**Operation** **Sysetem**

**作者：刘连友**

**时间：2021/4/26**

**版本：1.0**

1. **背景**

我们的TriggerEventSystem的基本执行单元是TriggerTask，在这个Task里面会有非常多得固定管线（Start->Prepare->Active->Finish->PostFinishe->End），这些管线都有自己的状态枚举值来表示，并且还有很多的变化状态(Pause, Resume, Stop, Sleep, Skip, Reset),这些状态都有一个对应的函数接口能够让子类自定义，或者有一个抛出事件能都让外部的模块添加自己的自定义的内容。

在应用里面需求是变化的，并且是不可统一预测的，因此这个就需要一个方法能够让这些接口或者事件知道应用信息用以区分不同的场景数据，能够做出相对应的策略。

就比如说一个TriggerTask的Start，可以是自己启动，也可以是被其他Trigger通知启动，也可以是被玩家碰到后启动，也可以是其他的游戏自定义。那这些不同的启动方式就需要一个策略来区分他们了的。这个区分的策略就是这个文档功能要做的。

1. **功能概述**

简单来说这个功能就是一句话：

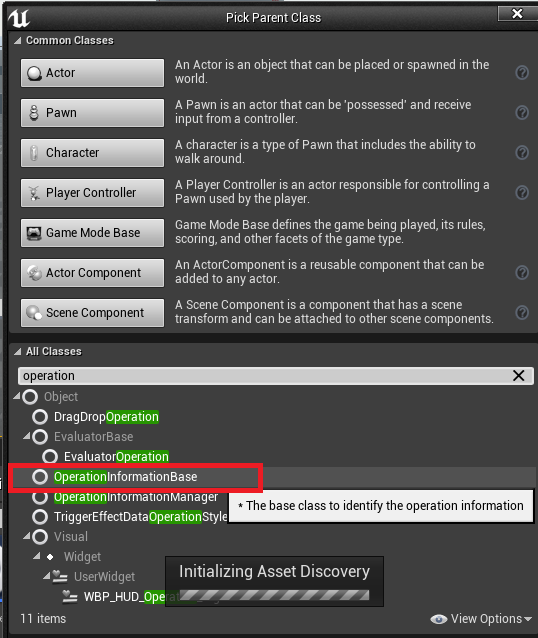
谁在何时何地以什么样的方式对目标做了什么样的操作。

就是常常说的：

**when, where, and how the target get one operation from who.**

1. **使用&功能**
2. 创建一个新的OperationInformationBase BP资源。

OperationInfo 里面包含了多个实际操作数。

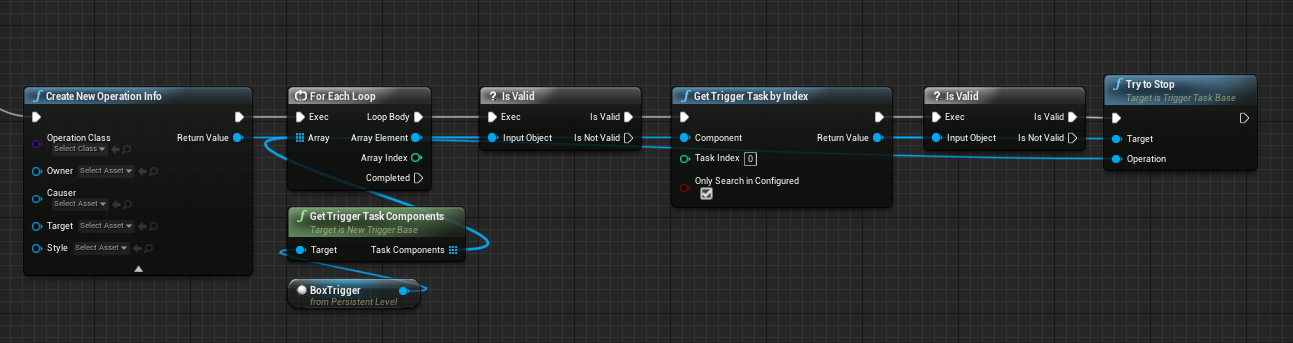


1. 然后在使用的地方调用CreateNewOperationInfo（默认的，有些Native的Operation可能有自己独有的创建方式）创建这个实例。

并且进行赋值，这里需要调用的几个函数有：

* AddNewOperation：添加一个新的操作方式，
* RemoveOperation，移除一个新的操作方式
* FindOperationScope：找到指定的操作
* CreateNewOperationInfo：创建一个新的OperationInfo。
* GetConstOperations：获得操作信息的不可改动的引用
* GetOperations：获得所有操作方式

1. 将这个实例赋值给想要使用的函数，比如图中的Stop 函数



这里只是用来显示一个新的OperationInfo该如何使用的方式，具体的还要根据实际需求添加其他的新的节点的。