

通俗点讲，有限状态机是：将对象的状态（攻击、闲置、晕眩等状态）的实现代码，提取出来，封装成状态。由状态机负责在各个状态之间调度。状态的切换由具体状态类满足条件自动触发、或者手动触发都可以。

对象持有状态管理类（状态机）的引用，与具体的状态解耦。而混合树是多个动作的一种组合，其实它也是一种特殊的状态。

注意！！：

区分状态转移和混合树是很重要的。虽然两者都被用来制作平滑动画，但是他们是分别在不同的情况下使用的。状态转移是在确定的时间内从一个动画状态平滑的转移至另一个时使用的。状态转移被认定为动画状态机的一部分。如果一个动作跳转到另一个完全不同的动作耗时很短，那么状态转移通常的表现令人满意。

动画混合树，是通过使用不同的角度，将角色不同的部分混合，然后达到平滑混合不同的动画的目的。他会用一个有动画控制器有关的、量化的混合参数来控制各个动作相对于最终效果的权重。为了使混合过后的动作起作用，必须混合**造型和时间相近的动作。动画混合树在动画状态机中是一个特殊类型的状态。**

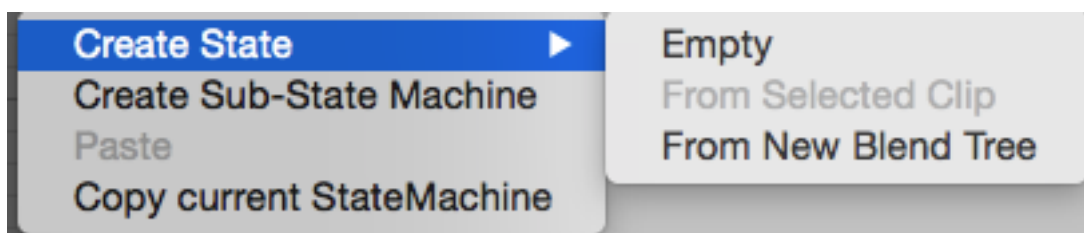
在了解以上信息之后。我们来看下Unity中的状态机。以下是状态机的参数说明。

名称	说明
State Machine	动画状态机，可包含若干个状态单元
Sub-State Machine	子动画状态机，可包含若干个状态单元子动画状态机
Blend Tree	混合树
Any State	任何状态
Entry	本动画状态机的入口
Exit	本动画状态机的出口

每一个动画控制都可以有若干个动画层，每个动画层都是一个动画状态机，动画状态机中可以同时包含若干个动画状态单元或子动画状态机。在Unity5.x版本中，每个动画状态机都必然会含有”Any State”、“Entry”、“Exit”动画状态单元，用于实现该状态机不同状态切换的必须功能。

### 5.3 动画状态单元搭建(状态机的添加)

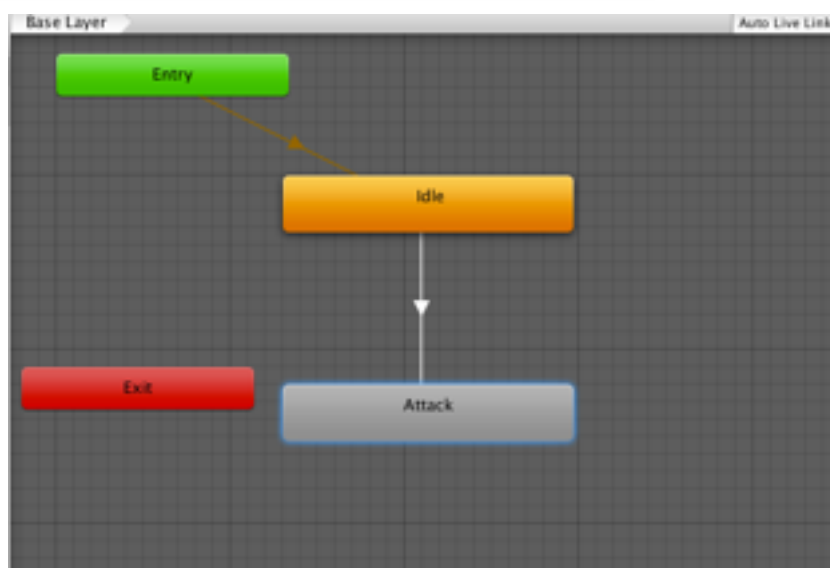
(1)在Project里面按右键新增一个Animator Controller，双击动画控制器文件打开动画控制器的专用窗口。在窗口的空白处按下右键选择Create State->Empty生成一个新的状态，如图所示。可以更换这个状态称。



(2)创建好的状态如下所示。选中该状态，右侧检视视图可以查看此状态的信息。其中Motion代表此状态播放的动画片段，可以为此状态添加一个动画。Speed为动画片段播放的速度。也可以直接将一个动画片段拖入到动画控制器窗口中。(黄色的状态为默认状态，其他的显示为灰色)




(3)选中一个状态右键Make Transition，会多出一条灰色线，选中另外一个状态就能完成一个状态到另外一个状态的过度，比如这里的Idle到Attack的过度，如下所示，箭头指向为目标状态。



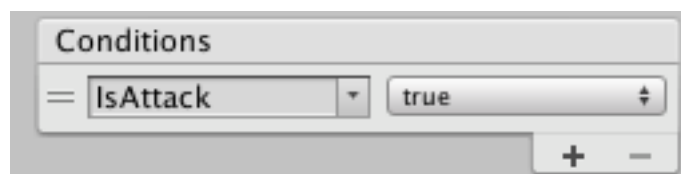
## 5.4 过度条件的参数设置

动画状态机和过度条件搭建完成后，就需要对状态机间的过度条件进行设置。为了实现对各个过度条件的操控，需要创建一个或多个参数与之搭配。Mecanim支持的过度参数类型有Float、Int、Bool、Trigger，其在动画控制器代表的意义需要游戏动画师提前设计好。

