

Software Engineering Lab 4

苏建锐 10195101553 East China Normal University

Assignment: 请列举出一个现实中的产品/系统/服务/技术/..., 并说明其包含了/使用了哪些软件架构, 这样设计的好处和/或坏处是什么?

以游戏《明日方舟》为例, 说说我的理解和看法。

《明日方舟》玩法由策略、塔防、RPG 相结合, 采用 Unity 开发。同时考虑到技术的实现, 采用了一个 2D 和 3D 相结合的表现形式。

MVC

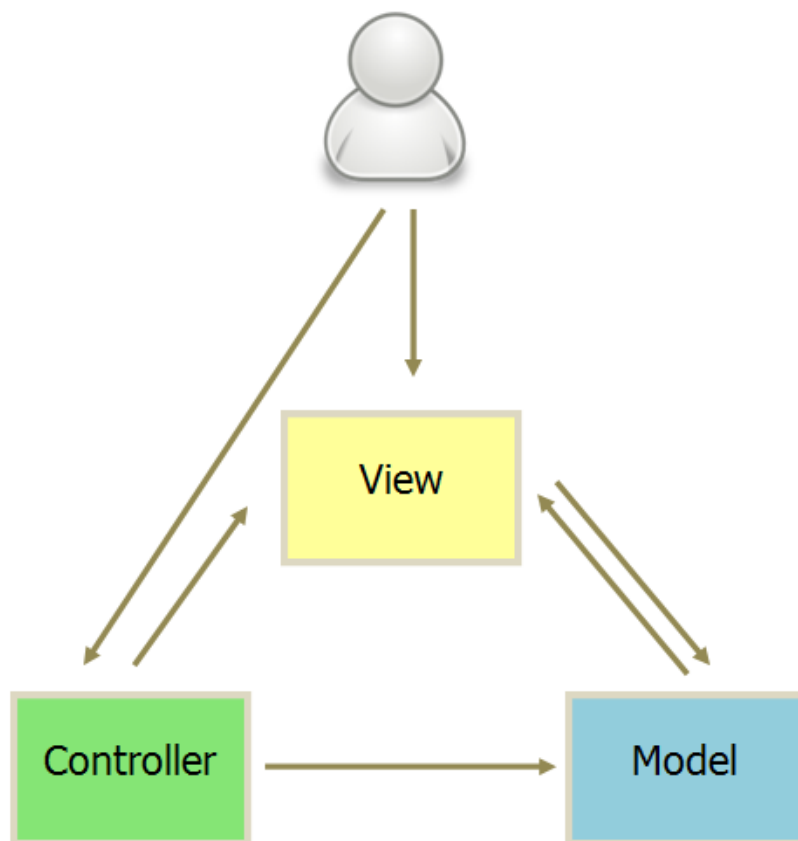
表现层 (View): 游戏画面。UI

逻辑层 (Controller): 数据接口, 操作控制, AI

数据层 (Model): 数据保存, 图片、声音等资源

SFramework 中, View 层是单独的, Model 放在基类中, Controller 则在派生类, 实现了 MVC 的分离

举例: PlayerHUD+PlayerCtrl (PlayerYuka) +IPlayer



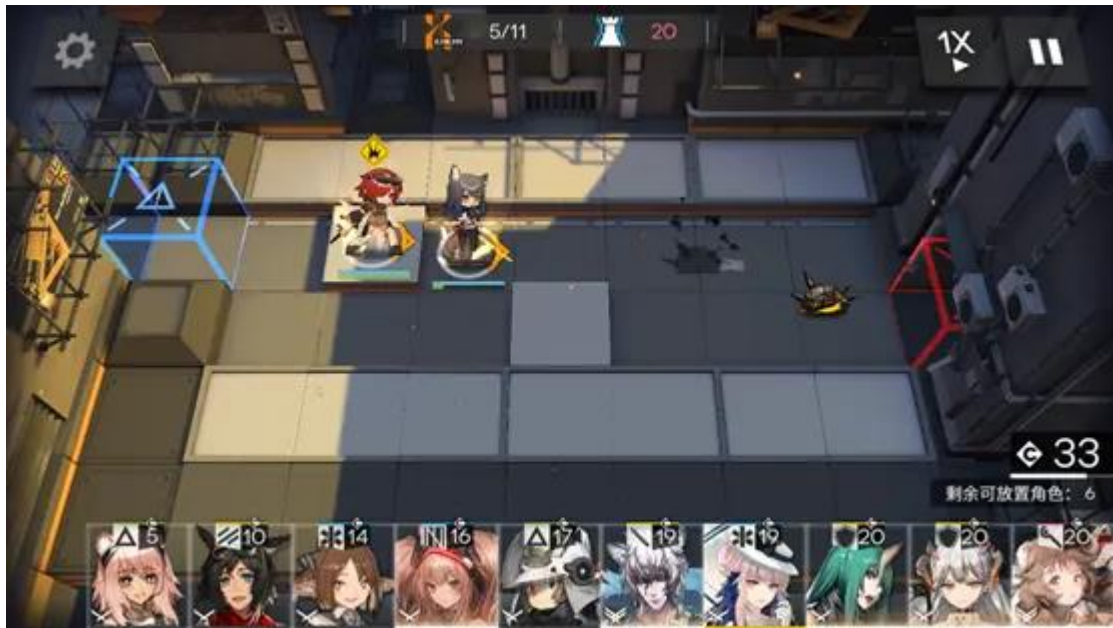
MVC

优点：实现了层次化设计，是一个基础常用的框架

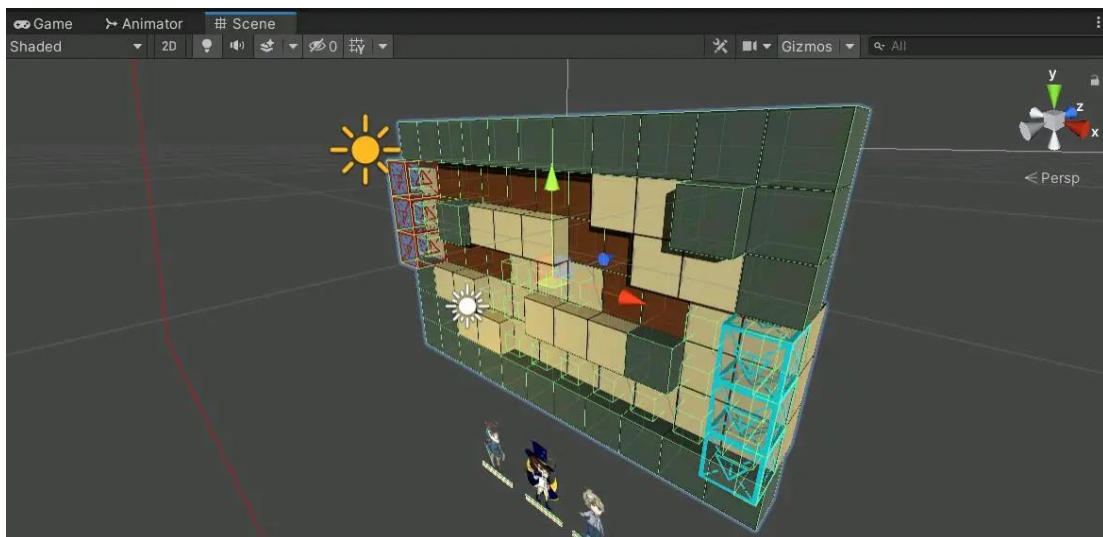
缺点：MVC 如果没有用好，容易出现某一层过重的现象（如 View）。MVC 更适合横向铺量的项目。游戏（非 UI 部分）则不太合适。

在游戏界面中最主要元素有两个：干员（我方可操纵单位）/敌人，和关卡场景

干员：2D 骨骼/Trail Renderer



关卡：2.5D/正交相机+透视相机



可以看到使用 Unity 开发降低了开发难度，同时降低了维护成本，使得整个开发不会非常的复杂。

容易出现的问题在于由于小人模型是 2D 平面，而场景模型是 3D 图形，且摄像机的位置大概是斜向上 60°左右，这就容易导致小人模型和 3D 模型之间的穿模，官方给出了这样的解决方案：

视觉方案

- 场景（地图，背景和小物件）
