll View)



# UNITY UI系统 (UI)

## (4) 使用RICH文本

- 文本不能变颜色？

`<color=red>RICH</color>`





# UNITY UI系统 (UI)

## (1) 课堂实验: MASK的使用

- 参考:

[http://blog.csdn.net/mr\\_ahao/article/details/44419887](http://blog.csdn.net/mr_ahao/article/details/44419887)



# 案例研究

## (1) 传统界面 (SCREEN OVERLAY)

- 导入案例项目
  - Game Jam Menu Template
- 回答以下问题：
  - 屏幕分辨率变化，菜单如何变化
  - 解释 8 段代码的功能
  - 自动布局与 Layout 组件
    - 检查 MenuButtons, MenuPanel 等元素
      - Vertical Layout Group 部件？
      - Canvas Group 部件？
    - 请阅读官方手册给出答案
  - 字体资源？



# 案例研究

## (2) 动画在菜单中应用

- 录制滚动文字按钮（ScrollTextButton）
  - 选择按钮 button
  - 在 button 对象添加 Mask 和 animator 组件
  - 在 animation 窗口创建动画，进入录制状态。
  - 将 button 内部的文字移到左边起始位置，
  - 在动画结束处添加关键帧，将文字放在右边结束处
  - 播放动画！
  - 将 button 拖到预制资源目录！



# 案例研究

## (3) 3D 菜单

- 制作一个简单 3D UI
  - 项目中创建一个新场景
  - 建立一个 cube
  - 然后建立一个子对象 Canvas 放在 cube 的一个面上，Canvas 设置为 world space
  - 然后再面上放一个按钮对象。这样，cube -> canvas -> button 形成树结构。
  - 用一个简单动画运动 cube ...



# 案例研究

## (4) 3D 菜单

- 导入案例项目
  - Sample UI
- Menu 3D 分析
  - 加载 Menu 3D 场景
  - 运行！很酷炫啊，菜单随鼠标飘动，而且有动画
  - Canvas 是 World Space 的
    - Window 代码 TitleWindows的作用？
    - 按钮动画，如 Settings？
  - GUI Camera 哪些属性与 Main不同
  - EventSystem
  - SF预制，包含哪些？
  - ApplicationManager
    - 蓝色、灰色，黑色的含义



# 案例研究

## (4) 课堂实验

- 使用预制以外的资源，完成如下效果：



# 案例研究

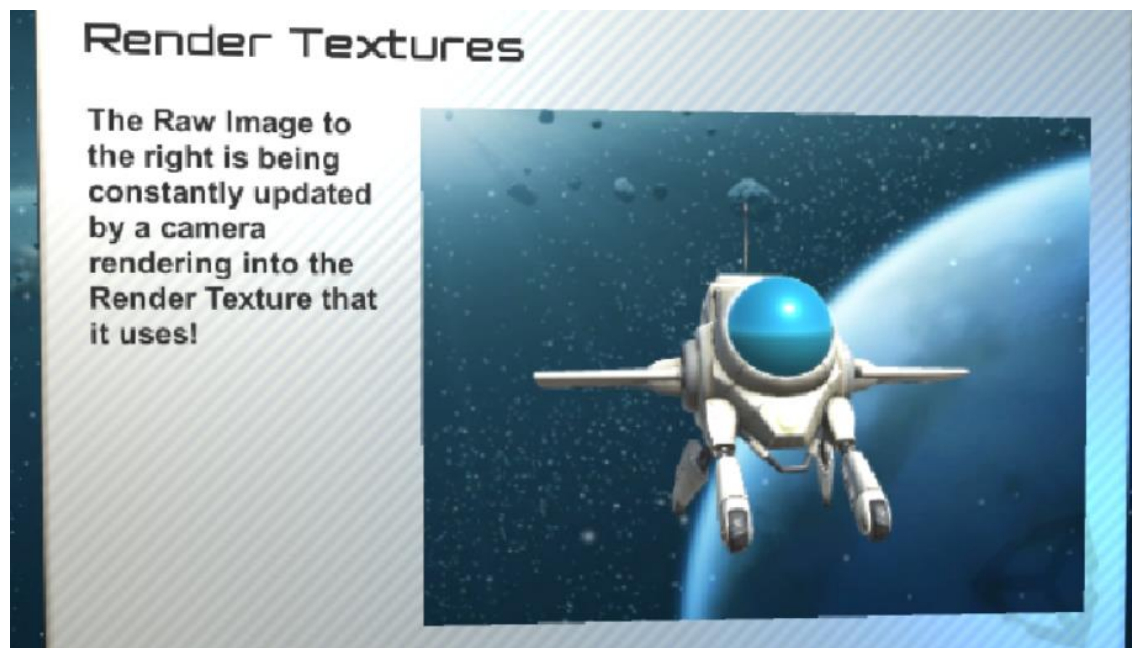
## (4) 3D 菜单

### RenderTexture

- 打开 Render Texture 场景
- 这个场景要点？
  - 有哪些相机
  - “画中画” ...

### 如何实现狙击步枪上的瞄准镜？

给出你的方案... ..



# 面向对象设计思考

## (1) 对象扩展技术

- 见以往课件





# 课程小结

## ○ Unity UI 系统基础

- 画布的种类
- 旋转点、锚点与排版
- 显示部件、交互部件
- Mask等效果部件

## ○ 案例研究

- 自动排版部件
- 动画的应用
- 多摄像机应用

## ○ 面向对象的编程思考

- 对象方法扩展
- 对象扩展与 iTween



# 作业 (LAB 10)

## ○ 作业

- 参考 <http://www.tasharen.com/ngui/exampleX.html> , 使用 UI 制作背包系统（武器装备系统）
- 或者, 参考 <http://www.tasharen.com/ngui/example9.html> 制作可展开公告牌
- 背包系统, 作业参考:
  - [http://blog.csdn.net/qq\\_20496459/article/details/51421360](http://blog.csdn.net/qq_20496459/article/details/51421360)
  - <http://www.cnblogs.com/wangsta/p/5516470.html>