以下为过度条件参数的设置步骤。

(1)设置过度条件的参数，首先需要添加参数，在动画控制器窗口，选中Parameters，点击号，就可以添加一个参数。我们以Float类型为例，添加一个Bool类型的参数变量IsAttack。

(2)选中我们以上例子中从Idle到Attack状态的过度的一条线，添加刚才(1)步骤的设置的参数，如下。



(3)如果跳转到攻击状态还需要其他参数，可以继续按照步骤(1)(2)手动添加。

## 5.5 代码对动画控制器的控制

案例见课上安排。

## 6 角色动画的重定向

在上节我们学习了Avatar的创建和配置，可能还并不知道创建的Avatar具有什么意义。

人形角色模型绑定的骨骼架构所包含的骨骼数量和名称不尽相同，难以实现动画的通用，为了解决这一问题，Mecanim动画系统提供了一套简化过的人形角色骨骼架构，而Avatar文件就是模型骨骼架构与系统自带

骨骼架构间的桥梁，重定向的模型骨骼架构都要通过Avatar与自带骨骼架构搭建映射。

映射后的模型骨骼可能通过Avatar驱动系统自带骨骼运动，这样产生一套通用的骨骼动画，其他角色模型只需借助这套通用的骨骼动画，就可以做出与模型相同的动作，即实现角色动画的重定向。通过这项技术的运用，**可以极大地减小开发者的工作量，以及项目文件和安装包的大小。**

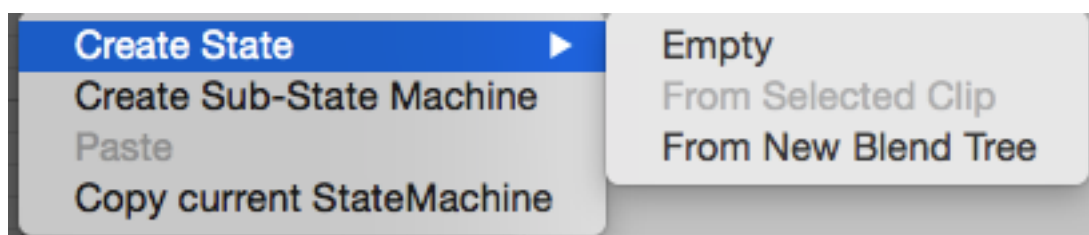
具体做法是，对具有阿凡达骨骼配置文件的模型，在将模型导入拖入到游戏场景后，在其Animator组件下，修改其Animator Controller为其他动画控制器。运行即可查看到效果。

## 7 混合树的添加

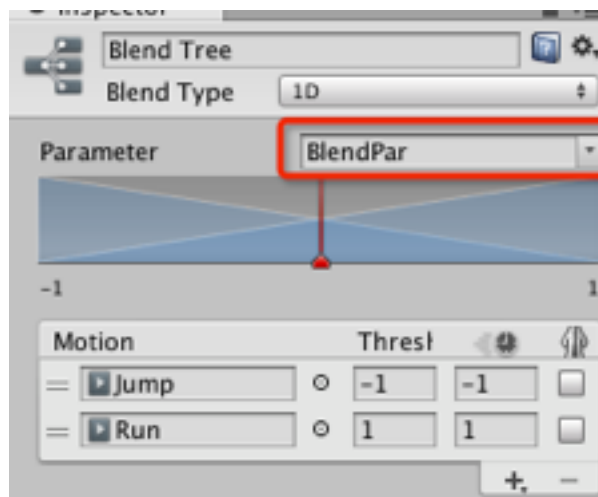
实际的游戏开发过程中，有时候会有将两个动画混合成一个动画的需求。那么在4.0之前，如此功能则必须再次 创建一套动画才行，而新的Mecanim动画系统为开发提供另一种途径，那便是接下来所要学习的动画混合。

在前面我们已经学习了动画混合树的概念。那么接下来，我们只需要学习如何在动画控制器窗口中添加混合树即可。

(1)打开新建的动画控制器。鼠标右键Create->From New Blend Tree(如下)。



(2)双击新建的混合树，再次右键Add motion，可以指定一个动画片段。默认情况下，在创建混合树的时候会添加一个Float类型的参数。混合树默认情况下必须要有至少一个参数，且是Float类型。如下所示。



改变Thresholds，即可改变每个动画的触发数值。以上为1D混合，还有2D混合，2D混合表示混合树有两个控制变量。

## 8 扩展知识(StateMachineBehavior类)

到了Unity5.0之后，开发人员可以为动画状态机或动画状态机单元添加继承于StateMachineBehavior类的脚本，用于在指定动画的播放过程中进行自定义操作。详细如下。

OnStateEnter(Animator animator,AnimatorStateInfo stateInfo,int layIndex)	当动画开始播放时调用一次
OnStateUpdate(Animator animator,AnimatorStateInfo stateInfo,int layIndex)	动画播放之后的每一帧都调用一次
OnStateExit(Animator animator,AnimatorStateInfo stateInfo,int layIndex)	动画播放结束时调用一次
OnStateMove(Animator animator,AnimatorStateInfo stateInfo,int layIndex)	动画被移动时播放
OnStateIK(Animator animator,AnimatorStateInfo stateInfo,int layIndex)	动画触发逆向运动学调用