

Buff表解析：

Effect 类型

标识Buff类型，Buff按效果来分可分为以下几种；

0普通 1冰冻 2灼烧 3减速 4禁止采集doodad 5无敌 6隐身 7潜行 8反隐身 9免疫DEBUFF 10攻击反弹 11吸收盾 12吸血 13长击飞 14伤害转治疗 15唤灵 16霸体 17禁锢 18麻痹 19中毒 20流血 21虚弱 22束缚 23化茧

Type 获益类型

标识是Buff还是Debuff；

TotalTime 生命周期

光环的可持续时间；

Interval 间隔时间

Buff触发的间隔时间；

Buff按触发方式来分，可分为两种：

1.一次性触发

Buff效果在初始化时一次性添加，直到TotalTime结束前无变化，此类Buff配置的Interval必定0；

2.间隔性触发

Buff效果在初始化时触发一次，之后每隔Interval时间再触发一次，直到Buff持续时间超过TotalTime为止；（Buff过期那个时刻不触发，比如Buff生命周期为3S,间隔1S，其会在时间点0，1，2秒的时候触发一次，第3S时刻的时候因为过期而不触发）

PS：区分Buff触发方式的方式就是直接判断Interval是否为0；

Overlap 叠加上限

Buff可叠加的上限，在叠加上限内，叠加层数越高，效果越强，超过叠加层数后，再次叠加只重置剩余时间；

StopCondition 停止条件

Buff停止的条件，满足条件后Buff将被移除；

ImmuneBuffIDs 免疫BuffID

此buff可免疫的buffID列表；

PreBuffIDs 前置BuffID

如果此配置字段不为空，假如Buff A配置有前置Buff B

- 1.若要在某单位上添加Buff A,则该单位上必须已有Buff B，才允许添加Buff A；
- 2.添加Buff A成功后，需要从该单位身上，移除其前置Buff B；

Scale 模型变化比例

当此配置不为空时，单位拥有此Buff后，会导致单位模型比例发生变化；

- 1.当单位被添加此Buff后，其模型比例会从角色配置表CharacterTableInfo.Scale变化到Scale，变化所需时间为Buff表配置值StartScaleTime；
- 2.当单位被移除此Buff后，其模型比例会从Scale变化到CharacterTableInfo.Scale；

变化所需时间为Buff表配置值EndScaleTime；

ExplodeBuff 可引爆的BuffID

当此配置不为空时，假设Buff A的ExplodeBuff是Buff B：

当单位M移除此buff后，会在场景所有敌方和友方单位中，检测是否有存在单位N，其拥有Buff B，且该Buff B是由单位M亲自施加上的，如果有，则引爆单位N身上的Buff B；

PS：当单位M是因为死亡而需要移除BuffA时，不触发BuffA的引爆效果；

ExplodeDmg 引爆伤害

接上述对ExplodeBuff 的描述，当单位N需要执行Buff B的爆炸效果时，具体造成的伤害计算方法如下：

1.获取单位M的当前物理攻击力S，和魔法攻击力P，令伤害参照值

$H = \text{Max}(S, P)$ ；

2.根据单位N上BuffB的叠加层数，和Buff B配置的ExplodeDmg，来计算出伤害比例W；

3.对单位N造成的爆炸伤害最终值 = $H * W$ ；

(Buff B被引爆后是否会移除)

补充：伤害比例W的计算方式：

```
int idx = -1;
for (int i = 0; i < explodeDmg.Count; i++)
{
    var dmgSectionMin = explodeDmg[i];
    if (overlapCountFactor <= dmgSectionMin.x)
        break;
```

```

        else if ((i + 1) >= explodeDmg.Count) //如果已到达最后一个，
则选择最后一个保底
        {
            idx = i;
            break;
        }
        else
        {
            var dmgSectionMax = explodeDmg[i + 1];
            if (overlapCountFactor > dmgSectionMin.x &&
overlapCountFactor <= dmgSectionMax.x)
            {
                idx = i;
                break;
            }
        }
    }
    return idx;
最终 W=ExplodeDmg[idx];

