//通用配置

class LightingEffect

{

["光效ID] nID = 0;

["光效描述"] strDesc = "";

[等待切换名称为此的状态机才开始计时，暂时没用] waitForState = "";

[是否全局光效，全局光效则不依赖于某个人物生存] bGlobal = false;

[一个人物是否只能有一个光效，用于光效反复发射时不叠加效果] bSingle = false;

[是否需要速度缩放(是否需要速度缩放,有的光效不需要速度缩放，也能正常，缩放了反而不正常，比如根据距离产生粒子的特效] bUseSpeedScale = true;

[施放者死亡时，关闭此光效。此功能适用于一次性类光效（例如技能），不建议用于BUFF等永久类光效，因为复活时不会自动加回] CloseWhenDead = false;

[Desc("启用距离优化")]

public bool EnableDistanceOptimtized = true;

[Desc("是否忽略单一光效优化")]

public bool bIgnoreSingleEffectOptimtized = false;

[Desc("强制缓存通用光效，一般通用光效都不会去缓存的")]

public bool NeedCache = false;

}

//绑定类光效

class LightingEffect\_Bind

{

[光效预制体] effectPrefb = null;

[光效统一缩放比例] scale=1.0f;

[光效开始播放时间（单位毫秒）] effectStartTick = 0;

[光效存在时间（单位毫秒）] effectLiveTick = 3000;-1为无限循环，适用于BUFF光效

[光效最长生命时间，超过这个时间必须销毁（单位毫秒）")] effectTotalTick = 3000;

[光效位置绑定骨骼名字(建议写全路径)] posBindBone;

[光效相对骨骼的偏移] posOffset;

[光效旋转绑定骨骼名字(建议写全路径)] rotationBindBone;

[光效相对绑定骨骼的旋转] rotationOffset;

[光效面向镜头] faceToCamera=false;

[光效向镜头偏移到物体外部] offsetToSurface = false;

[光效绑定类型：StayAlone创建后不再运动；BindWithBone创建后跟随骨骼运动] BindType bindType = BindType.StayAlone;

[光效面向，勾选后背向对手] faceAganistTarget = false;

[将target绑定到source上，用于穿刺效果] bindTargetToSource = false;

[将target绑定到source上时target的骨骼名] bindTargetBone = "";

}

//飞行类光效

class LightingEffect\_Fly :

{

[光效预制体] effectPrefb = null;

[光效统一缩放比例] scale = 1.0f;

[光效开始播放时间（单位毫秒）] effectStartTick = 0;

[光效存在时间（单位毫秒）] effectLiveTick = 3000;

[光效最长生命时间，超过这个时间必须销毁（单位毫秒）] effectTotalTick = 3000;

[源点和目标点最大距离，超过距离关闭，0时无效] maxDistance=0.0f;

[光效位置绑定骨骼名字(建议写全路径)] posBindBone;

[光效旋转绑定骨骼名字(建议写全路径)]string rotationBindBone;

[光效相对骨骼的偏移] posOffset;

[目标点的偏移] Vector3 targetOffset=Vector3.zero;

[光效飞行类型：TargetObject飞向目标物体（追踪）；TargetPosition飞向目标位置（定点）] flyType = FlyType.TargetObject;

[光效是否始终面向目标] faceTarget=true;

[光效相对地面最低高度，为了使得光效尽量不穿到地面下，如果是负数则此设定无效。] lowestHeight=-1.0f;

[光效相对地面最高高度，为了使得光效贴地，如果是负数则此设定无效。] highestHeight = -1.0f;

[光效飞行速度（可以等于0）] flySpeed = 5.0f;

[光效到达目的地时是否消失] destroyWhenReached = true;

[对手死亡时关闭光效] destroyWhenTargetDead = false;

[光效是否进行碰撞检测] useCollision = false;

[光效产生碰撞时是否消失] destroyWhenCollision = true;

[到达目的地后返回] effectReturn = false;

[击中人时的后续光效] string effectIDWhenHitPlayer = "";

[击中怪物时的后续光效] string effectIDWhenHitMonster = "";

[击中地面时的后续光效] string effectIDWhenHitGround = "";

[飞到最远处的后续光效] string effectIDWhenReached = "";

[使用碰撞盒检测光效是否到达] triggerHitDetect = false;

[结束时后续动作] playAnimWhenClose = "";

[是否做上抛运动] upcastMotion = false;

[上抛运动的高度] upcastHeight = 0.5f;

