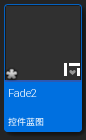
# 淡入淡出动画

这里，我们要做一个淡入淡出动画。这个动画可以让我们在进入对局、（由于死亡等原因）重置对局时，有一个类似电视剧中切换场景的动画效果

在内容管理器中创建Spikes文件夹，用于存储：淡入淡出动画、死亡动画、碰撞致死物体、死亡声音、死亡摄像机晃动的内容

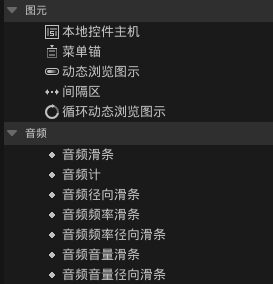
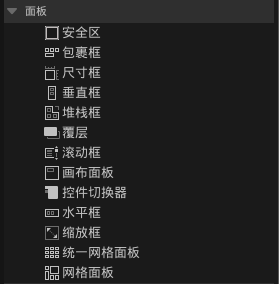
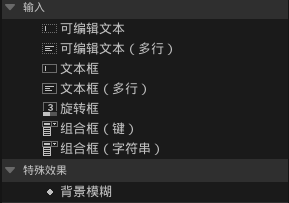
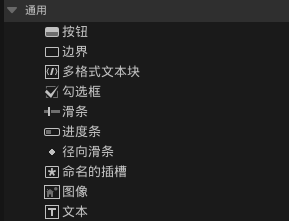
首先创建淡入淡出动画，右键，选择“用户界面-控件蓝图”，父类选择用户控件（User Widget），将蓝图命名为Fade



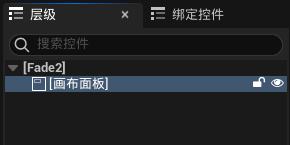


Widget Blueprint是一种特殊的蓝图，用来制作2D的东西，比如UI主菜单、死亡画面、死亡动画或淡入淡出动画

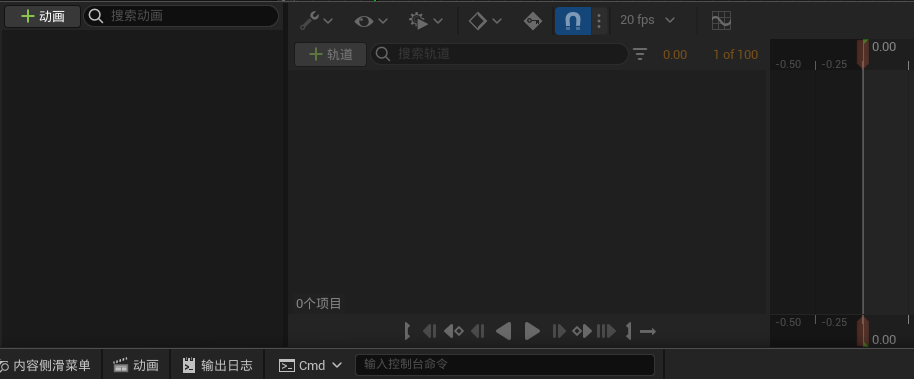
在控制板里（特别是通用控制板）可以创建多种控件（Widget）来进而创建UI



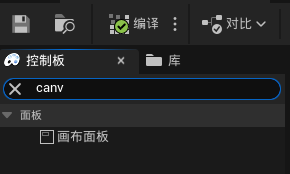
在Widget Blueprint控件蓝图中，要特别的注意层级的设置,将那个控件设置成哪个控件的子控件



