

CAE Document

Website: <https://www.supercline.com/>

Email: xfan_hsl@aliyun.com

Version: 2017.4.16f1 or higher

一、Project Organization

- ActionEngine [common code for skill fixed pipeline]
- Demo [demo project]
 - Media [art resource]
 - Resource [programmer resource with prefab and json data]
 - GameData [json data which are generated by action editor]
 - Action
 - ActionInterrupt
 - _Unit
 - Scene [demo scene, launcher for starting]
- Editor [action editor resource and source code]
 - Layout [default editor layout for user]
 - Materials [editor scene with material]
 - Scene [editor scene]
 - Script [editor script]
 - Skins [editor skin]
 - Textures [editor texture]

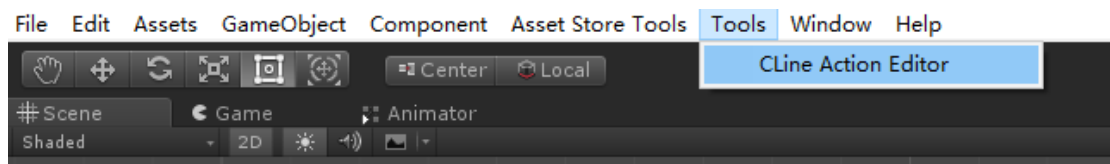
二、 Demo

Just run /Demo/Scene/Launcher.unity.

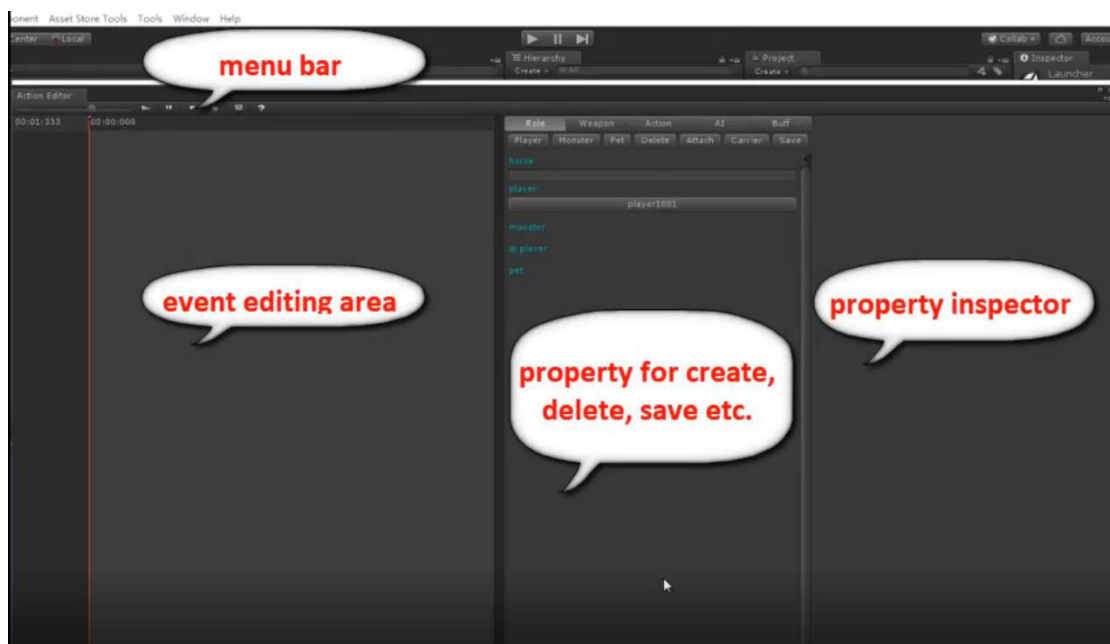
Please pay attention to json data which are generated by action editor, you can open editor to how to using editor.

三、 How To Use Action Editor By Designers

1. click Menu/Tools/Cline Action Editor to open action editor



2. as you can see, the editor main UI looks like:



it consists of four components, the menu bar, event edit panel, property edit panel, property inspector.

