

终于到主动技能的最后一块了——点目标技能
之前的 4 个主动技能教程的内容依然可以用在点目标技能

先附上之前 4 个教程的链接

先来点目标的基础设置

```
"MyAbility_point"
{
    "AbilityBehavior"
    "DOTA_ABILITY_BEHAVIOR_POINT|DOTA_ABILITY_BEHAVIOR_AOE"
    "BaseClass"                "ability_datadriven"
    "AbilityTextureName"        "juggernaut_omni_slash"
    "AbilityCastAnimation"      "ACT_DOTA_ATTACK_EVENT"

    "AOERadius"                "300"
    //-----
    "AbilityCastPoint"          "0.3"

    //-----
    "AbilityCooldown"           "1"

    //-----
    "AbilityManaCost"           "0"

    "OnSpellStart"
    {
        "Damage"
        {
            "Target"
            {
                "Types"          "DOTA_UNIT_TARGET_BASIC|DOTA_UNIT_TARGET_HERO"
                "Teams"           "DOTA_UNIT_TARGET_TEAM_ENEMY"
                "Flags"           "DOTA_UNIT_TARGET_FLAG_NONE"
                "Center"          "POINT"
                "Radius"          "%radius"
            }
            "Type"                "DAMAGE_TYPE_PURE"
            "Damage"              "100"
        }
    }
}

"AbilitySpecial"
{
    "01"
```

```

        {
            "var_type"      "FIELD_INTEGER"
            "radius"        "300"
        }
    }
}

```

英雄的设置

```

"npc_dota_hero_juggernaut"
{
    "override_hero"      "npc_dota_hero_juggernaut"
    "AttackRate"         "0.5"
    "Ability1"           "MyAbility_unit"
    "Ability2"           "MyAbility_null"
    "Ability3"           "MyAbility_point"
    "VisionNighttimeRange" "1800"
}

```

注意一下哦，这里有些地方不同

AbilityBehavior 里面多了 DOTA_ABILITY_BEHAVIOR_AOE

这代表这个技能是点目标的 AOE 技能

"AOERadius" "300"这句话设置了鼠标指示器的大小

有 DOTA_ABILITY_BEHAVIOR_AOE 加上 AOERadius 这个才能把鼠标变成一个蓝色的圈圈
两者缺一不可，可以删除其中一个看看效果

接下来加点效果

声音就不加了，就加个特效和击退效果

特效用 particles/base_destruction_fx/gensmoke.vpcf

```

"OnSpellStart"
{
    "Damage"
    {
        "Target"
        {
            "Types"      "DOTA_UNIT_TARGET_BASIC|DOTA_UNIT_TARGET_HERO"
            "Teams"      "DOTA_UNIT_TARGET_TEAM_ENEMY"
            "Flags"      "DOTA_UNIT_TARGET_FLAG_NONE"
            "Center"     "POINT"
            "Radius"     "%radius"
        }
        "Type"          "DAMAGE_TYPE_PURE"
        "Damage"        "100"
    }
}

```

```

    }

    "FireEffect"
    {
        "EffectName"      "particles/base_destruction_fx/gensmoke.vpcf"
        "EffectAttachType" "world_origin"
        "TargetPoint"      "POINT"
    }

    "Knockback"
    {
        "Target"
        {
            "Types"      "DOTA_UNIT_TARGET_BASIC|DOTA_UNIT_TARGET_HERO"
            "Teams"       "DOTA_UNIT_TARGET_TEAM_ENEMY"
            "Flags"       "DOTA_UNIT_TARGET_FLAG_NONE"
            "Center"      "POINT"
            "Radius"      "%radius"
        }
        "Center"      "POINT"
        "Duration"     "0.5"
        "Distance"     "300"
        "Height"       "300"
        "IsFixedDistance" "0"
        "ShouldStun"   "1"
    }
}

```

这里里面的部分知识都在前面的教程提到了，这里不再赘述

接下来是 Lua，还是弄伤害就太单调了

技能效果：快速移动到指定的地方，到达之后对那个地方的单位造成伤害

需要修改一下技能，先去掉上面 OnSpellStart 里面的内容

OnSpellStart 里面加入 RunScript，所以我们需要在之前 MyAbility.lua 里面加入 MyAbility_point 函数

```

"MyAbility_point"
{
    //-----
    "AbilityBehavior"      "DOTA_ABILITY_BEHAVIOR_POINT"
    "BaseClass"            "ability_datadriven"
    "AbilityTextureName"   "juggernaut_omni_slash"

    //-----
}