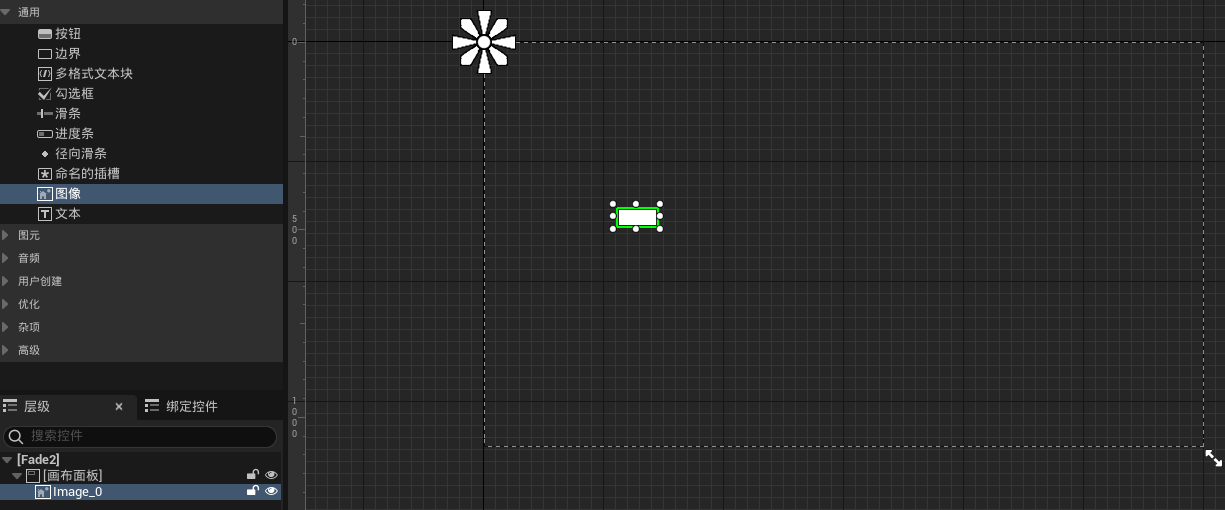
我们也可以创建动画，让UI展示一些动画效果



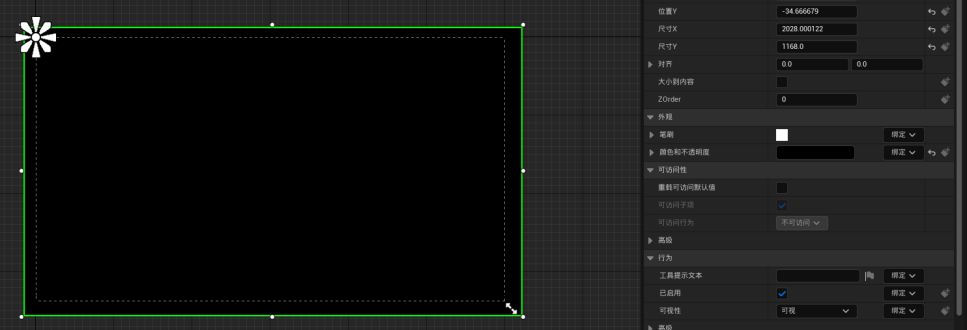
打开该蓝图，在控制板中创建画布面板（Canvas Panel）



接下来创建图像控件，将图像控件作为Canvas Panel的子控件



将该图像（图片）拉大到比画布面板稍微大一圈的大小，并将图片颜色在细节中改为黑色



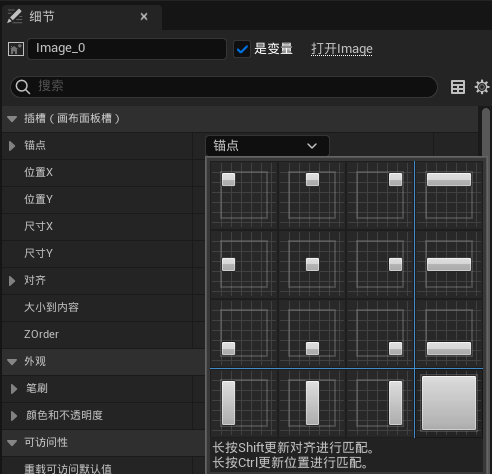
注意此时canvas panel的左上角有一个白色的像花朵一样的东西。这是锚点的工作方式，也是要在意创建的图片绑定为画布面板的子控件的原因。

此时我们放大缩小画布面板的大小，会发现这个黑色图片也在跟着放大缩小

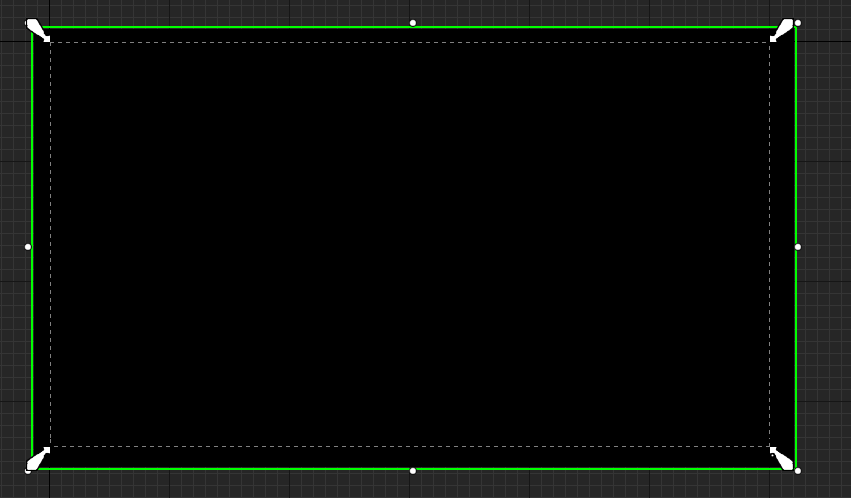
问题就在于，画布面板是你所设置的屏幕的大小。但是大多数情况下，你设置的1080p大小的面板，但是很多人不是1080p

这里我们要给图片一个针对于画布面板的锚点

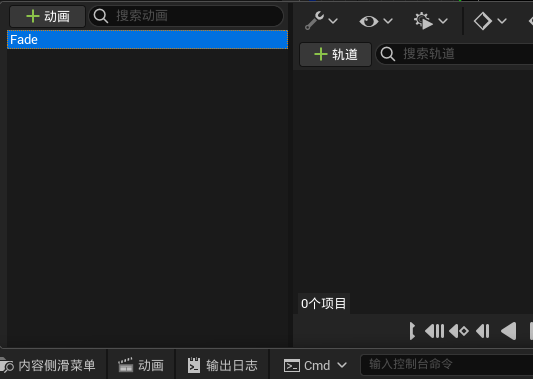
点击图片（注意不是画布面板。可以在层级中点击），在图片的细节中找到锚点，选择右下角的