```

AttrIdValues4Once 一次性属性

一次性属性，是buff初次添加到单位上时，为单位添加的属性
Buff表里的如下配置参数，均为一次性属性：

speedPara：// 移动速度系数 百分比

absorbPara1：//抵挡伤害值

absorbPara2：// 转化为治疗系数 百分比

reboundPara：// 反弹伤害 百分比

ShieldPara:// 护盾

VampPara: // 吸血系数

parryPara:// 格挡减伤

AttrIdValues4Period 周期性属性

周期性属性，是buff在初次，和后续每次间隔时间后添加到单位上的属性，其是一个属性List，内容由buff配置表里的参数决定；

Buff表里的相关参数：

固定属性1	改变值1	是否基于最大值的百分比1
EffectID1	EffectValue 1	IsPercentOnMax1

上述属性组，从固定属性1到固定属性4，如配置ID为非0，则添加到上述属性List 里（规定从1到4顺序添加中，如果检测2的配置ID为0，则不予读取3和4，读取过程直接中断）；

Buff简介：

Buff分为两种:

第一种：

技能直接添加的buff，我们称之为EotEffect，其Buff来源是“技能”；

第二种：

光环Aura为处在光环范围内的单位添加的效果，我们称之为AuraEffect，其Buff来源是“光环”；

下面介绍Buff的主要逻辑流程：

(如EotEffect和AuraEffect实现细节不同，会分别阐述；如相同，会用Buff来进行通用描述)

一.Buff的添加流程

1.首先判断是否可以添加，分三类情况：

a.对于Aura，其必定可以添加；

b.对于AuraEffect，其必定可以添加，但会在这个过程中，判定其是否被抑制；

判断光环是否被抑制的方法，是与单位的“激活的AuraEffect列表”里的光环逐一进行等级比较，结果分3种情况：

i.给定光环等级小于已有光环等级，则被抑制；

ii.给定光环等级大于已有光环等级，则跳过,不做处理；

iii.给定光环等级等于已有光环等级，且存在数已达到Overlap，则被抑制；

补充说明：

单位的所有AuraEffect效果，被分为两个List，一个是“激活的”，一个是“受抑制的”，单位只受“激活的” AuraEffectList里的效果影响；

c.对于EotEffect,首先需要判断

i.若是此buff A的添加，需要攻击者拥有Buff B才可以添加成功，则检测攻击者身上是否有B，如没有，则不允许添加A；

补充说明：

Buff A的添加是否需要攻击者有Buff B,是配置在SkillConlision里的preBuffID 字段的，意思是每个碰撞盒能否给受击者挂上SkillConlision里配置的Buffs，需要看碰撞盒所属的攻击者身上，有没有SkillConlision里配置的PreBuff。

ii.若是受击单位免疫此buff，则不允许添加；判断依据是看受击单位的所有Buff的配置字段ImmuneBuffIDs是否含有待添加的buff;

iii.若待添加Buff A的Buff表配置字段PreBuffID有配置Buff B，则受击单位必须拥有BuffB，才可以添加Buff A;

2.检测是否已存在待添加Buff

设待添加Buff为A，检测是否相同的单位已有Buff为B

i.对于EotEffect, 如果A是需要攻击者有特定Buff的，则除了检测ID之外，还需要检测A与B的施放来源单位是否相同，相同才被认定为相同的EotEffect;

ii.对于AuraEffect，同时需要A与B等级相同，另外需要额外检测A与B的来源Aura是否一样，来源同一个Aura才算是相同的EotEffect;

iii.对于Aura，同时需要A与B等级相同，无额外判定条件；

3.正式添加Buff A

如果2中检测不存在待添加Buff，将Buff A 添加到直接单位上；

如果2中检测已存在与A 相同的Buff B,则判断A 是否能叠加到B：

i,对于AuraEffect,若等级相同，则A的效果变成将B的已运行时间和间隔时间，生命周期都重置为初始，但不触发效果；

ii.对于Aura,与上述AuraEffect逻辑相同，额外需要置Aura所有影响到的单位身上的AuraEffect；

iii.对于EotEffect,

如果A等级小于B，则不允许叠加；

如果A等级大于B，则先去掉B，再添加A；

如果A等级等于B，重置B的持续时间，间隔，和生命周期；如果叠加后未超过最大可叠加数，则应用一次Eot效果；如果叠加后超过最大

可叠加数，则看A是否是间隔Buff，如是，则仍然应用一次Eot效果，如是一次性Buff，则不应用；

4.添加后的处理

1.如果添加的是EotEffect类型，设其为A，则若A的ImmuneBuffIDs字段不为空，设为B，则检测单位身上是否含有B，若有则去掉；

2.如果添加的是AuraEffect类型，设其是A，若单位身上非抑制效果里含有同ID的AuraEffect，设为B，且其等级小于A，则将B转为抑制状态；

二.Buff的更新流程

主体更新流程

更新：

- 1.在刚加入Effect初始化时，会应用一次Effect效果，并重置时间；
- 2.后续加入Effect时，会执行叠加层数逻辑；
- 3.如果间隔时间到，需要再次执行一次Effect效果和时间重置；
- 4.如果是Aura，并检测到光环可以被释放了，执行光环释放效果；
- 5.如果是EotEffect,并检测到其爆炸了，则去除该效果；
- 6.如果过期了，则移除该效果；