```

"AbilityCastPoint" "0.0"

//-----

"AbilityCooldown" "0.3"

//-----

"AbilityManaCost" "0"

"OnSpellStart"

{

 "RunScript"

 {

 "ScriptFile" "scripts/vscripts/MyAbility.lua"

 "Function" "MyAbility_point"

 "Target" "POINT"

 }

 "ApplyModifier"

 {

 "ModifierName" "modifier_MyAbility_point_act"

 "Target" "CASTER"

 }

 "ApplyModifier"

 {

 "ModifierName" "modifier_phased"

 "Target" "CASTER"

 }

}

"Modifiers"

{

 "modifier_MyAbility_point_act"

 {

 "Duration" "0.2"

 "OverrideAnimation" "ACT_DOTA_ATTACK_EVENT"

 }

}

"AbilitySpecial"

{

 "01"

 {

 "var_type" "FIELD_INTEGER"

```

        "radius"      "300"
    }
}
}

```

这里可以看出我把 AbilityCastAnimation ， DOTA_ABILITY_BEHAVIOR_AOE 和 AOERadius 去除了

技能 CD 改为 0.3 秒

OnSpellStart 里面有两个 ApplyModifier，第一个运行了 modifier_MyAbility_point_act 这个 modifier 里面有个 OverrideAnimation，就是修改动作的意思，并且这个动作持续时间 0.2 秒

主宰的剑刃风暴就可以用这个做，这个是修改技能施法前摇结束之后的动作

我删掉的 AbilityCastAnimation 这是用来修改施法前摇的动作

本质上有些区别的

第二个运行 modifier_phased，但是可以看到 Modifiers 里面却没有这个 modifier

其实这个 modifier_phased 是自带的，就是相位移动的 BUFF，给单位之后就不会有碰撞

因为我们做的是移动，移动过程中肯定不要有碰撞，没有这个 modifier，如果移动到那个地方，刚好有个单位在那里，两个有碰撞的单位靠得太近会卡住，两个都无法移动，这时候添加 modifier_phased 再删除就可以解决这个问题

删除这个 modifier 我是在 Lua 里面删除的

Lua 代码

```

function MyAbility_point( keys )
    local caster = keys.caster
    local point = keys.target_points[1]

    local radius = 400      --设置范围
    local teams = DOTA_UNIT_TARGET_TEAM_ENEMY
    local types = DOTA_UNIT_TARGET_BASIC+DOTA_UNIT_TARGET_HERO
    local flags = DOTA_UNIT_TARGET_FLAG_NONE

    --设置施法者的面向角度
    caster:SetForwardVector( ((point - caster:GetOrigin()):Normalized()) )

    GameRules:GetGameModeEntity():SetContextThink(DoUniqueString("MyAbility_point_time
"),
    function( )

        --判断单位是否死亡，是否存在，是否被击晕
        if caster:IsAlive() and IsValidEntity(caster) and not(caster:IsStunned()) then

            --不是死亡，存在这个单位，没被击晕，就运行这里面的内容

```

```

        local caster_vec = caster:GetOrigin()          --获取施法者的位置

        if (point - caster_vec):Length()>50 then --当单位移动到距离指定地点小于
50 时不在进行移动

            local caster_abs = caster:GetAbsOrigin()
            local face = (point - caster_abs):Normalized()
            local vec = face * 75.0
            caster:SetAbsOrigin(caster_abs + vec)

            return 0.01
        else
            --获取范围内的单位，效率不是很高，在计时器里面注意使用
            local targets = FindUnitsInRadius(caster:GetTeam(), caster:GetOrigin(),
nil, radius, teams, types, flags, FIND_CLOSEST, true)

            --利用 Lua 的循环迭代，循环遍历每一个单位组内的单位
            for i,unit in pairs(targets) do
                local damageTable = {victim=unit,      --受到伤害的单位
                    attacker=caster,      --造成伤害的单位
                    damage=100,
                    damage_type=DAMAGE_TYPE_PURE}
                ApplyDamage(damageTable)    --造成伤害
            end

            caster:RemoveModifierByName("modifier_phased") --删除 modifier
            print("MyAbility_point Over")
            return nil
        end

        else
            caster:RemoveModifierByName("modifier_phased") --删除 modifier
            print("Caster is death or stunned")
            return nil
        end
    end, 0)

    print("Run MyAbility_point Succeed")
end