此时我们的设计器中，层级展示为这样



接下来创建动画，命名为Fade

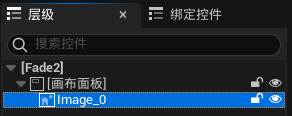


点击Fade动画即可创建轨道

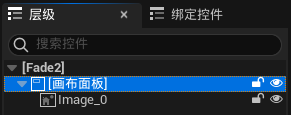
接下来创建轨道

（这里要注意叫法。追踪轨道和创建轨道本质上都是创建轨道。但在引擎操作中，点击绿色加号且需要点击层级中控件的叫追踪轨道，点击灰色加号并在时间戳区域创建轨道的叫创建轨道）

注意，在我们追踪轨道时，我们需要在层级中提前选择我们要追踪的动画对象。比如这里我们要给图片创建动画，我们需要在层级中点击图像控件，这样在追踪轨道时才能追踪图像控件的动画轨道

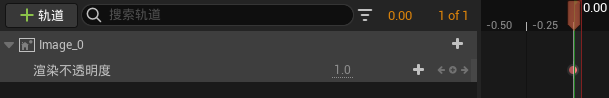
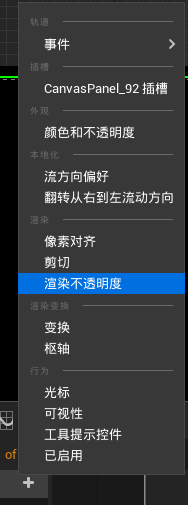


如果点击的是画布面板，那么就不能追踪图像控件的动画



接下来点击图像控件轨道右侧的“+”号来创建动画轨道

因为我们希望创建淡入淡出动画，也就是控制黑色的不透明度即可。所以我们选择“渲染不透明度”



