技能时间轴

0s前端表现

1. 技能整体机制是以时间轴为核心，在不同的时间，触发不同的节点来完成一个技能的播放。
2. 节点类型分为
3. 动作节点 actionnode
4. 声音节点 soundnode
5. 震屏节点 shakecarmenode
6. 伤害节点（弃用，以服务器消息触发为准）singlehitnode
7. 放置特效节点 placefxnode（使用最多）
8. 挂接特效节点 attachfxnode(已用最多)
9. 追踪特效节点 followfxnode
10. 事件节点（未使用）eventnode
11. 移动节点（未使用）movenode
12. 发射特效节点Arrowfxnode（飞行特效 ）
13. 多次伤害节点（未使用）multhitnode
14. 时间缩放节点 scletimenode（未使用）
15. 所有节点对应的有一份配置读取类，名字一一对应，配置里面存放各个节点相关的信息，如id ,播放时间，动作名字，特效名字等 播放需要信息。技能配置分别保存在effect/role\_skill\_def.unity3d 人物 和effect/npc\_skill\_def.unity3d（npc）两个配置文件里面
16. 技能配置读取由于历史原因，最开始是json形式，后来改成ScriptableObject储存成对用的asset文件，然后打包成assetbundle读取。配置类包含关系

SkillConfig-》SkillEffectTable-》SkillEffectProp-》EffectNodeConfig-》EffectNodeProp

1. 客户端在时间轴的基础上，切分成3个状态，准备状态，攻击状态，收招状态

准备状态对应技能表格里面的准备时间（包含读条时间 如吟唱类型的读条时间）一般技能此字段为0

攻击状态是从准备时间结束到僵直时间，引导技能就是引导时间

收招状态是僵直时间之后，没有配置，一般到动作结尾

这三种状态主要用来控制技能动作的播放，在攻击状态时不可放其他技能，其他两个状态均可以放技能。实际上是僵直时间内不可以放技能。

1. 此外在做性能优化时，做了技能节点的对象池，所以每个节点都有FreeToNodePool函数用来控制节点的回收，但是由于逻辑控制原因，有的节点并未实现，每个特效节点会有额外的handle类来控制特效的管理和释放。

和服务器交互部分

1. 客户端上行请求释放技能stMultiAttackUpMagicUserCmd\_C 只有主角有 主角开始播放表现
2. 服务器下行消息 4个
3. stPrepareUseSkillSkillUserCmd\_S 准备播放技能，非主角收到此消息开始播放技能表现
4. stMultiAttackDownMagicUserCmd\_S 技能伤害包下行 收到此消息开始播放伤害表现 同步血量相关
5. stMultiAttackReturnMagicUserCmd\_S 技能使用失败 用来打断技能播放，通知失败原因
6. stSendSkillCDMagicUserCmd\_S 技能cd同步