# 使用框架制作流程

## 步骤1-创建框架场景物体

## 步骤2-配置资源

**说明：**Slot最外层包括不限于以下效果。细节部分比如需要多个Animator去实现效果，则以默认的最外层Aniamtion设置动画事件帧，子物体的animato动画触发订阅到事件(使用AnimatorCTR)。

### 配置图片、特效、Spine

**JackpotBorder(彩金分数框)：**设置图片

**FadeFg(免费过场)：**子物体设置图片、特效、Spine

**FadeJp(彩金过场)：**子物体设置图片、特效、Spine

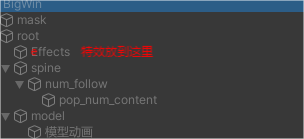
**NpcNg(普通界面Npc)：**设置模型或Spine，模型的情况创建到NpcNg子物体（改成了**Npc\_spine**）

**NpcFg(免费界面Npc)：**普通免费通用的情况则删除，只是为了演示spine和模型的区别，以及不通用的情况（改成了**Npc\_model**）

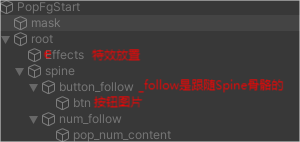
**Loading(加载界面)：**

****

**BigWin**：model下面是放人物animator这种动画(model下会自动把触发动画订阅最外层Animator的动画事件，只订阅一个animator。扩展使用AnimatorCTR)



**PopFgStart等提示牌弹窗：**提示牌都是此结构



### 配置数字、分类游戏面板

**说明：**在TA\_GameManager上面分配



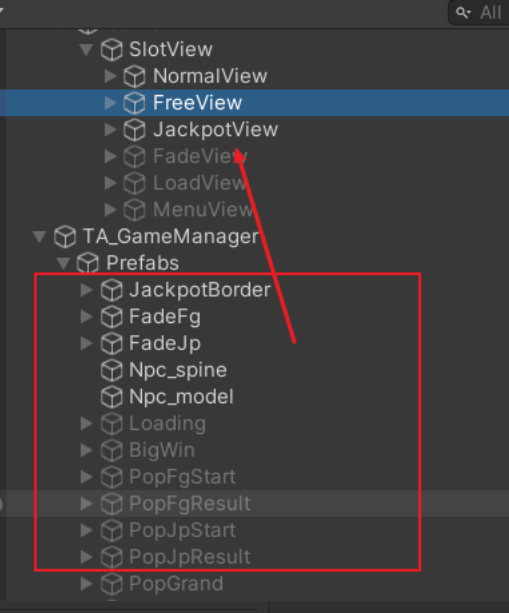
**数字：**一般是12张图片(0-9，。)

****

**面板分类(GamePanelClassify)：**方便制作、转场动画切换。哪里会出现的物体就放哪个列表，都会出现就都放。

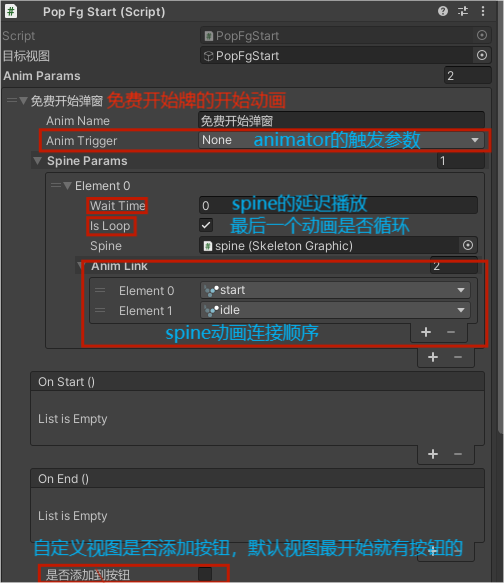


### 移动到程序的结构



## 步骤3-配置播放参数

**说明：**

* 每个视图都有专门的脚本控制。而视图都有一个或多个动画，比如一个弹窗有开始和退出（待机一般是接着开始的）。
* 每个动画播放参数有animator的触发参数、spine的动画名参数
* 默认的视图都已经有默认的参数数据了，在Scripts\TA\Frame\Resources\Data\BaseViewData，会自动读取。
* 如果动画资源命名不是按照默认设置的，则需要在每个对应视图脚本去修改参数
* 触发器为None时，操作是将视图激活
* Ps：有时候spine播放延迟不好调试的话，可以使用Animation事件，在spine上挂载SpinePlayController，配置参数调用方法。也可以使用自己的播放实现脚本

## 步骤4-远行点击按钮播放

空

## 以上是Slot的制作流程-以下是补充说明

### GameEvent类

**说明：**TA\_GameManager上，所有的视图动画注册在这个事件列表。

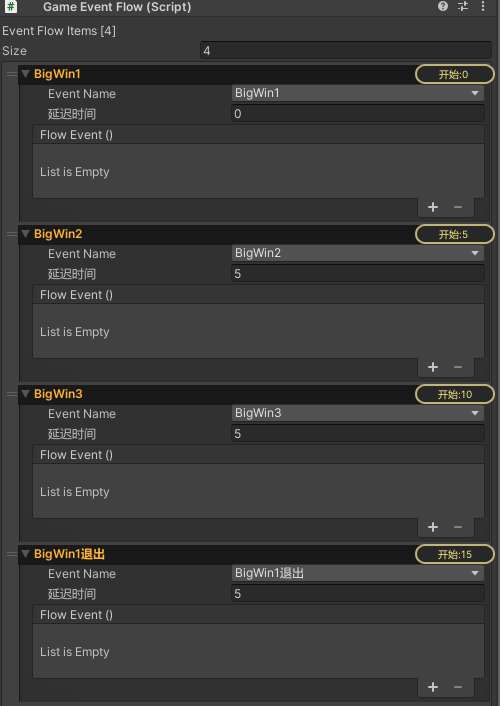


### DebugView按钮

**说明：**Debug按钮事件是依据按钮的物体名和动画事件名对应，将TA\_GameManager上的事件自动绑定。

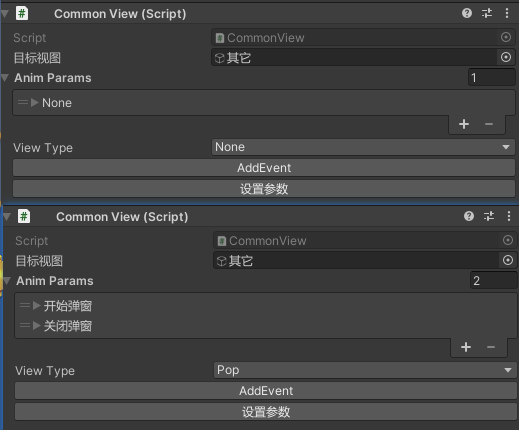
### 游戏流程

选择GameEvent上的事件连接在一起，下面是设置好的BigWin播放，



### 自定义视图创建

默认的视图一般够用了，如果需要创建新的，则复制在Views下面的其它弹窗视图或其它弹窗，修改配置里面的参数，点击AddEvent编辑时添加事件



### 脚本实现玩法和动画

继承BaseScriptAnim类，实现RegisterFun和Init方法。点击ScriptAnims物体上ScriptAnimManager的注册脚本事件按钮(会将事件添加到GameEvent)