此时我们可以修改数字来决定这一帧的不透明度，决定完成后点击右侧的小加号



由于我们希望创建的是fade动画，所以我们第一帧为透明，最后一帧（在第0.25s的位置）为不透明

所以第一帧处，将数值改为0后点击添加新关键帧

最后一帧处，将数值改为1后点击添加新关键帧

此时在动画帧间滑动播放指示针即可看到fade动画效果

将fade动画绑定在相应过程中的操作在下面会讲到

# 死亡动画

死亡动画包括信息提示动画和摄像机晃动两个部分。

创建“用户界面-控件蓝图，父类为用户控件的”DeathScreen控件蓝图

打开并创建画布面板，在“特殊效果”中创建“背景模糊”作为画布面板的子控件，将背景模糊拉到比画布面板略大一圈

将背景模糊的锚点修改为最右下角的锚点（和上一部分一致）

创建图像控件作为背景模糊控件的子控件，将图像的颜色变成暗红色

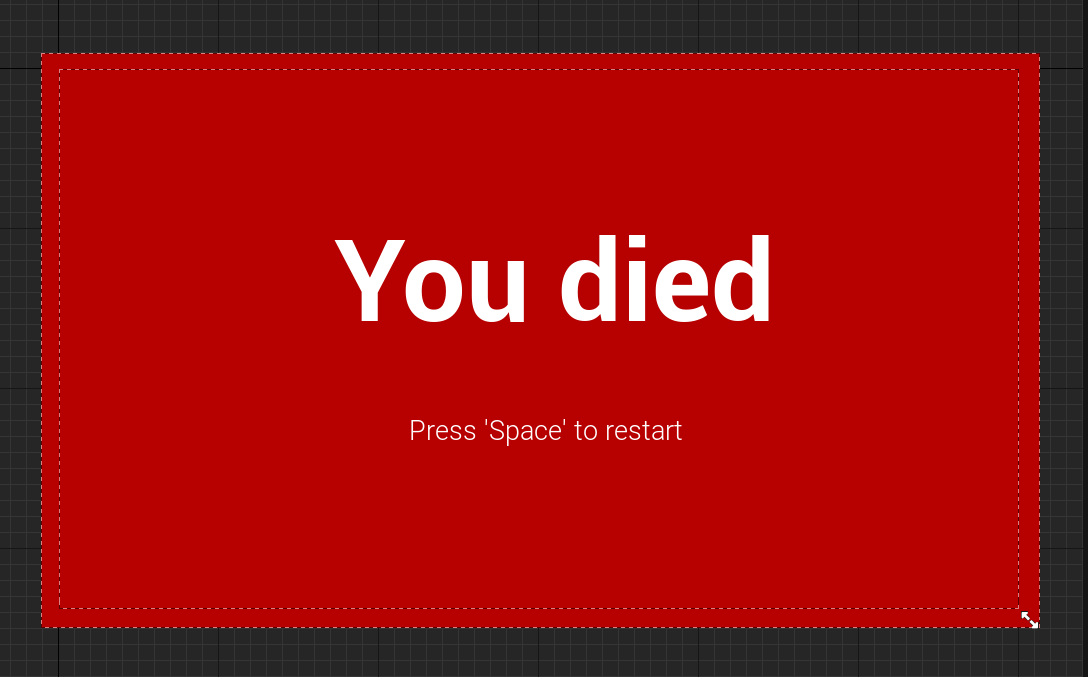
创建两个文本控件，作为画布面板控件的子控件

在细节中

其中一个文本显示You died，另一个显示Press 'Space' to restart

将文字调整字体、大小和居中对齐，将文字的锚点定在屏幕中间

最终效果如图所示



接下来创建动画

动画命名Death，创建追踪画布面板的轨道，创建渲染不透明度的轨道

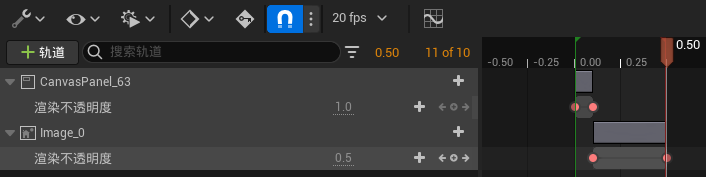
在第一帧将数值设置为0，在最后一帧（第0.1秒）将数值设置为1

接下来创建其他控件的动画

进行红色图片的追踪轨道

创建动画，使得红色图片在死亡的瞬间有一个红色闪烁，之后闪烁会暗下去，这样玩家可以在暗下去之后看到地图状况

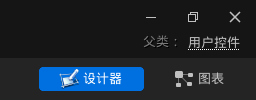
设置图像在最后一帧（第0.5s）的数值为0.5



这样动画创建完毕。保存，编译。

接下来将动画绑定到游戏流程中

在右上角点击图表，进入蓝图的设置



注意给出的三个节点中的Event Construct节点和Event Tick节点

**Event Construct节点**

//在下方的slate控件构造后调用。由于从层级中添加和移除，此事件可能多次调用，具体取决于slate对象的使用方式

//如果需要一个真实的“创建后即调用一次”事件，则使用OnInitialized

//目标是用户控件



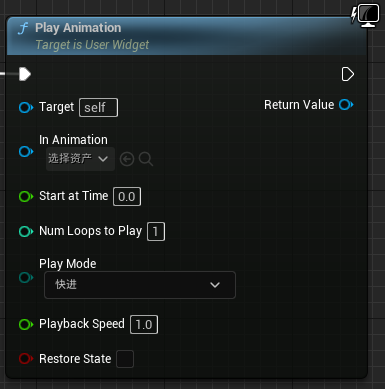
从Event Construct节点链接出Play Animation节点

**Play Animation节点**

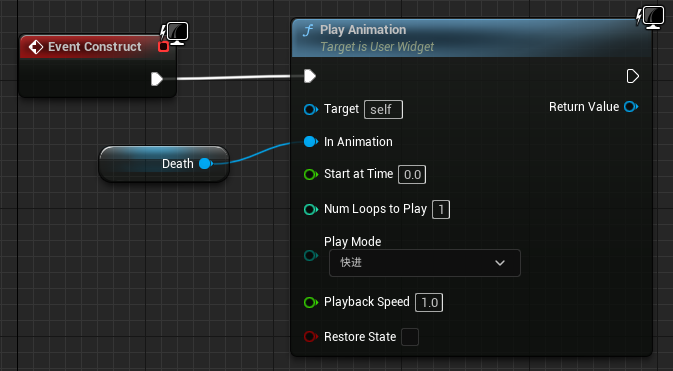
//在此控件中按指定次数播放动画

//目标是用户控件

//Cosmetic.This event is only for cosmetic,non-gameplay actions.



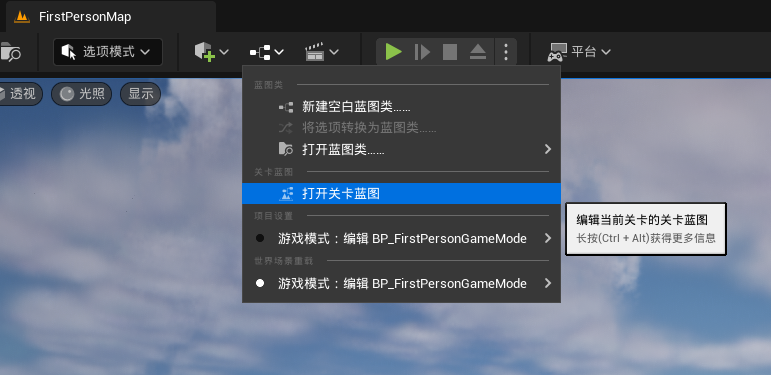
其中的In Animation是要播放动画的引用。我们在左侧变量中直接拉进一个Death动画变量即可



此时当我们create this widget，他就会播放这个动画。

此时我们需要一个事件，让这个事件来触发播放动画。

接下来进入关卡蓝图



创建Keyboard Event **1节点**//Events for when the 1 key is pressed or released.

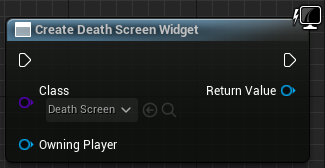


**Create Widget节点**

//创建新控件



注意，这个节点的原生名称是Create Widget，但是刚创建的节点显示为Construct NONE。此时我们需要选择类（这里选择Death Screen），这时节点名称变为Construct [name of class] Widget



**Add to Viewport节点**

//仅用户端

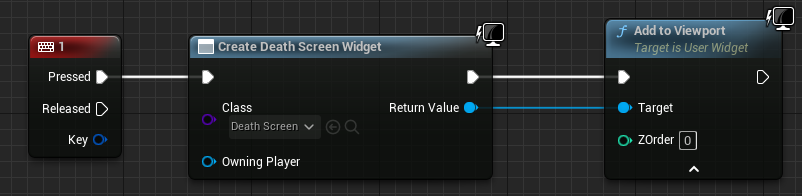
//将其添加到游戏视口并填充整个屏幕，除非调用SetDesiredSizeInViewport来显式设置大小

//目标是用户控件

//Cosmetic.This event is only for cosmetic,non-gameplay actions.