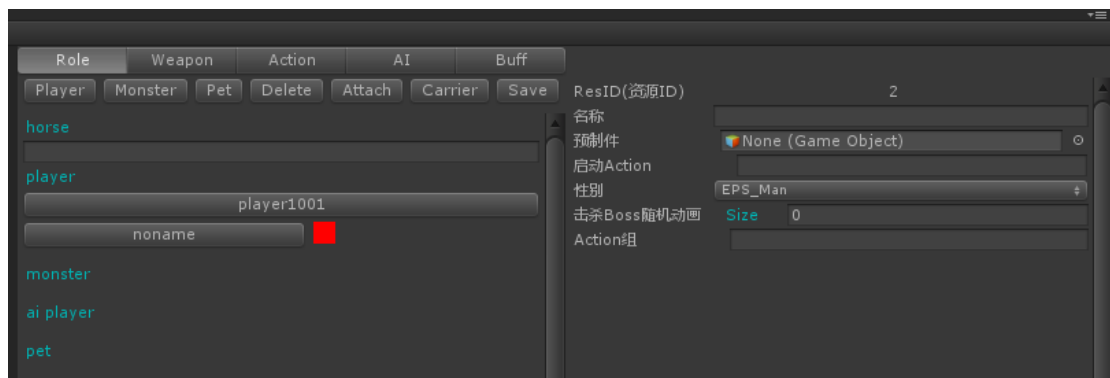
menu bar: play, stop, play with frame, save, help

event edit panel: you can add your customize event to pipeline

property edit panel: edit player, pet, monst, weapon, action etc. property with create, delete, save.

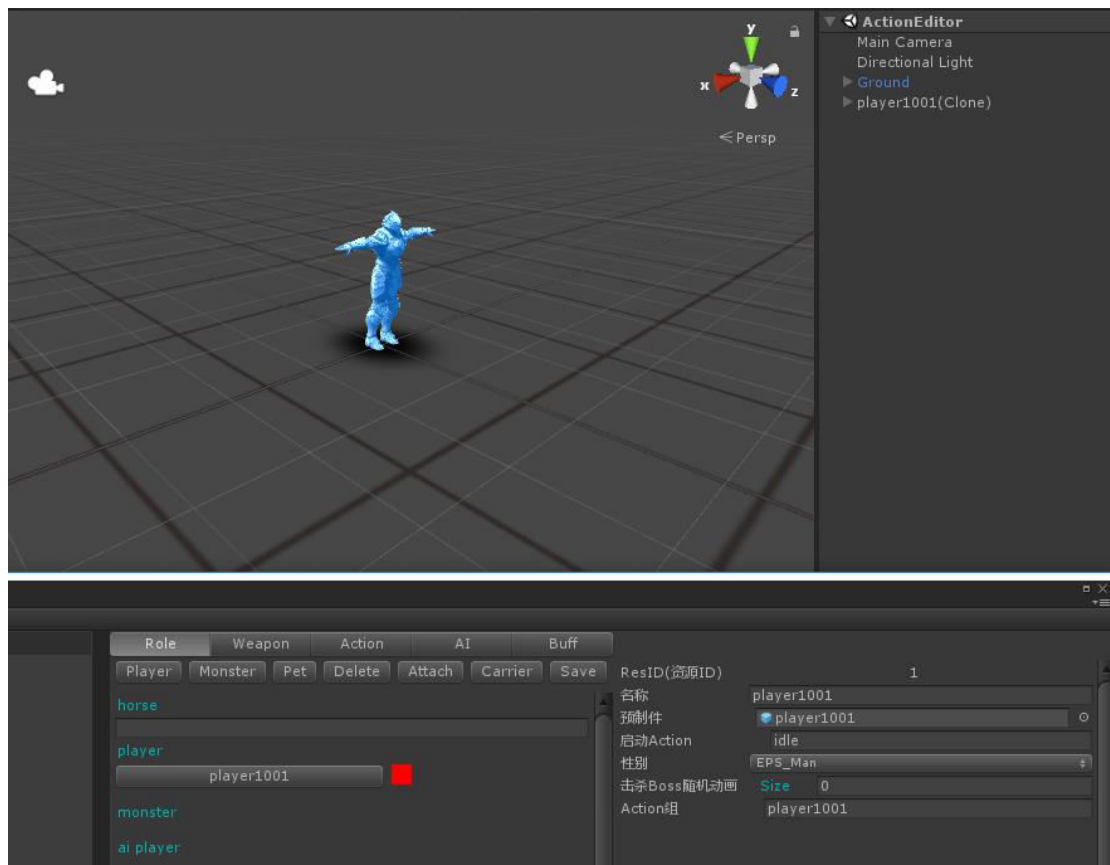
property inspector: inspector every thing just like unity inspector

