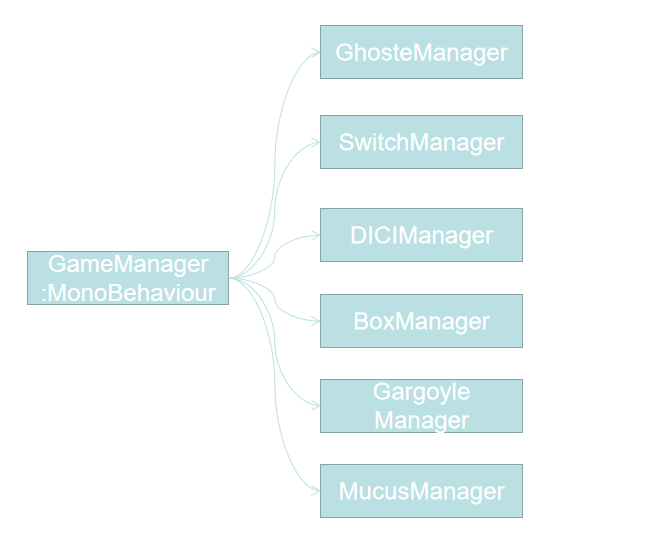
## 代码结构修改

首先修改了代码结构，将怪物等需要创建多个游戏对象的交由 GameManager 统一管理，方便后续对怪物重生等功能的实现。

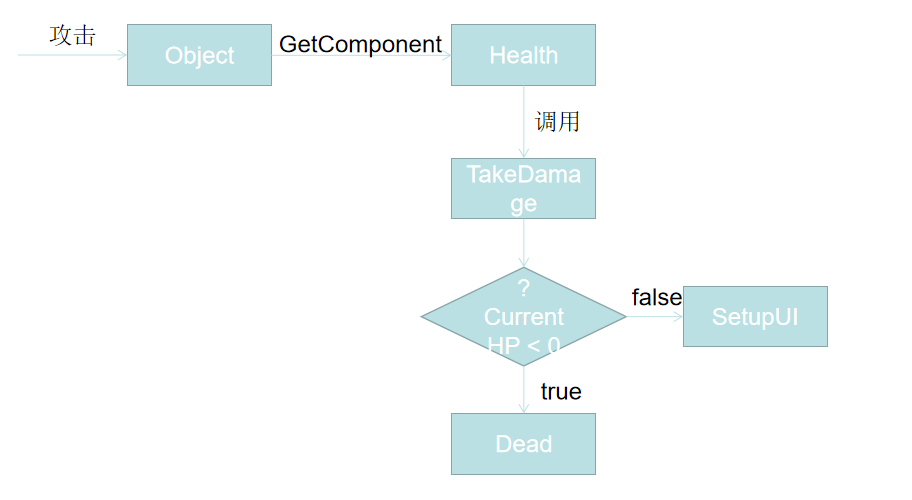


## 血量与伤害机制的实现

2.1. 血量

相关代码：Health.cs

当A对B发起攻击时，会获取B的Health脚本，调用TakeDamage方法实现扣血、UI更新、受伤音效播放。



2.2. 怪物攻击机制

相关代码：Observer.cs

修改部分：Line 33 if中添加判断Time.time >= m\_AttackTime

怪物创建后3秒内不会对玩家造成伤害。

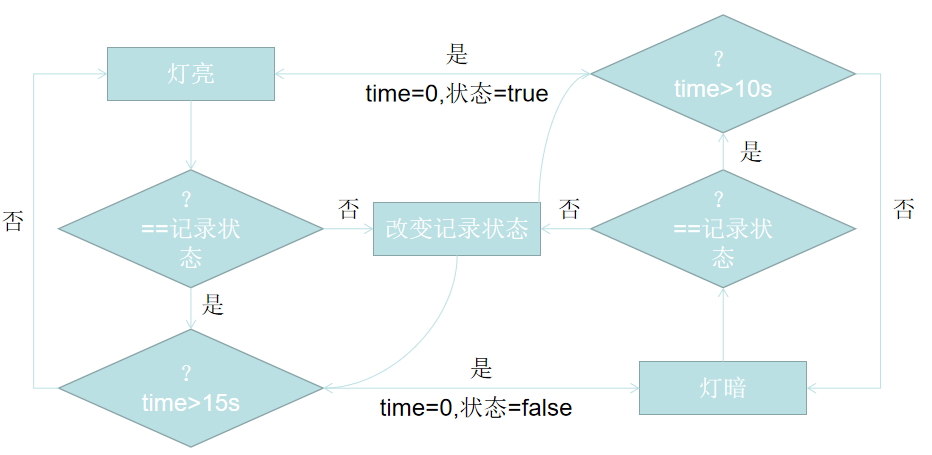
2.3 灯光伤害机制

相关代码：LightHarm.cs

通过触发器获取在灯光范围内的怪物，然后通过射线判断是否直接照射到怪物从而判断是否对怪物造成伤害

## 场景机制

相关脚本：LightHarm.cs



## 机关设置

* 1. 开关

相关代码：SwitchAttribute.cs

4.2. 地刺

相关代码：DICIAttack.cs, DICITimer.cs

DICIAttack.cs

用于判断DICI对角色的攻击， 当地刺处于攻击状态且角色进入触发器范围内时，角色受到伤害。

DICITimer.cs

用于管理地刺开启与休息的时间，并操作地刺动画

* 1. 减速

相关代码：MuCuAttribute.cs, PlayerMovement.cs

