FighterSkill //一个玩家的技能信息

{

Skill\_level\_ //map玩家拥有的所有技能

Skill\_use\_data\_ //map玩家主技能

{

Final\_cfg\_ //主技能经过加成后的技能配置

Use\_cfg\_ //主技能的基础加成配置

Cd\_param\_ //冷却时间加成系数

Damage\_radio\_param\_ //伤害百分比加成系数

Damage\_value\_param\_ //伤害值加成系数

Change\_parmas\_ //map触发事件

Skill\_event\_ //vector 子技能技能事件

Buff\_change\_param\_//子技能buff扩展

}

Skill\_child\_data\_ //map 玩家子技能

Fight\_skill\_passitive\_

{

Map\_other\_skill\_ //玩家被动技能

Map\_added\_buff\_info //被动技能加的buff以及最大lv

}

Skill\_replace\_

{

Skill\_replaced\_ //主技能的替换技能信息

}

}

**进入场景服流程：**

1. FighterSkill::RefleshAllSkillCfg()玩家切换地图或者技能变化刷新FighterSkill
2. sPlayeruseSkillData::AddChildSkillParam()加载子技能给主技能附加的各种能力。整理所有自己能给主技能添加的cd、伤害系数、伤害值增益。触发事件（n个），子技能技能事件，子技能buff扩展
3. FightSkillPassitive::ResetAllOtherSkillBuff() 整理被动技能给玩家增加的(buffMap\_added\_buff\_info)，移除旧buff,添加新buff。

**玩家主动使用技能流程：**

1. UseSkill::ReqUseSkill()，确认使用技能坐标，玩家活着，技能冷却。
2. FightSkill::GetTrueUsedSkillID()技能可能被其他技能替换获取替换技能的id和lv
3. 如果是格挡技能走格挡技能相关逻辑，FightSkillSpecial::ReqUseSabreSkill()
4. 扣技能消耗
5. CharacterBuff::CanSpecialSkill()如果不是普通攻击，判断是否能释放技能（判断身上buff效果类型type\_effect\_mask\_）**static整理类型优化**
6. CharacterPlayer::IsSkillIntervalEnableBreak()是否在技能公共cd冷却中，**配置一个技能cd系数给网络延时一个公共cd的允许误差**。
7. 将玩家设置到技能使用的坐标(x,y)和面朝向。如果技能无法移动释放停止移动。
8. CharacterFighter::BeginSkill()，depth参数防止调用层数或者递归过深。广播通知周围人使用技能了。
9. UseSkill::BeginSkill() 得到真正的skillCD，加入冷却时间map\_skill\_cd\_。获取技能释放中心点。
10. CharacterBuffSpecial::BeforeUseSkill() 检查call\_skill\_begin\_buffs当中是否有使用技能之前待触发的buff。sBuffEventData::AddUseSkillCount()，CharacterBuff::CheckEventCanCall()查看这个buff是否能触发（间隔，次数，冷却时间等条件），可触发则重制计数条件，走buff事件流程CharaterBuff::ApplyBuffEvent()。
11. UseSkill::CallSkill()生成sSkillInfo的技能开始时间、技能唯一ID、如果有对这一击技能buff加成的属性。
12. UseSkill::CheckEventBegin()如果目标id不为0则获取目标坐标，循环判断技能事件触发（如果该事件没伤害、没有主技能和子技能buff信息则不需要判断），将该事件的信息复制给sRepeatedEvent对应需要的参数中（技能，位置，伤害，目标，buff，技能附加属性、事件等）。如果时间立即触发则直接调用CallEvent，延时触发则放入vec\_repeat\_event\_等待触发。
13. UseSkill::Process()处理vec\_repeat\_event\_，如果玩家死掉清空技能事件，如果到了触发时间则触发事件，触发后删除事件。
14. UseSkill::CallEvent()，

1: UseSkill::GetTargetAround()获得技能范围内所有对象，如果技能目标id存在，则UseSkillCollision::IsSkillTargetValid()算出两个目标坐标差，根据技能范围形状（圆形、矩形、弧形）判断目标是否有效范围内，如果两个目标距离太近也算在判断范围内。根据技能目标类型的不同把自己或者合法目标或者加入判断对象vector。UseSkillCollision::GetTargetAroundByAreaType()获取玩家视野内的单位，根据上一句当中的流程判断在技能范围内的目标加入multimap（距离为key值）。判断技能可打到的人数，如果在范围内的人数过多，则按照距离或者随机选够目标加入受击对象vector()。

