

伤害计算流程：

设攻击者为A，受击者为B，A施放的技能为S，造成的伤害为Damage

1.判断无敌

首先判断B是否是无敌状态，如果是，则结束流程；

2.计算是否暴击

A的暴击率 - B的抗暴击率，取随机值看是否在此范围内，来决定是否产生暴击；（此过程只是计算是不是会产生暴击，不对B具体应用暴击效果）

3.计算霸体值伤害

伤害对B的霸体消耗值=

A的技能霸体伤害值属性（由S赋值）* A的破霸体伤害值系数

4. 计算触发被动

判定A的攻击行为，是否会触发A的被动技能；

判定B的受击行为，是否会触发B的被动技能；

* 若部分被动技能可以加深 | 减免 伤害，则判定此次供给是否满足条件，如满足，则修改攻击者的InsDamageAddPer属性，或修改受击者的InsDamageReducePer属性。

5.计算基础伤害

根据伤害类型（物理伤害与魔法伤害）的不同，确认需要攻守双方的物理攻击力还是魔法攻击力参与计算，确定后按下面计算基础伤害：

设 SrcAtk表示A的攻击力，TarDef表示B的防御力

如果SrcAtk > 3 * TarDef

则 $\text{BaseDamage} = \text{SrcAtk} - 3 * \text{TarDef}$;

否则

$\text{BaseDamage} = \text{SrcAtk} * \text{SrcAtk} / (4.5f * \text{TarDef})$;

$\text{Damage} = \text{Max} (\text{BaseDamage} , 1)$; //保底

6.计算技能基础伤害率

设S为攻击者A的技能基础伤害率属性

$\text{EBattleAttrId.SkillBaseDamageRating}$

则 $\text{Damage} * = (1 + S * 0.001)$;

7.计算技能基础伤害

设M为攻击者A的技能基础伤害属性

$\text{EBattleAttrId.SkillBaseDamage}$

则 $\text{Damage} + = M$;

8.计算蓝量转换的伤害加成

假如攻击技能是耗尽攻击者所有蓝量，则会有一个额外的攻击加成
攻击加成的计算方法：

$$\begin{aligned} \text{DmgPlus} = & 1 + (\text{CurPower} - \text{PowerConsumeList}[0]) \\ & / \text{powerConsumeList}[1] \\ & * (\text{powerConsumeList}[2] * 0.01) \\ &); \end{aligned}$$

则 $\text{Damage} * = (1 + \text{DmgPlus})$;

9.计算普通伤害加成

a. 获取攻击者A伤害加成率 $\text{SecDamageAddRating}$

与受击者B的伤害减免率SecDamageReduceRating的差值
NormalAddRate;

b.修正NormalAddRate =Max(1 + NormalAddRate * 0.0001, 0.2f);

c.应用普通伤害加成： Damag = Damag *(1+NormalAddRate);

10.计算暴击伤害加成

如果2中计算出此次攻击为暴击效果，则进行此步：

a. 获取攻击者暴击加成率SecCriticalDmgAddRating与受击者的暴击减免率SecCriticalDmgReduceRating的差值 NormalAddRate;

b.修正NormalAddRate =Max(1 + NormalAddRate * 0.0001, 0.2f);

c.应用普通伤害加成： Damag = Damag *(1+NormalAddRate);

11.随机值处理

Damage *= CRandom.Range(0.95f, 1.05f);

12.技能键位增伤

获取攻击者的技能键位增伤属性SkillBtnDamageRating值
BtnDmgRating

Damage *= (1+ BtnDmgRating) ;

此阶段结束，将Damage保底为1；

Damage =Max (Damage , 1) ; //保底

13计算技能引爆Buff伤害加成

若技能S的配置字段ExplodeBufs不为空，则表示该技能的施放可引爆B身上ExplodeBufs配置的Buff，并根据其叠加层数来进行伤害加

成；

ExplodeBufs的配置格式：Buff1,伤害加成千分比1；Buff2，伤害加成千分比2；Buff3，伤害加成千分比3；.....

计算方式：

1.检测B身上是否含有Buff1，Buff2，Buff3.....