通过触发器判断角色是否进入，若进入，则调用 PlayerMovement 的 InMuCu 方法，离开时则调用 ExitMuCu 方法

## 增加场景物件和角色道具

* 1. 武器

相关代码：WeaponAttribute.cs

武器中挂载了攻击和装弹音效，在玩家获取到武器时，会将音效赋给玩家。

* 1. 宝箱

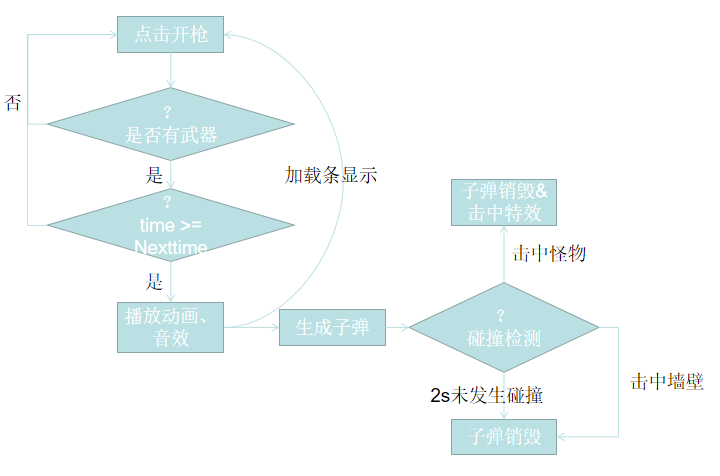
相关代码： BoxAttribute.cs, PlayerAttack.cs, PlayerSkin.cs

通过 GameManager 设置宝箱，并随机在其中一个宝箱内放置武器

当玩家开启宝箱时判断是否拿到武器，若拿到武器，则调用PlayerAttack与PlayerSkin中的相关函数，获取武器中的相关信息。

## 玩家攻击能力

相关代码：PlayerAttack.cs, ShellHarm.cs, circleProcess.cs



## 遇到的问题

1. 开枪时点击位置与实际开枪位置相差较大

使用的是通过摄像机发出射向点击方向的射线，通过光线与碰撞器相交来获取位置，从而可以朝指定方向发射子弹。但在一开始发现在下面场景时，朝向与点击方向严重不一致。



后通过Debug输出，判断是射线与墙壁相交，导致获取的位置在墙壁上，从而使得设计方向严重不一致。后查到射线相交可以只检测与指定层的对象相交，于是添加了Level层，将判断改为 if (Physics.Raycast(ray, out hit, 50, 1 << (LayerMask.NameToLayer("Level")))) 后，问题解决。