3. select **Role** property, and click **Player** button to create player property with 'noname' template, and select it looks like:

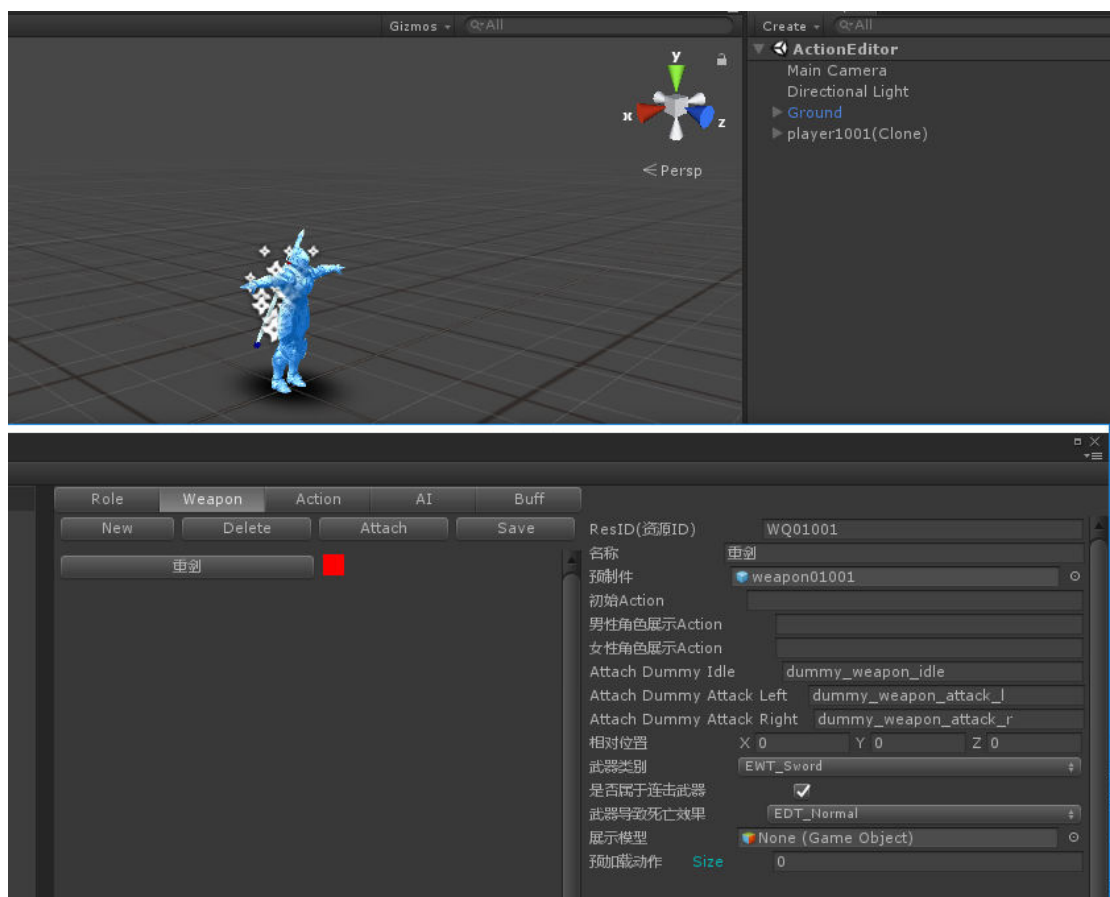


and you can edit player property on inspector on the right

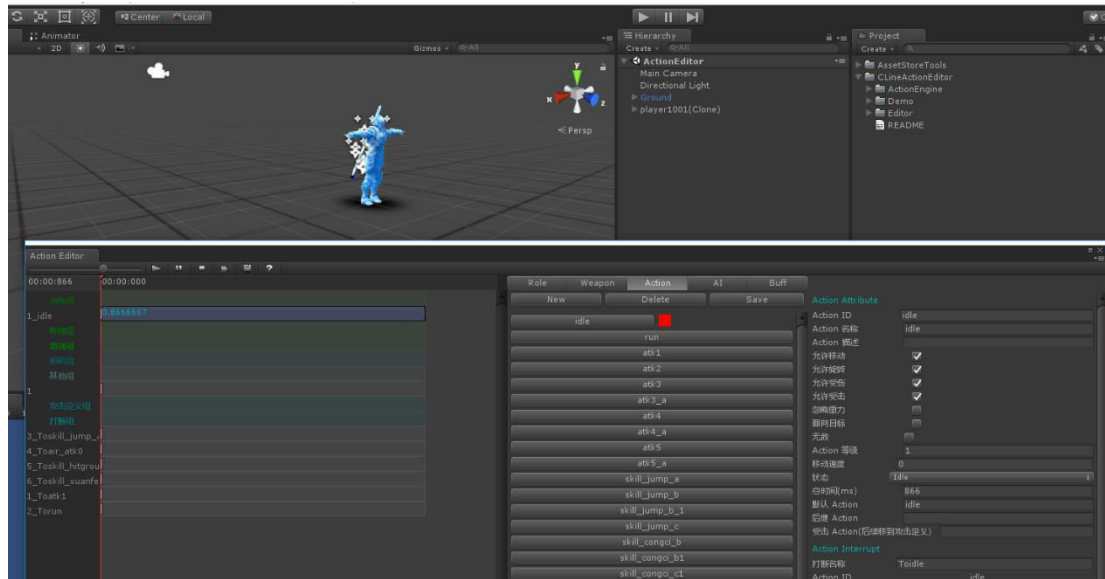