2: 判断每个vector当中的对象如果目标在技能范围，但是无法被攻击（攻击方阵营、安全区判定、是否能攻击判定，被攻击方是否无敌buff、安全区判定、阵营），则添加免疫伤害消息。

3：如果是伤害性技能，

31: UseSkillSpecial::BeforeDamageTar()判断玩家是否会被斩杀，扣除所有剩余血量。32: 如果没被斩杀DamageFormat::CaculateDamageWithLightSword()如果是怪物打挂机玩家不算伤害，判断技能是否命中、幸运一击、神圣一击、格挡、反射、暴击、并且加上对应的damgeType标志（计算的时候附上技能附带的属性）。 33: DamageFormat::GetDamageA()如果触发幸运伤害则取攻击最大值否则随机范围内的攻击值，计算属性增益后的攻击值TotalAttack。如果触发神圣一击，则不计算对方防御值TotalDeffend。两者得到伤害值DamageA。 根据双方元素攻击和防御算出damageB,额外附加伤害得到damageC，再根据暴击、格挡、幸运一击伤害算出最终伤害totalDamage。

34: 根据双方的加成伤害和减免伤害、技能加伤和技能减伤算出skillDamage。

35: UseSkillSpecial::AfterDamageTar()计算各类属性对skillDamage的伤害影响，计算伤害是否大于hp且是否会触发濒死设定，触发后不造成伤害，hp设定到特定属性影响值，同时给目标添加免疫死亡后触发的buff。CharacterBuff::AddDelayBuff()添加buff流程。

36：UseSkillSpecial::AttackAddSelfBuff()被攻击后，根据攻击方属性给目标添加被攻击后的buffID。CharacterBuff::AddDelayBuff()添加buff流程

37: 如果目标格挡成功加入格挡vector，如果伤害反弹成功伤害为0，根据双方属性算出反弹伤害。

38：FighterBattle::OnBeDamage()如果攻击方是玩家并且在副本当中，则把伤害信息通知所有玩家。

39: CharacterBuffSpecial::OnHpLose()检查call\_hp\_lose\_buffs\_是否有hp损失待触发的buff。如果血量百分比符合触发条件。获取buff信息，CharacterBuff::CheckEventCanCall()查看这个buff是否能触发（间隔，次数，冷却时间等条件），走buff事件流程CharaterBuff::ApplyBuffEvent()。

310: FightBattle::OnChangeHp()改变目标血量，广播当前自己血量，重算被打之后会影响的属性。

311: CharacterFighter::OnBeKilled(),根据击杀者属性计算击杀回血，并且增加血量。

312: CharacterFighter::OnDead(),移除目标玩家身上的死亡后无法保存的buff，CharacterBuff::RemoveBuff(),停止目标玩家移动，通知死亡信息。

313: UseSkillSpecial::CheckDamageResultBuff()，伤害结算后可能会触发的buff（Dizzy、LuckAttack、Freezy、Weak），根据双方特殊buff属性计算结果。CharacterBuff::AddDelayBuff()添加buff流程。

314: UseSkillSpecial::CheckDamageRsultAttr(),伤害结算后会触发的额外属性效果（目标闪避回血、闪避增加buff CharacterBuff::AddDelayBuff()添加buff流程，攻击方攻击冰冻目标回血，攻击回血）。

315: CharacterBuffSpecial::OnBeSkillAttacked()检查call\_be\_attacked\_buffs\_是否有被攻击后待触发的buff。如果血量百分比符合触发条件。获取buff信息，CharacterBuff::CheckEventCanCall()查看这个buff是否能触发（间隔，次数，冷却时间等条件），走buff事件流程CharaterBuff::ApplyBuffEvent()。

316: 如果反弹伤害不为0判断是否免疫伤害，如果不免疫伤害，走FighterBattle::OnBeDamage()流程（38），并且把自己也加入伤害vec当中。

