我们需要在ActorComponent中实现相应的数值，如最大生命值、当前生命值等，并提供对应的改值委托

以HP（代指CurrHP）为例子，进行下面的表述

出于封装性的考虑，我们需要将HP为private属性，并在public中提供Get和Set函数。并在改值时调用动态多播委托。

首先在头文件中

实现数值的定义

private:

    UPROPERTY(EditAnywhere)

    float MaxHP;

    UPROPERTY(EditAnywhere)

    float CurrentHP;

public:

    UFUNCTION(BlueprintPure)

    float GetCurrentHP(){return CurrentHP;}

    UFUNCTION(BlueprintCallable)

    float SetCurrentHP(float Value);

然后实现广播，在类外声明委托

DECLARE\_DYNAMIC\_MULTICAST\_DELEGATE\_TwoParams(FOnCombatValueChanged,float,OldValue,float,NewValue);

在类中，实现值变化委托

private:

    // 实际修改HP的函数

    void ModifyHealth(float Value);

public:

    UPROPERTY(BlueprintAssignable)

FOnCombatValueChanged OnCombatValueChanged;

public:

    UFUNCTION(BlueprintCallable)

    void ApplyDamage(float DamageAmount);

    UFUNCTION(BlueprintCallable)

    void ApplyHealing(float HealAmount);

然后实现对应的函数

首先是实现ModifyHealth函数，它是所有其他改值函数的基础

void UElementalCombatSystem::ModifyHealth(float Delta)

{

    if(FMath::IsNearlyZero(Delta))  return;

    float OldHealth = CurrentHP;

    float NewHealth = FMath::Clamp(CurrentHP+Delta,0,MaxHP);

    CurrentHP=NewHealth;

    float ActualDelta=NewHealth-OldHealth;

    OnCombatValueChanged.Broadcast(OldHealth,NewHealth,ActualDelta);

    // if (FMath::IsNearlyZero(CurrentHP))以执行死亡检测

}

然后是Apply Damage、ApplyHealing、SetHP函数

void UElementalCombatSystem::SetCurrentHP(float NewValue)

{

    float Delta=NewValue-CurrentHP;

    ModifyHealth(Delta);

}

void UElementalCombatSystem::ApplyDamage(float DamageAmount)

{

    ModifyHealth(-FMath::Abs(DamageAmount));

}

void UElementalCombatSystem::ApplyHealing(float HealAmount)

{

    ModifyHealth(FMath::Abs(HealAmount));

}

当我们需要创建新的属性时，

类比于HP，我们需要实现的有

private:// 数值

    UPROPERTY(VisibleAnywhere,Category="InitValues")

    float CurrentHP;

    // 实际修改HP的函数

    void ModifyHP(float Delta);

public: // 属性值的Set和Get

    UFUNCTION(BlueprintPure,BlueprintCallable)

    float GetCurrentHP(){return CurrentHP;}

    UFUNCTION(BlueprintCallable)

    void SetCurrentHP(float Value);

public:// 改值委托

    UPROPERTY(BlueprintAssignable)

    FOnCombatValueChanged OnHPChanged;

public:// 开放函数，用于实际数值调整

    UFUNCTION(BlueprintCallable)

    void ApplyDamage(float DamageAmount);

    UFUNCTION(BlueprintCallable)

    void ApplyHealing(float HealAmount);

本文这里省略了MaxHP、SP、MP、ATK、MAG的代码。相关请参考Github