4. suppose you have edited the properties as like 'player1001' and attach to create player in the editor scene, later it will be used to play skills.



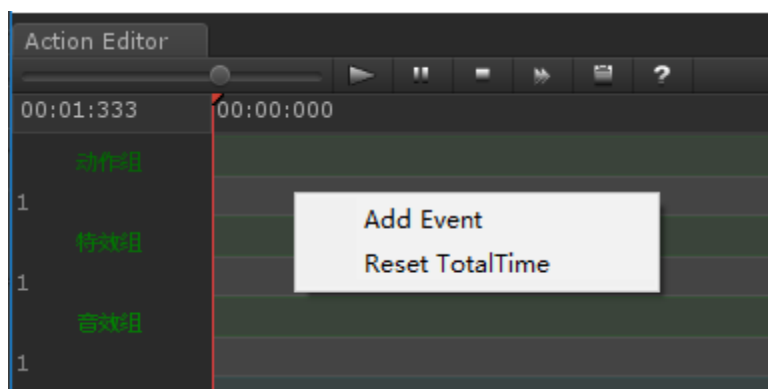
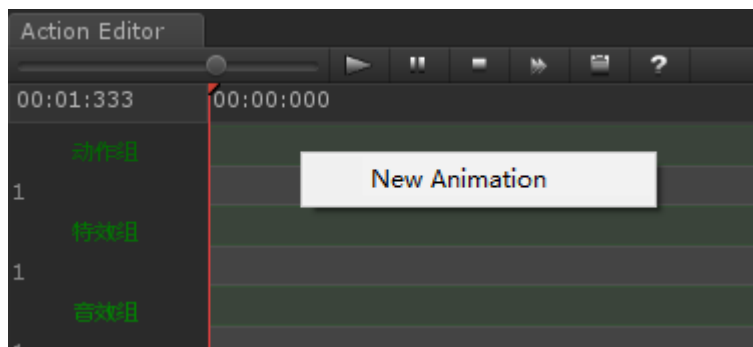
5. next create the weapon as above, and attach again