 - 3D模型：通过标准PBR工作流程制作
 - 块面化的雕刻
 - 风格化的贴图绘制
 - PBR Shader
- 单位（角色和敌人）
 - 2D动画：通过Spine骨骼动画软件制作
 - Customized Unlit Shader

视觉方案-场景

- Unity的PBR渲染模型
 - 采用略微优化和修改后的BRDF2
 - Performance optimized
 - Colored shadow
 - Height fog
 - Lightmap fix on mobile (more details later)
 - 使用Unity的静态烘焙（lightmap and shadowmask）
 - One mixed main light (dynamic + baked)
 - Various of baked lights
 - Additional reflection probs
- Additional unlit FXs with traditional shaders

问题一：空间关系错误

- 错误的思路
 - 渲染角色时关闭z-test
 - 需要被前方的场景模型正确遮挡
 - 关闭z-write，手动对角色和场景物件进行排序
 - Break batching，并且有一些跨格点的小摆件不适用
- 解决思路
 - 角色需要“看起来”垂直于摄像机但“实际上”垂直于场景

问题一：空间关系错误

- 解决方案：tweak the depths
 - 着色时垂直于摄像机（Billboard）
 - 在Vertex Shader中将深度变换到垂直于场景的空间
 - 对【深度测试/深度写入】来说，就好像角色是垂直于场景一样

（不甚了解所以暂时无法给出评论）

Summary：使用 Unity 开发，2D+3D 的画面表现方案，MVC 框架包含游戏中的元素。

参考资料：

<https://www.163.com/dy/article/FT0L439H0526E124.html>

<https://www.bilibili.com/read/cv10606906/>