若含有，则将对应的伤害加成参数，配合叠加层数，应用给Damage;

设B均含有以上三种Buff，则设MultiplyPara初始为1；

MultiplyPara += buff1的叠加层数 * 伤害加成1千分比；

MultiplyPara += buff2的叠加层数 * 伤害加成2千分比；

MultiplyPara += buff3的叠加层数 * 伤害加成3千分比；

则 $Damage = Damage * (1 + MultiplyPara)$;

14.计算攻击者的伤害加成

获取攻击者A的伤害加成属性InsDamageAddPer

应用：

$Damage * = (1 + InsDamageAddPer * 0.001)$

15.计算受击者的伤害减免

获取攻击者A的伤害加成属性InsDamageReducePer

应用：

$Damage * = (1 - InsDamageReducePer)$

16.应用受击者的护盾减伤属性

获取受击者B的护盾属性值SpeShield,如不为0，则消耗B的护盾值，减免对应伤害；

$Damage -= SpeShield$

17.应用受击者的吸收伤害属性

获取受击者B的吸收伤害属性值SpeAbsorbDamage,如不为0，则消耗B的吸收伤害属性值，减免对应伤害；

$Damage -= SpeAbsorbDamage$

18.应用格挡技能减伤

如果伤害来源是Buff，则不应用格挡减伤；（格挡技能挡不了Buff伤害）

如果B的格挡是只能格挡来自前方的伤害，且A不在B的前方，无法格挡，则不能应用格挡减伤；

1.获取B的格挡减伤百分比属性SpeParyDamgePercent，如不为0，则；

$Damage * = 1 - SpeParyDamgePercent * 0.01;$

2.获取B的格挡减伤固定值属性SpeParryDamgeValue，如不为0，则；

$Damage -= SpeParryDamgeValue;$

（一种格挡技能只会配置一种格挡类型，以上两个不同时为非0）

19.计算飞弹衰减

如果伤害来自飞弹，则应用飞弹衰减；

获取飞弹的配置字段flyBombDmgAttenuationData，根据飞弹已飞行距离占可飞行距离的比率，来对伤害进行衰减；

flyBombDmgAttenuationData配置格式：

距离比1，伤害比1；距离比2，伤害比2；距离比3，伤害比3；

（距离是百分比，伤害是千分比）

若飞弹飞行距离比率 在小于距离1，则应用伤害比1，小于距离2，但大于距离1，则应用伤害比2，依次类推（固定三段配置，且距离比3固定为100）

根据上述得出伤害比为FlyBombDmgPercent.

则 $\text{Damage} * = \text{FlyBombDmgPercent} ;$

20.应用精力伤害加成|衰减

获取攻击者的副本精力属性SecEnergyMax，若其大于0，则说明攻击者身处副本需要计算 由于精力不足而导致的伤害衰减，具体计算过程如下：

1.获取精力衰减的配置数据，其格式为 改变阈值1，伤害改变比1（百分比），改变类型2；改变阈值2，伤害改变比2（百分比），改变类型2.....

2.获取攻击者的当前精力值属性CurEnergyValue；

3.将此属性与配置值改变阈值X对比，如果小于该值，则应用改变伤害比Y，

如果改变类型为0，则是伤害衰减，

则 $\text{Damage} * = (100 - Y) * 0.01 ;$

如果改变类型为1，则是伤害加成

则 $\text{Damage} * = (100 + Y) * 0.01 ;$

21.吸收的伤害转治疗

在17中，若B有吸收伤害，则会将在此步将其转化为治疗，用来恢复自身血量；

设单位B受伤害之前的SpeAbsorbDamage属性值为500；

受伤害之后的SpeAbsorbDamage属性值为200；

则M吸收的伤害量 $\text{AbsorbDmg} = 300$;

则最终转为治疗的血量:

$\text{HealValue} = \text{AbsorbDmg} * \text{SpeAbsorbDamage2HealPercent} * 0.01 * (1 + \text{治疗增益系数} * 0.0001)$

$\text{CurHP} += \text{HealValue}$;

22.反弹伤害给攻击者

根据受击者B自身受到的伤害值，按

SpeReboundDamagePercent属性的比率反弹伤害，并将伤害应用到攻击者所属主人的身上；

23.攻击者吸血

攻击者A根据造成的伤害值，进行吸血，增加自身当前血量；

获取上面20得出的最终Damage

则M吸血量为:

$\text{Finalvalue} = \text{Damage} * \text{SpeHemophagiaPercent} * 0.01 * (1 + \text{治疗增益系数} * 0.0001)$

$\text{CurHp} += \text{Finalvalue}$;

24.重置部分属性

重置攻击者A的InsDamageAddPer属性为0；

重置受击者B 的InsDamageReducePer属性为0；

这两个属性的赋值是由被动技能在每次攻击时完成的，因此在每次攻击结束时，其都会被清零。

(每次攻击都需要判定此次攻击是否满足被动里的伤害加成或减免的条件，如果满足，就对上述两个属性赋值，故需要每次伤害结算之后将这两个属性清0)