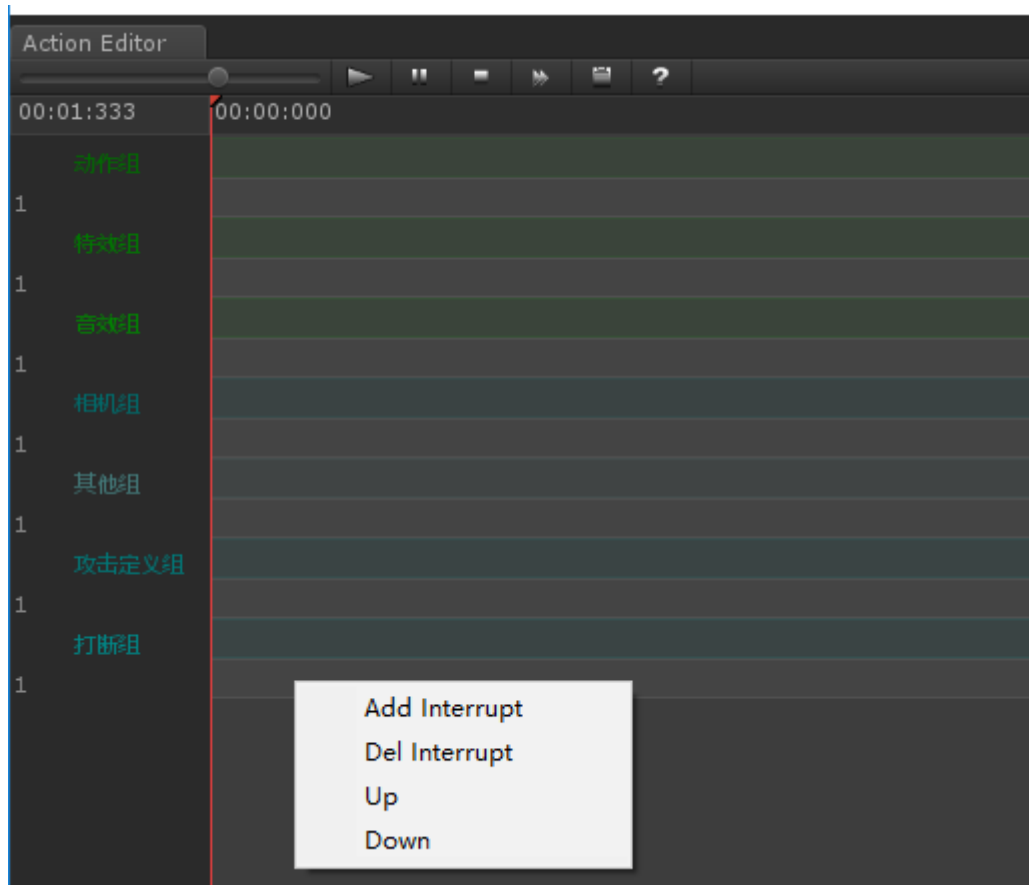
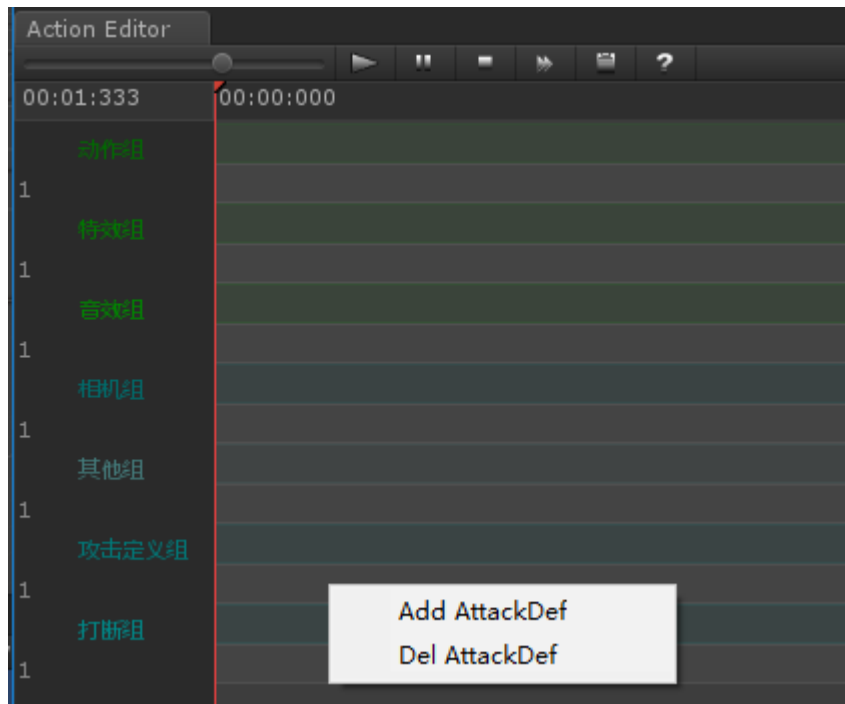