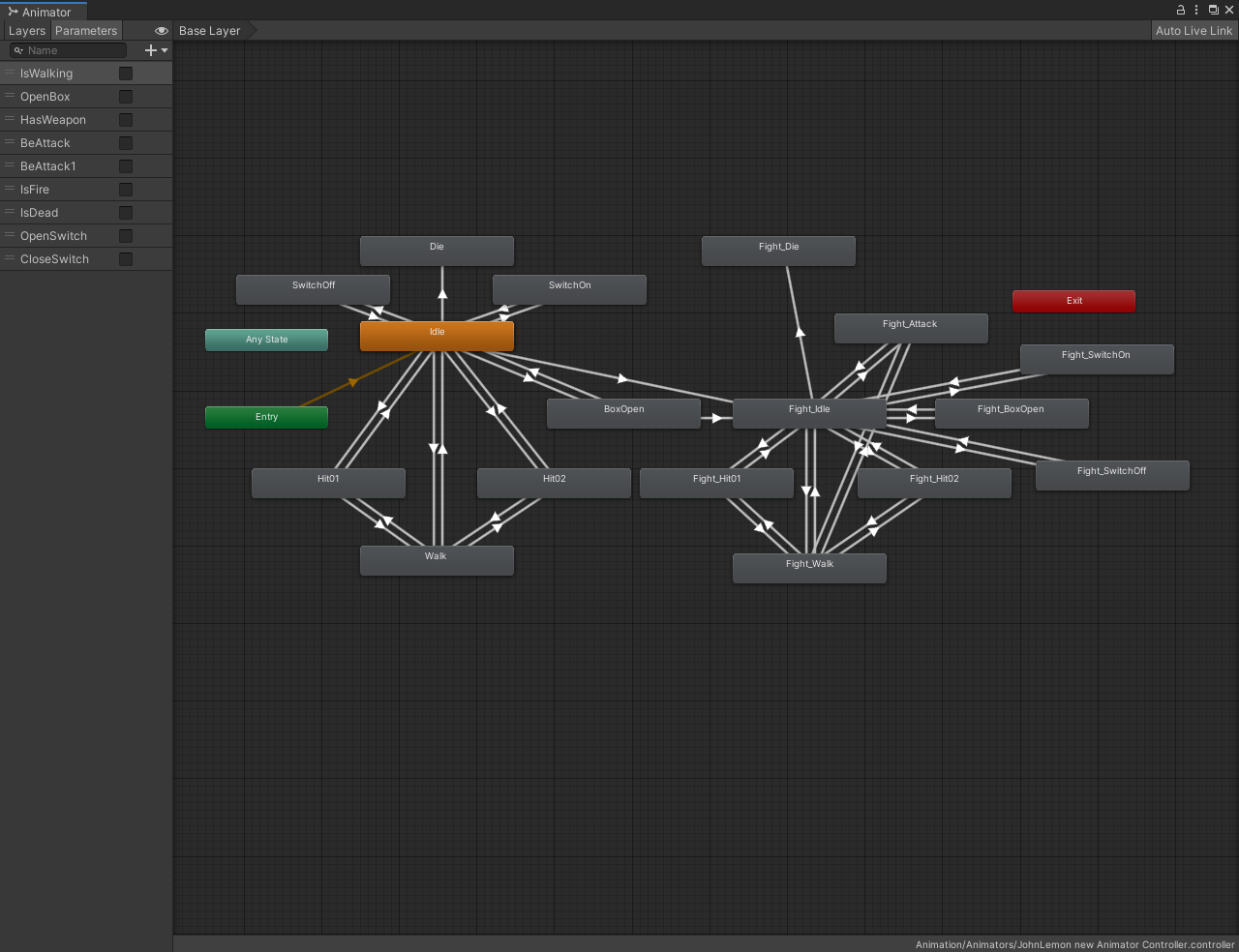
1. 灯光对石像鬼没影响

思路是通过触发器获取在灯光范围内的怪物，然后通过射线判断是否直接照射到怪物从而判断是否对怪物造成伤害。但在写的过程中发现只对幽灵有效果，石像鬼不受影响。后查相关资料发现是触发器只对有刚体的对象才有效，对石像鬼添加刚体后便解决该问题。

1. 如何设置主角动画

添加了新的动画之后，如何在AnimatorController 中添加新的动画，一开始思考了很久。最开始是打算将 IsWalking 改成 int 类型的 State，通过改变 State 的值，然后改变角色的动画。

但在实现过程中发现，PlayerMovement会一直设置State的值，这样就无法播放其他动画了，后来跟其他人进行讨论，发现利用多个 bool 值来控制动画实现起来会更加容易，便采用了该方案。



1. 关于拿到武器后动作与音效的设置

当时考虑到可拓展性，在拿到武器后，一般来说持不同武器的动作与攻击时的音效一般都是不一样的，如何在添加进新的武器后只需改变少量代码是当时的一个思考方向。一开始的想法是将动作与音效都挂载在武器上，当角色获取武器后，直接通过武器获取作动和音效对象，然后对象对应参数。

在与其他人讨论之后，决定是将音效放置在武器中，获取武器时取得音效，然后将动作直接放置在 AnimatorComtroller 中，不同的武器，改变不同的bool值即可。