此时将他们链接，即可实现：当Pressed 1 key，会在Viewport上播放Death Widget的死亡动画



此时我们开始游戏，摁下数字键1（不是小键盘上的数字键），即会播放DeathScreen的死亡动画。

Tip：当多次摁1键的时候，该动画效果会重叠，使得本设计好的透明效果变得消失。如果不希望，可以加个Do Once节点

接下来实现：当摁下空格键时，重启关卡

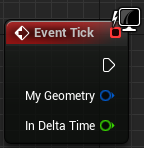
在Death Screen中，我们需要用Event Tick节点来判断某个tick中是否摁下空格键

**Event Tick节点**

//Tick此控件。在派生类中重载，但固定调用父实现

//目标是用户控件

//Cosmetic.This event is only for cosmetic,non-gameplay actions.



**Was Input Key Just Pressed节点 & Was Input Key Just Released节点**

//如给定键、摁钮在上一帧朝上，在此帧摁下，则返回true。Released同理

//目标时玩家控制器



**Get Player Controller节点**

//迭代本地和可用的远程玩家控制器，返回找到的玩家控制器

//在网络客户端上，这将只包括本地玩家，因为远程玩家控制器不可用

//只要没有新玩家加入或离开，索引将为一致，但在不同客户端和服务器上不一样

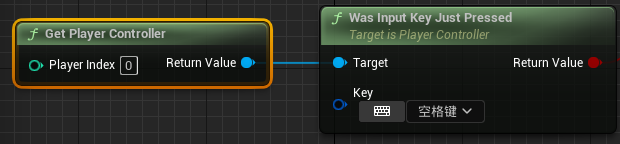
//目标是Gameplay静态



此处要特别注意，在没有打开情景关联的时候，会有许多Get Player Controller节点。注意每个节点的不同之处（将鼠标悬浮在目标节点上即可查看该节点的基本功能）



此处我们将上述两节点进行连接，将Key设置为空格键



**Get Current Level Name节点**

//获取当前打开关卡的命名

//目标是Gameplay静态



**Open Level (by Name)节点**

//漫游到另一个关卡

//目标是Gameplay静态



接下来链接蓝图结构

首先链接关卡结束时的场景切换动画Fade

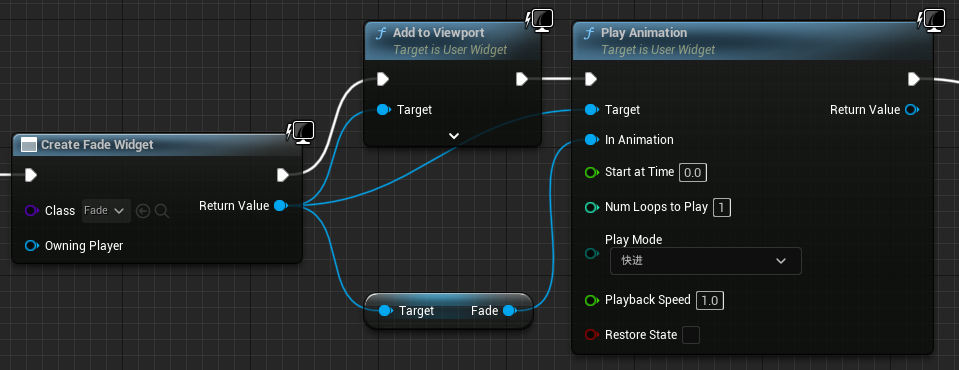
首先创建Create Widget节点，将节点与fade动画进行绑定

然后将该widget， Add to Viewport节点

然后开始Play Animation

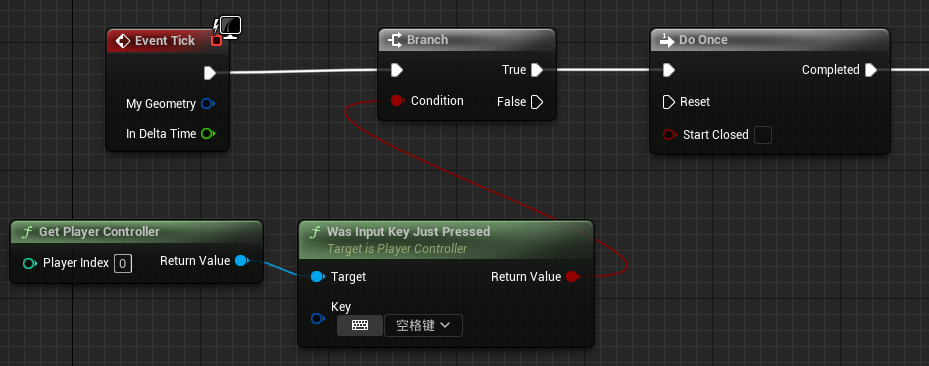
播放的动画是Fade，所以要用Fade的动画变量给Play Animation的In Animation参数赋值