6.now create action as above, and select 'idle' action, it looks like:



6.1 event edit panel with action, when you right click on any area on the left, there will be a lot of tips look like:





6.2 property edit panel will show all skills looks like:



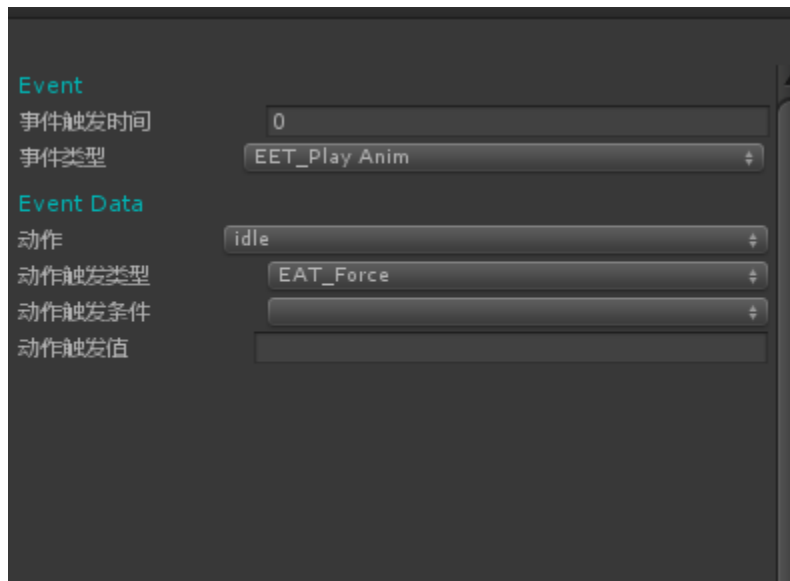
6.3 property inspector will show user custom property looks like:

Action Attribute

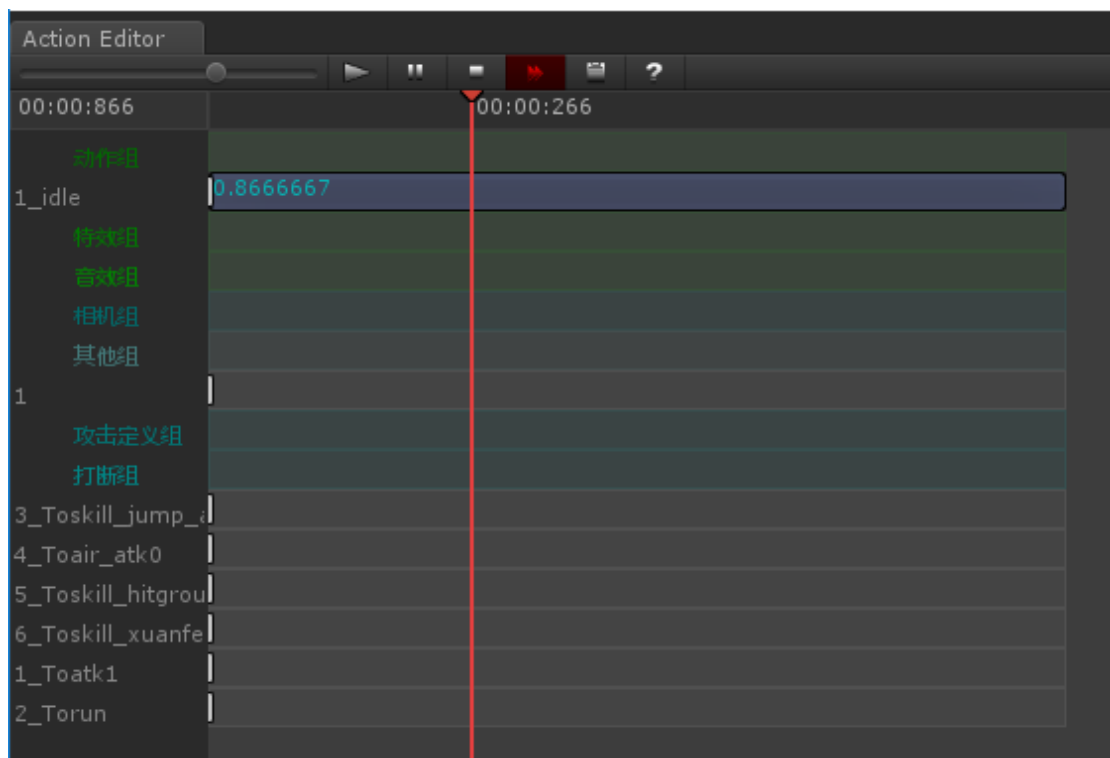
Action ID	idle
Action 名称	idle
Action 描述	
允许移动	<input checked="" type="checkbox"/>
允许旋转	<input checked="" type="checkbox"/>
允许受伤	<input checked="" type="checkbox"/>
允许受击	<input checked="" type="checkbox"/>
忽略重力	<input type="checkbox"/>
面向目标	<input type="checkbox"/>
无敌	<input type="checkbox"/>
Action 等级	1
移动速度	0
状态	Idle
总时间(ms)	866
默认 Action	idle
后继 Action	
受击 Action(后续移到攻击定义)	

Action Interrupt

打断名称	Toidle
Action ID	idle
打断开启	<input checked="" type="checkbox"/>
打断时间	0
检查所有条件	<input checked="" type="checkbox"/>
判断死亡	<input type="checkbox"/>
判断目标存在	<input type="checkbox"/>
判断目标不存在	<input type="checkbox"/>
判断是否着地	<input type="checkbox"/>
着地比较值	<input type="checkbox"/>
判断闪身输入	<input checked="" type="checkbox"/>
闪身输入比较值	<input type="checkbox"/>
判断移动输入	<input checked="" type="checkbox"/>
移动输入比较值	<input type="checkbox"/>
判断跳跃输入	<input checked="" type="checkbox"/>
跳跃输入比较值	<input type="checkbox"/>
判断攻击输入	<input checked="" type="checkbox"/>



7. finally, you can click play (shortcut - number 1) or frame-by-frame play button (shortcut - number 2) to preview the skill in real time.



ok, let's enjoy the action editor.