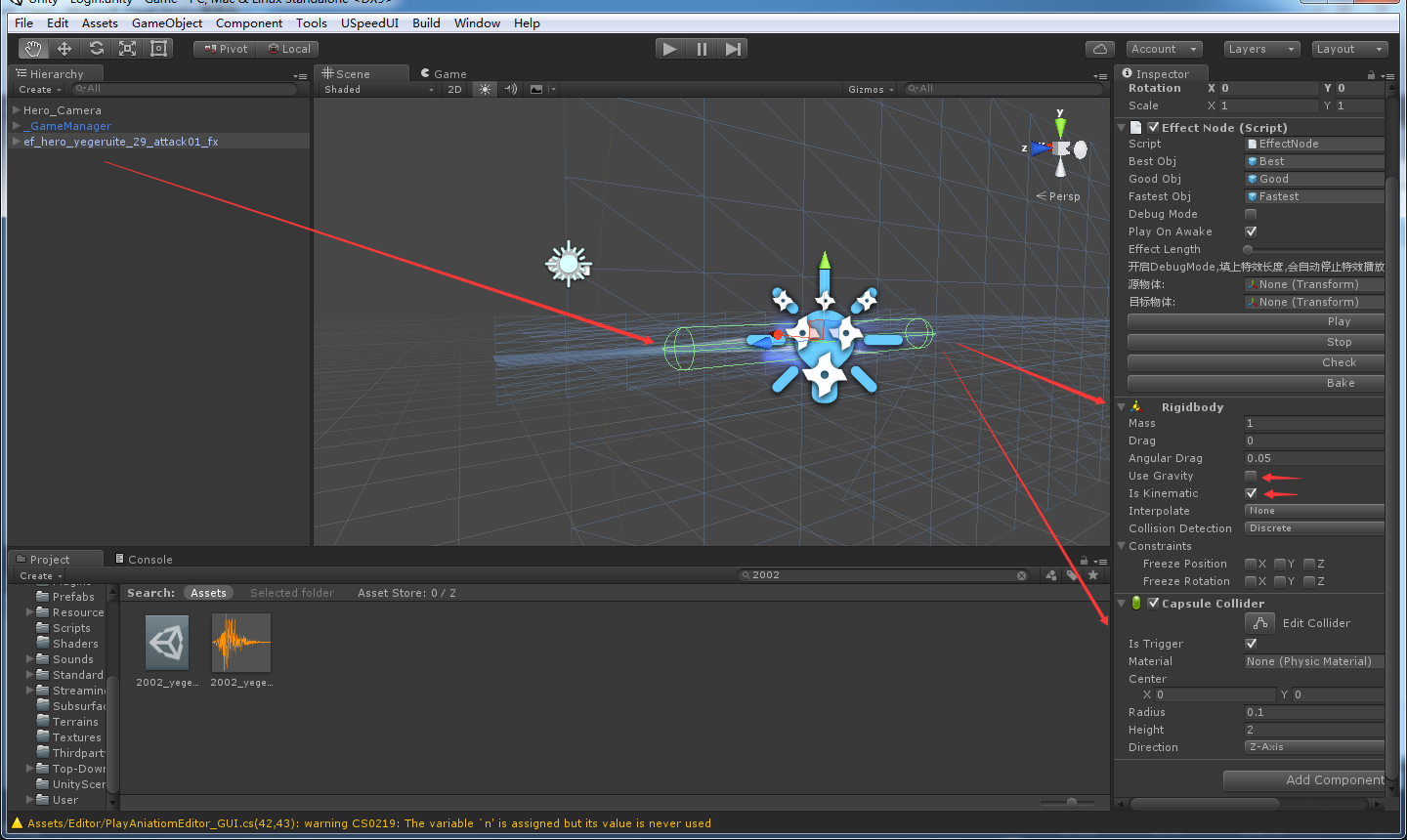
[光效延迟消失的时间（单位毫秒），当立即消失destroyWhenReached为无效时才生效，0表示不起效")] delayDestroyTick = 0;

[光效是否会影响护盾，当光效装到护盾上时，会关闭这个光效] AffectedShield = true;

[光效延迟消失的时候隐藏美术指定部件多少毫秒，必须小于delayDestroyTick] delayHideTick = 0;

[光效延迟消失时吸附在目标上] attackToObject = false;

}



飞行类光效如果想要triggerHitDetect 生效，必须给光效预制体配置物理碰撞体。在光效预制体的最上层添加Rigidbody组件以及任何3D形状的Collider组件。Rigidbody中的Use Gravity不勾，Is Kinematic要勾。Collider也是是任何3D形状（Box,Capsule,Sphere,...）,Is Trigger要勾，调整形状参数至想要的状态。