317: 将本次对目标的伤害数值，伤害类型，反弹伤害放入伤害消息当中，伤害不为0将目标放入伤害vec当中。

318: UseSkill::AddHitTargetBuff()给目标玩家添加技能事件附带的buff效果，包括main\_buff\_info\_和child\_buff\_info\_当中的buff CharacterBuff::AddDelayBuff()添加buff流程。

319: 如果目标死亡则将目标加入死亡vec，如果自己死亡则将自己加入死亡vec。

320: UseSkill::CheckHitNumAddSelfBuff() 根据本次技能击中的目标数量，给玩家自己添加技能事件当中vec\_hit\_buff\_ 的buff CharacterBuff::AddDelayBuff()添加buff流程。

。

321：给伤害消息加上一个唯一key值标明是同一次伤害。通知给视野内玩家。将伤害vec当中每个玩家受到的伤害广播给视野内玩家。将死亡vec当中每个玩家的死亡消息广播给视野内玩家。

1. UseSkillSpecial::AfterCallSkill()清空本次技能计算临时修改的包含的信息（属性）。
2. CharacterBuffSpecial::RemoveReliveBuff()如果有重生buff需要移除。CharacterPlayer::PetAttack() 如果有宠物触发宠物攻击。RideInfo::ResetRideTiming()重置自动上马时间，打断采集动作。
3. 如果技能允许位移，则向前端发来的坐标移动。检查终点坐标是否可移动。SkillMove::SkillMoveBegin()根据技能配置的位移距离，位移速度，如果实际要位移的距离超过技能配置的距离则不允许位移。重置技能位移参数，将本次的位移参数赋值。SkillMove::SkillMoveProcess()，先判断技能是否开始或者已经结束移动，根据时间差算出本次移动到的坐标，如果可以走将玩家设置停到移动点位置，流程被打断需要重制所有技能位移参数。

**格挡技能流程：**

FightSkillSpecial::ReqUseSabreSkill()确认能否使用格挡技能（原力值是否够），设置格挡开始时间，格挡时长，扣除原力值。如果技能无法移动释放则停止移动。CharacterFighter::BeginSkill()（走8流程）。

**buff流程：**

1. CharacterBuff::AddDelayBuff()如果调用层数过深直接返回，如果buff需要延时则将参数赋值给sDelayBuffInfo加入vec\_delay\_buffs\_等待调用，不需要延时直接调用CharacterBuff:: AddBuff()。
2. CharacterBuff:: AddBuff()，

21: CharacterBuff::CheckCanAddBuffByRelace()检查是否有可替换buff等级高替换等级低的相同buff，有的话CharacterBuff::RemoveBuff()流程。 22: CharacterBuff::CheckCanAddBuffByAddTime()检查是否身上有相同buff可以延长时间，有的话增加时间，并且通知。

23: Characterbuff::CheckOverlay()检查身上有相同buff可以叠加，如果有删除旧的buff CharacterBuff::RemoveBuff()流程，没有的话叠加层数加一直到上限。

24: 如果叠加层数<=1 且不会强制覆盖buff特效，CharacterBuff::IsInRejectEffectType()则判断当前已有buff是否已经排斥这个buff的buff效果，如果有该效果则不添加buff。

25: CharacterBuff::AddRejectEffectType(),添加该buff会排斥的buff效果。

26: CharacterBuff::RemoveBuffByRejectLst()。检查身上的buff，如果buff的特效类型会被该buff排斥则删除buff CharacterBuff::RemoveBuff()流程。

27: 生成一个guid, 把buff各种信息填入sBuffData（配置，技能触发buff时附带的属性，buff状态）加入map\_buffs\_中。

28: CharacterBuff::AddbuffEffectType()添加这个buff加的效果，如果是第一次添加这个效果则检查是否需要处理CharacterFight::FirstAddBuffEffectType()如果特效可以使玩家停止移动需要停止等。

29: CharacterBuffSpecial::CheckAddSpecialBuff()。特殊buff处理，重生buff移除旧的重生buff，记录新的重生buff唯一id。格挡技能buff移除旧的格挡buff，记录新的格挡buff唯一id。 CharacterBuff::RemoveBuff()流程。是否是根据周围怪物数量触发的buff，是的话记录buff唯一id。是使用技能触发的buff，将buff唯一id加入call\_skill\_begin\_buffs\_。是被攻击后触发的buff，将buff唯一id加入call\_be\_attacked\_buffs\_。是hp丢失之后触发的buff，将buff唯一id加入call\_hp\_lose\_buffs\_。