```

里面有些内容在之前的教程讲过了
新内容是移动的那部分

先看到 `caster:SetForwardVector(((point - caster:GetOrigin()):Normalized()))`

(point - caster:GetOrigin()):Normalized())是计算 caster 的位置面向施法点的向量
可以用面向角度来理解，但是实际上是一个向量，是个 Vector
这是高等数学里面知识了

接下来通过(point - caster_vec):Length()可以获取两点间的距离，这里的 point 和 caster_vec
谁减谁都没关系，都可以求出两点间的距离

距离大于 50 就进行移动，这里说一下这个移动方法

我这个移动方法是参考下面网址里面的实例，参考之后加以利用完成

https://developer.valvesoftware.com/wiki/Dota_2_Workshop_Tools/Scripting/Data_Driven_Motion_Controller_Example

这种移动有点像黑洞吸引的感觉，把速率调高之后就可以做成很好的移动效果

如果编过魔兽地图，应该晓得魔兽的位移方法，但是在 dota2 里面并不是很好
不过，还是简单提一下，因为 dota2 的计算很方便

比如移动到施法者面前 500 距离处，这样我们就需要先获取面前 500 距离处的点

```
local vec=caster:GetOrigin() + 500*caster:GetForwardVector()  
caster:SetOrigin(GetGroundPosition(vec, caster))
```

就这么简单两句话，魔兽的话还要计算 Cos 和 Sin

GetGroundPosition 这句话是必须的，否则就会出现移动之后看不到单位的模型

上面的代码较多，不懂代码或者初学的可能看起来不太懂
不懂就问吧

还需要注意一点，这代码有时候很奇怪的，直接复制进去运行却得不到我说的这种效果
太奇葩了，群里有出现过这种情况

但是呢，有时候自己照着打也会运行错，不用担心，代码就是这样，不会出错就不叫代码了
就像我老师说的：计算机没有错，错的只有你自己

也就是说要么 V 社错了，要么你错了

所以写代码的时候要仔细看清楚，按照运行流程自己好好查一查

就如上面的 Lua 代码，其实我已经敲过几次那个代码了，但是在我做这个教程的时候自己再
敲了一遍，结果就是错了，经过十多分钟的修改才正确运行，毕竟之前做过了，所以可以很
快解决了，我第一次敲这个移动代码，弄了几个小时才成功，所以，不要起来，来路漫漫，
还有更过错误等着你呢

最后在模拟一下米拉娜的月神箭

不过这里就不做那么复杂了，命中之后直接击晕 5 秒吧

为了方便测试，我先注释掉运行函数那一块

修改之后如下

```
"OnSpellStart"  
{  
    // "RunScript"  
    //{
```

```

// "ScriptFile" "scripts/vscripts/MyAbility.lua"
// "Function" "MyAbility_point"
// "Target" "POINT"
//}

"LinearProjectile"
{
    "EffectName"
    "particles\\econ\\items\\mirana\\mirana_crescent_arrow\\mirana_spell_crescent_arrow.vpcf"
    "MoveSpeed" "1000" //发射速率
    "StartRadius" "0" //获取的开始范围
    "EndRadius" "50" //获取的终止范围
    "FixedDistance" "1000" //距离
    "StartPosition" "CASTER"
    "HasFrontalCone" "0" //不晓得是什么玩意
    "ProvidesVision" "1" //是否给予视野
    "VisionRadius" "500" //视野大小
    "TargetTeams" "DOTA_UNIT_TARGET_TEAM_ENEMY"
    "TargetTypes" "DOTA_UNIT_TARGET_HERO"
    "TargetFlags" "DOTA_UNIT_TARGET_FLAG_NONE"
}

"ApplyModifier"
{
    "ModifierName" "modifier_MyAbility_point_act"
    "Target" "CASTER"
}
}

"OnProjectileHitUnit"
{
    "Damage"
    {
        "Type" "DAMAGE_TYPE_PURE"
        "Damage" "110"
        "Target" "TARGET"
    }

    "Stun"
    {
        "Duration" "5"
        "Target" "TARGET"
    }
}
}

```


`LinearProjectile` 发射一个线性投射物，米拉娜的月神箭或者帕克的波
记得特效要预载入

OK，这次点目标的教程结束了，这也就代表主动技能的教程到此结束了
接下来的技能教程当然就是被动技能，被动技能就可以不用这样分那么细了