- a.执行叠加层数逻辑，依各效果不同而不同；
- b.执行Effect效果的逻辑，依各效果不同而不同；
- c. 执行重置时间的逻辑相同，均为：

- i.对于间隔性Effect($\text{TableInfo.Interval} > 0$),

$\text{Instance.IntervalTime} += \text{TableInfo.Interval};$

ii,对于一次性Effect (TableInfo.Interval > 0)

Instance.IntervalTime += Instance.LifeTime

d.执行移除效果的逻辑，依各效果不同而不同；

1.对于Aura

Effect效果逻辑：

i.根据Aura的模式（分普通光环的固定半径模式和释放光环的可变半径模式），来确定光环半径的大小；

ii.移除光环范围内已死亡的单位，和超出范围的单位，并根据光环配置的可最大影响单位数AuraAddBuffNum，来新增可影响的单位，并向新增单位添加AuraEffect，其对新增单位Effect的影响时间为：

InstanceLifeTime - Instance.RunningTime

+ auraTableInfo.AuraLastTime;

移除逻辑：

对Aura所影响的所有单位执行去除AuraEffect效果；

2.对于EotEffect（ AuraEffect与其等同）

Effect效果逻辑：

i.如果是首次添加,则获取到该EotEffect配置属性List：

AttrIdValues4Once；

ii.如果属性是移动速度系数 反弹伤害 吸血系数 格挡减伤这四类属性，则属性值Value = Value * LapCount;(乘以叠加层数)

iii.如果属性ID 是 基础属性 ||次级属性|| 移动速度系数 ||抵挡伤害值 ||转化为治疗系数 ||反弹伤害 ||护盾 || 吸血系数 属性，则属性Value值乘以 单位的属性值 buff增益系数；

即 Value = Value * (1 + buffGain * 0.001) ；

iv: 恢复之前一次性属性值对单位属性的影响，并应用新的属性值List到单位属性；

v: 记录应用的属性值类型和具体值 List<ID ,Value>

b,在首次添加时，应用一次性状态改变

i.读取buff表的配置字段Effect，并将其转化为对应状态，应用到单位上，为单位增加对应状态属性；（如冰冻，束缚等）

c.在每次应用效果时，应用周期性效果改变AttrIdValues4Period

i.获取该AttrIdValues4Period所包含的所有的属性ID和Value组合；

ii.对所有value值乘以其有效叠加次数OverlapCount;

（PS:一次性效果的overLapCount 恒定等于1，即其Interval==0）

iii.如果是当前血量属性，Value还需要乘以单位的治疗增益系数属性；

iv:按照上述a中iii的描述，对其Value值应用Buff增益系数属性；

v:应用属性加成；

vi:对于非当前类属性（除去当前血量，当前蓝量，当前霸体值三种），记录其周期性效果的属性ID和Value；

移除逻辑：

i.在移除buff时，需要对记录的一次性效果改变，周期性效果改变，和状态改变，进行恢复；

ii,如果Buff的ExplodeBuff字段有配置，即其可以引爆其他buff,且buff所属单位未死亡，则直接引爆场景内由该单位所施加的所有携带ExplodeBuff的单位；

(只对EotEffect有效)

其他逻辑：

i.在buff存在期间，如果有外界会减少buff添加的一次性属性加成效果时，需要记录外界的消耗值，并在buff结束只扣除剩余的属性加成效果；

ii.对于AuraEffect效果，如果单位是不受抑制的，且其所属Aura配置的AuraAddBuff字段为Buff A，则会为AuraEffect所属单位添加Buff A，在此AuraEffect整个生命周期中，不论其抑制状态怎么切换，均不会再次添加BuffA，但是当该AuraEffect被移除，或被重叠层数时，可以再次添加A；

（即 若单位M处于N的光环中，N光环Aura配置有AuraAddBuff，则若M频繁进出N的光环，会被频繁加上该Buff）