如图所示



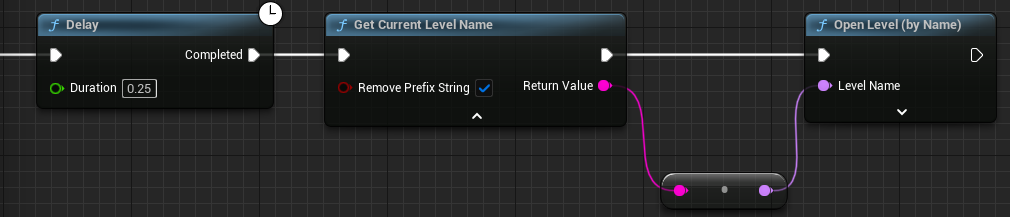
接下来链接摁键判断和tick事件

在每个Tick，若判断条件为True，则执行true相关处理。即Branch节点



接下来设置重置关卡链接

在重置关卡时，我们首先需要获得关卡的名字，并将该名字赋值给新开关卡的函数



接下来将三部分进行连接即可

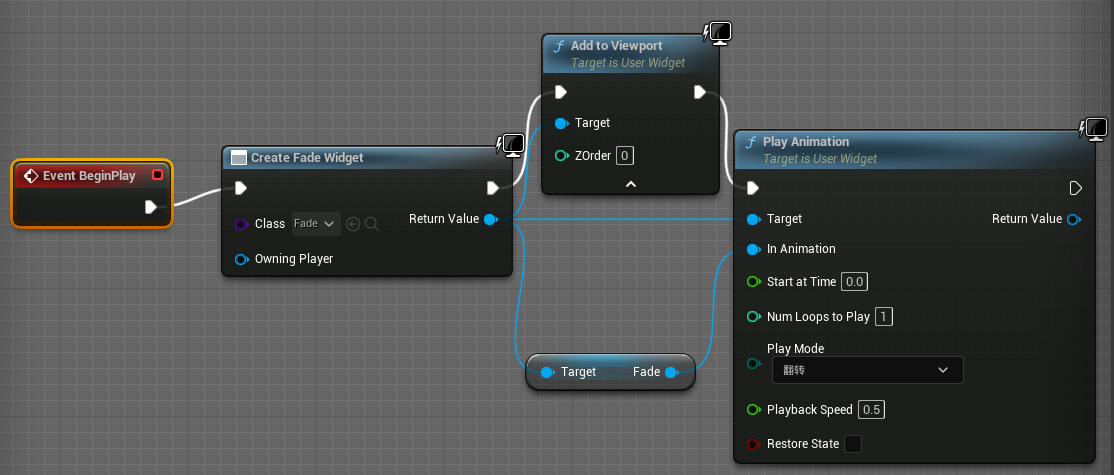
这样就完成了死亡时死亡动画的播放和重置关卡时淡出动画的播放和关卡的重置

但是这样并不能在关卡开始时产生淡入淡出动画

所以我们要回到关卡蓝图，在关卡开始时（**Event BeginPlay节点**），播放淡入动画

我们制作的淡入淡出动画Fade本质上是淡出动画，淡入动画是淡出动画的反向播放

所以我们只需要在Fade动画的Play Animation节点中将Play Mode参数选择为翻转即可



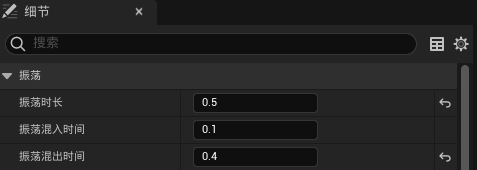
这样在关卡开始的时候就有了一个淡入效果

接下来创建死亡摇晃效果

在Spikes文件夹创建父类为Legacy Camera Shake的蓝图类，将蓝图命名为DeathShake

打开蓝图

在细节中修改震荡时长



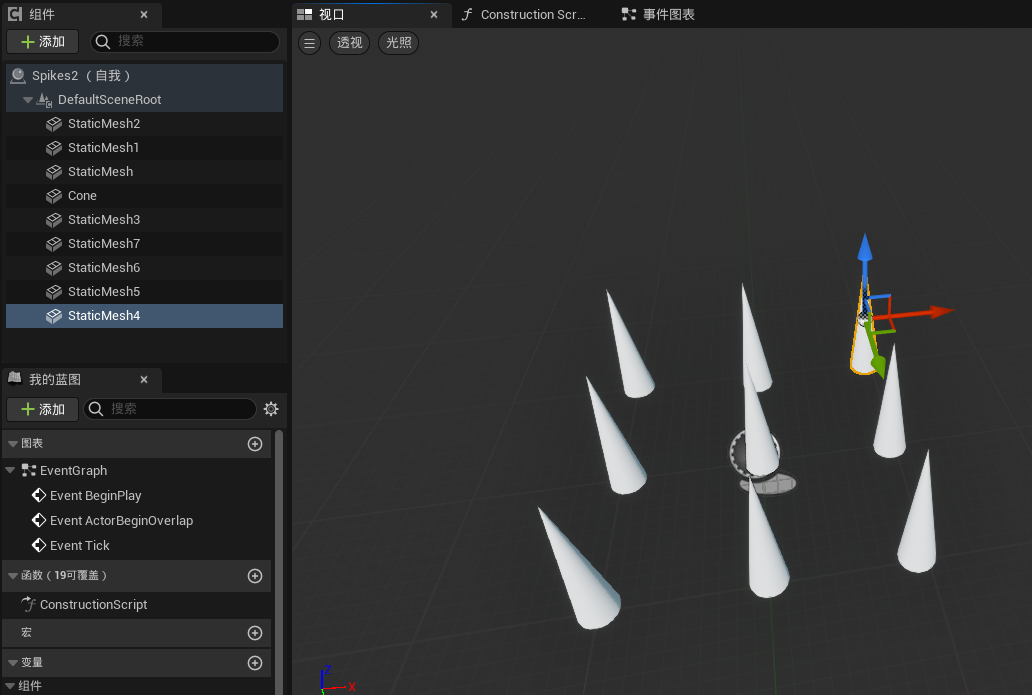
三个旋转震荡，振幅为6，频率为60

三个位置震荡，振幅为40，频率为100



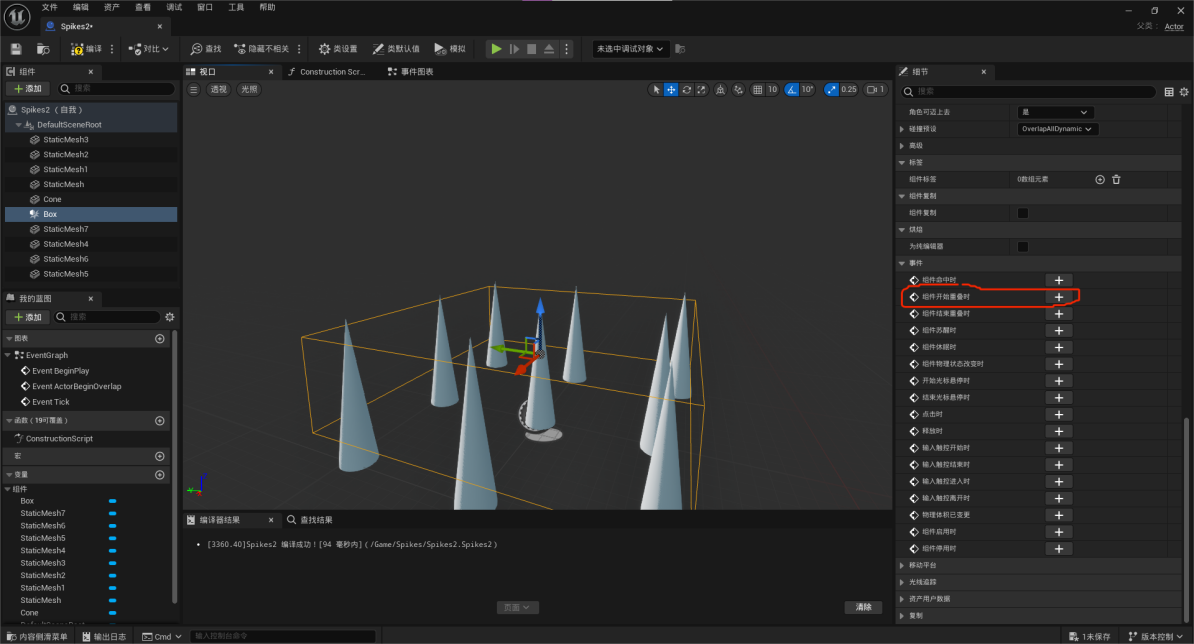
接下来，创建父类为Actor的蓝图，命名Spikes

首先在视口界面中创建9个锥体组件



接下来创建Box Collision组件，将hitbox的大小设置为与椎体占地体积一致

并在Box Collision组件的细节中添加“组件开始重叠时”事件