//范围类光效

class LightingEffect\_Area

{

[光效开始播放时间（单位毫秒）] effectStartTick = 0;

[诞生光效间隔时间（单位毫秒）] effectIntervalTick = 3000;

[诞生光效ID] subEffectID = -1;

[光效位置绑定骨骼名字(建议写全路径)] posBindBone;

[光效指定相对落地点的位移（指定后其他初始点无效）] customOffset;

[光效范围是投影到地面的，用于指定地面范围光效] projectOnGround=true;

[光效距离一直延长到最大距离，用于设计与地面有爆炸效果的光效] extentToMaxDistance = true;

[光效数量] effectCount = 10;

[光效最长生命时间，超过这个时间必须销毁（单位毫秒）] effectTotalTick = 3000;

}

//摄像机控制特效

class LightingEffect\_Camera

{

[以动画形式制作的镜头动画效果] animationClip = null;

[动画镜头的原点取的对象] animationOrgPos

(Hero动画原点取动画开始时英雄的位置和朝向;

FollowHero动画原点跟着英雄移动;

TargetPos动画原点取调用参数传过来的TargetPos;

SourceObject动画原点取调用参数传过来的SourceObject)

[动画镜头的原点需要贴地] animationOrgProjectFloor = false;

[动画镜头强制取主摄像机（其他UI之类的场景需要挂镜头动画时请勾选）]

animationMainCamera = false;

[震屏光效需要附加到哪些人身上] attackType;

[震屏光效影响范围,0表示无视范围] effectRad = 10.0f;

[光效存在时间（单位毫秒）] effectLiveTick = 3000;

[光效延迟开始时间（单位毫秒）] delayTick = 0;

[光效中断时摄像机恢复用的秒数（单位毫秒）]effectFinishTick = 500;

[结束时是否恢复一开始的摄像机] recoverTransform = false;

[摄像机移动关键帧节点，可以有多个，展开后配置EffectCameraNode]cameraAxis;

}

//某个关键帧的摄像机属性

class Effect\_CameraNode

{

[到第n毫秒，所达到的效果（角度，距离等），请注意是截止点的时间，例如第一个效果的tick参数不应该是0而应该是第一个效果结束的时间点]tick;

[摄像机位置锁定]lockPosition;

[摄像机离主角Fov不变]lockFov;

[摄像机离主角距离不变]lockDistance;

[摄像机离主角平移不变]lockTran;

[摄像机离主角偏移方向不变]lockDir;

[摄像机绕主角旋转角度不变]lockAngle;

[摄像机离主角距离]distance;

[摄像机离观察点位置] pos;

[摄像机离观察点平移] tran;

[摄像机离主角方向偏移]dir;

[摄像机绕主角旋转角度]angle;

[摄像机视锥角，可用于镜头拉近拉远] fov;

[摄像机特效时，玩家是否能操控摄像机和角色] playerControl=true;

[摄像机是否震动] shakeCamera=false;

[摄像机震动时的配置参数，展开后配置Effect\_ShakeCamera]shakeParam;

}

//摄像机震动配置参数

class Effect\_ShakeCamera

{

[震动最小速度] minShakeSpeed;

[震动最大速度] maxShakeSpeed;

[震动最小幅度] minShake;

[震动最大幅度] maxShake = 2.0f;

[震动最少次数] minShakeTimes;

[震动最多次数] maxShakeTimes;

}

///残影效果参数

public class LightingEffect\_Shadow : LightingEffect

{

[开始时刻，单位毫秒] nStartTick;

[结束时刻，单位毫秒] nEndTick;

[每个残影开始的透明度] fStartAlpha=1.0;

[每个残影存在的时间，单位毫秒] nShadowMaintainTick;

[每个残影淡出的时间，单位毫秒] nShadowFadeoutTick;

[插入残影的时间间隔，单位毫秒，大于0时关键帧失效] shadowIntervalTicks=0;

[需要插入残影的关键帧，单位毫秒] shadowFrameTicks;

[替换为的材质] alphaMaterial = null;

[替换为的材质的透明参数名字] alphaAttributeName = "";

[武器无残影] DisableWeapon = false;

[替换为的材质的透明参数类型] alphaAttributeType = AlphaAttributeType.Color;

[残影与残影/人物间最短间隔距离，少于这个距离则不建立残影，小于0.01f时此设置无效] minDistance = 0.0;

}

//光效优先级配置



光效按照播放优先级分为三个档：

1. Low

2.Middle

1. High

档位越高，播放优先级越高，高档位可以顶掉低档位光效

勾选强制播放后，将无视此规则，强制播放