210: 将新增buff信息发送给前端，只通知自己的发给自己，可以广播的则广播消息。

211: CharacterBuff::ApplybuffDataWhenBegin(),修改buff事件sBuffEventData当中的开始时间，最后一次调用时间等信息。如果buff事件是掉血触发的类型。则检查是否本次能触发，(玩家主动使用技能流程当中的39)。如果buff时间是计时器类型CharacterBuff::CheckEventCanCall()，检查buff时间已经结束，是否还能触发（在cd中，还没到触发时间，已经触发到最大次数等条件）。如果条件满足走buff事件流程CharaterBuff::ApplyBuffEvent()。

1. CharacterBuff::Process()计时器检查buff相关的判断。

31: CharacterBuffSpecial::Process()特殊类buff计时器检查，判断是否周围怪物数量触发的buff类型buff，判断是否到了触发时间，CharacterBuffSpecial::MonsterAroundCheck()，确认buff时间是否可以触发，获取配置范围内怪物数量，如果数量大于需要数量，确认buff触发概率是否成功，如果成功触发则添加buff CharacterBuff::AddDelayBuff()添加buff流程。

32: vec\_delay\_buffs\_当中有可触发buff则添加buff CharacterBuff::AddDelayBuff()添加buff流程。

33: 如果map\_buffs\_当中有待删除buff则删除。剩余不需要删除的buff，检查如果是时间触发类型的buff检查是否可触发，如果条件满足走buff事件流程CharaterBuff::ApplyBuffEvent()。

34: CharacterBuff::CheckEventCanEnd()检查buff事件是否可以结束，buff事件是否已经到了结束时间，调用次数到上限。需要结束buff事件CharacterBuff::RemoveBuffEvent()，扣除给玩家带来的属性，清空属性变化，将事件结束标识为true。如果buff事件结束，则移除buff CharacterBuff::RemoveBuff()流程。

**Buff事件：**

1. CharaterBuff::ApplyBuffEvent()。改变buff事件最后调用时间，记录调用次数，按照概率随机得到是否触发成功。

11. 如果事件类型是改变属性，则给玩家增加对应配置\*叠加次数的属性(CharacterBuff::ChangeOwnerAttr)。

12. 为技能结算添加额外的特定属性（UseSkillSpecial::AddBuffExtraattr()）。

13. 为移除buff类型，移除特定buff效果(buff流程26)。

14. 给其他目标添加配置当中的buff（CharacterBuff::Addbuff流程）。

15. 给自己添加配置当中的buff（CharacterBuff::Addbuff流程）。

16. 触发释放技能（CharacterFighter::BeginSkill()**玩家主动使用技能**走8流程）

17. 触发对周围人造成伤害 CharacterBuff::CallDamageAround()，

CharacterFighter::GetCanAttackTargets()获取视野内玩家信息，遍历set中的玩家是否能被攻击，如果可以攻击则用距离作为key值得到可攻击的目标。调用DamageFormat::CaculateDamageWithLightSword()计算伤害（**玩家主动使用技能流程32步骤**）。如果玩家无法受到伤害则伤害为0，可以受到伤害处理伤害FighterBattle::OnBeDamage()（**玩家主动使用技能流程38步骤**）。如果目标死亡将目标加入死亡vec，目标受到伤害将目标加入伤害vec，并将伤害信息加入vec。将伤害信息发送给前端显示S2C\_NtfBuffAroundDamage。最后将死亡、伤害目标发送。

移除buff流程

CharacterBuff::RemoveBuff()。CharacterBuff::RemoveBuffByData(),移除buff事件效果(CharacterBuff::RemoveBuffEvent())、移除buff影响类型、移除buff拒绝类型，通知buff移除。CharacterBuffSpecial::OnBuffGuidRemoved(),根据buffguid移除特定的buff标识(重生、格挡buff、怪物周边buff、技能使用触发vec、被攻击触发vec、hp损失触发vec)。将wait\_delete \_标识为true。