接下来就会进入到事件图表窗口。

**On Component Begin Overlap节点**

//内容开始重叠次组件时（如玩家走入触发器）调用的事件

//在对象拥有阻挡碰撞的事件中（如玩家与墙壁碰撞），参见“命中”事件

//@注意：此组件和其他组件必须将GetGenerateOverlapEvents（）设为true，才能生成重叠事件

//@注意：从另一个物体运动接受重叠时，”Hit.Normal”和”Hit.ImpactNormal”的方向将被调整，表明其他物体对此物体施加力。

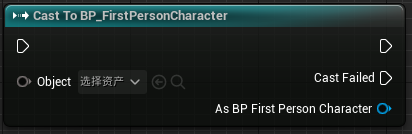


从OtherActor接口链接Cast To BP\_FirstPersonCharacter节点

**Cast To BP\_FirstPersonCharacter节点**

//尝试将Object作为它可能作为实例的一个蓝图类”BP\_FirstPersonCharacter”来访问

//注意：这将导致蓝图固定被加载，开销可能较高



然后我们需要一个DoOnce节点，来保证自己不会死很多次

既然我们已经死了，我们也不希望死亡后还能移动。所以我们要禁用输入

**Disable Input节点**

//从玩家控制器所处里输入的堆栈中移除此Actor

//目标是Actor



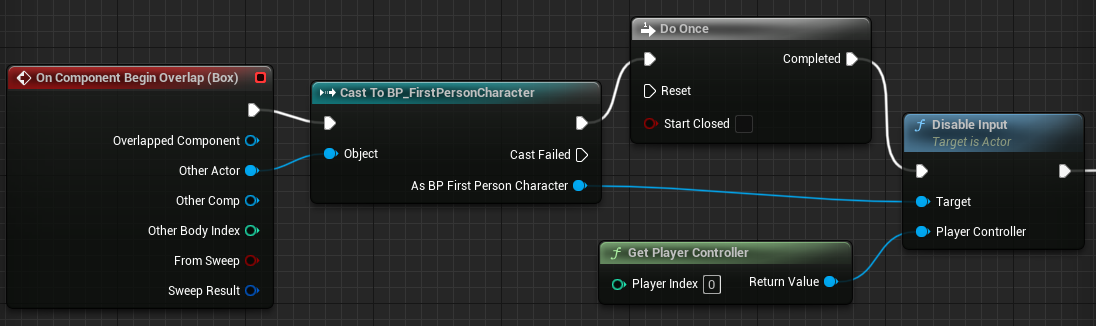
Get Player Controller节点



将两个节点进行连接

将Cast To BP\_FirstPersonCharacter节点与DisableInput节点进行链接

将DoOnce节点链接在Cast To BP\_FirstPersonCharacter节点和DisableInput节点中间，组成死亡绑定与禁止输入节点族