四、How To Extend Action Editor By Programmers

1. we have customized the EditorProperty property reflection mechanism, and even the reflection mechanism you can extend, as follows

```
public enum EditorPropertyType
{
    EEPT_Bool = 1,
    EEPT_Int,
    EEPT_Float,
    EEPT_String,
    EEPT_Vector2,
    EEPT_Vector3,
    EEPT_Vector4,
    EEPT_Color,
    EEPT_Quaternion,

    EEPT_Transform,
    EEPT_Texture,
    EEPT_Material,

    EEPT_GameObject,          // [GameObject => GameObject]
    EEPT_GameObjectToString, // display at GameObject, configure at string [GameObject => string]
    EEPT_TransformToString,

    EEPT_Enum,

    EEPT_AnimatorStateToString,
    EEPT_AnimatorParamToString,

    EEPT_CustomPropertyToString,

    // for AI
    EEPT_AICondition,
    EEPT_AIVariable,

    // for list
    EEPT_GameObjectToStringList,
    EEPT_StringList,
}
```

2. You can customize anything, just add the EditorProperty keyword as follows:

```

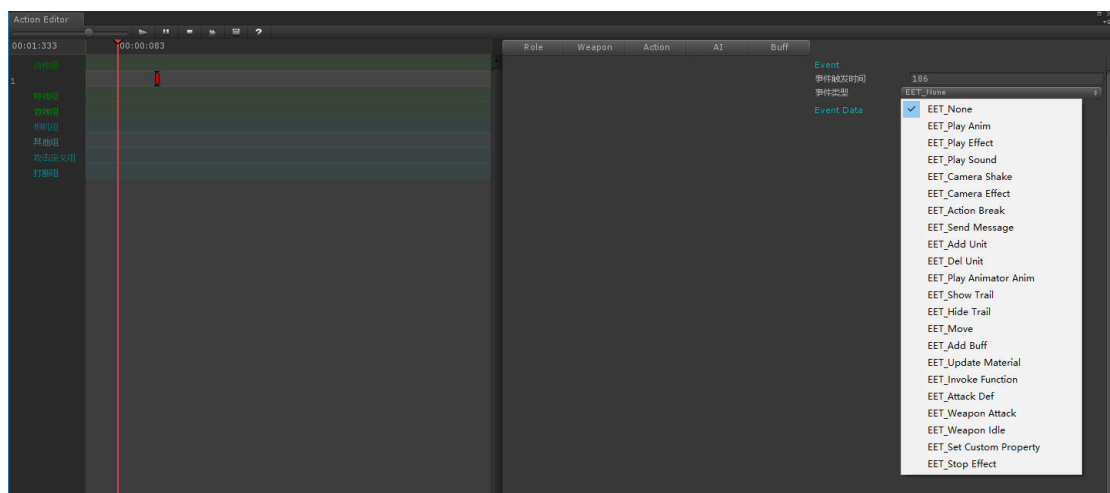
public sealed class Event
{
    private int mTriggerTime;
    private EEventType mEventType;
    private EventData mEventData;
    private string mActorID;

    [EditorProperty("事件触发时间", EditorPropertyType.EEPT_Int)]
    public int TriggerTime
    {
        get { return mTriggerTime; }
        set { mTriggerTime = value; }
    }

    [EditorProperty("事件类型", EditorPropertyType.EEPT_Enum)]
    public EEventType EventType
    {
        get { return mEventType; }
        set
        {
            if (mEventType != value)
            {
                mEventType = value;
                switch (mEventType)
                {
                    case EEventType.EET_None:
                        mEventData = null;
                        break;
                    case EEventType.EET_PlayAnim:
                        mEventData = new EventPlayAnim();
                        break;
                    case EEventType.EET_PlayEffect:
                        mEventData = new EventPlayEffect();
                        break;
                    case EEventType.EET_PlaySound:
                        mEventData = new EventPlaySound();
                        break;
                    case EEventType.EET_CameraShake:
                        mEventData = new EventCameraShake();
                        break;
                    case EEventType.EET_CameraEffect:
                        mEventData = new EventCameraEffect();
                        break;
                }
            }
        }
    }
}

```

3. select any event and the event property show in inspector as follows:



That's all, thank you for purchasing action editor, I will do my best to

serve you.