接下来创建死亡部件。

死亡部件包括死亡动画、死亡音效的播放

首先创建死亡动画节点族

首先创建CreateWidget节点，将class选择DeathScreen

然后将该动画添加至Voewport，即Add to Viewport节点

在播放死亡动画时，我们需要开始进行死亡摄像机晃动，即Start Camera Shake节点

**Start Camera Shake节点**

//在次摄像机上播放一个摄像机晃动

//目标是玩家摄像机管理器



我们需要给目标（target）接口一个Get Player Camera Manager节点

**Get Player Camera Manager节点**

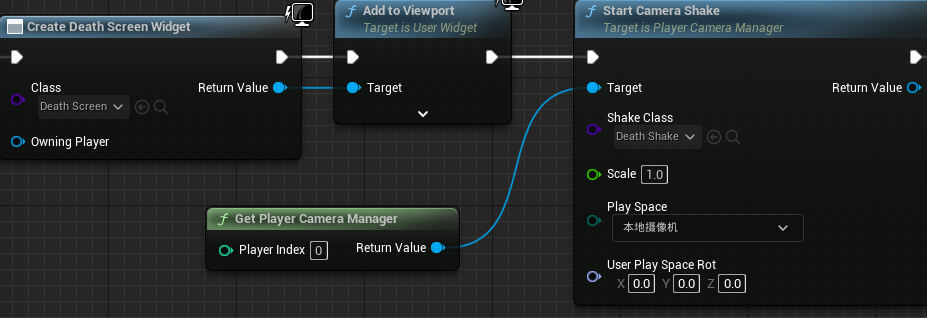
//返回指定玩家索引处的玩家控制器的摄像机管理器

//不包括没有玩家控制器的远程客户端

//目标是Gameplay静态



将节点进行连接



然后创建死亡音效节点族

实际上，死亡音效的播放在这里只需要一个节点，也就不用称呼为节点族了

**Play Sound 2D节点**

//仅客户端

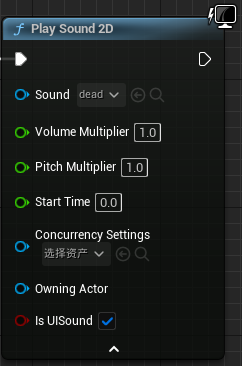
//播放不含衰减的音效，最适用于UI音效

//\*发射之后即遗忘

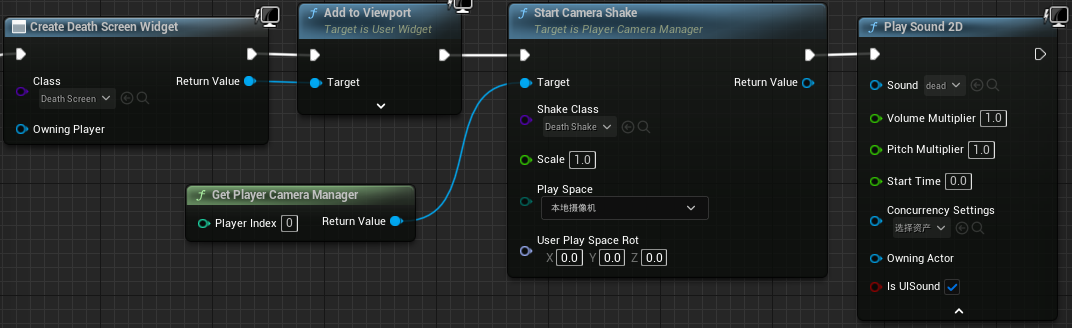
//\*并非复制

//目标是Gameplay静态

//Cosmetic.This event is only for cosmetic,non-gameplay actions.



然后将死亡音效节点族与死亡动画节点族连接在一起组成死亡特效节点族，即完成了死亡动画的制作



然后将死亡绑定与禁止输入节点族和死亡特效节点族进行连接，即完成了死亡动画的制作。