



角色系统

重点部分：

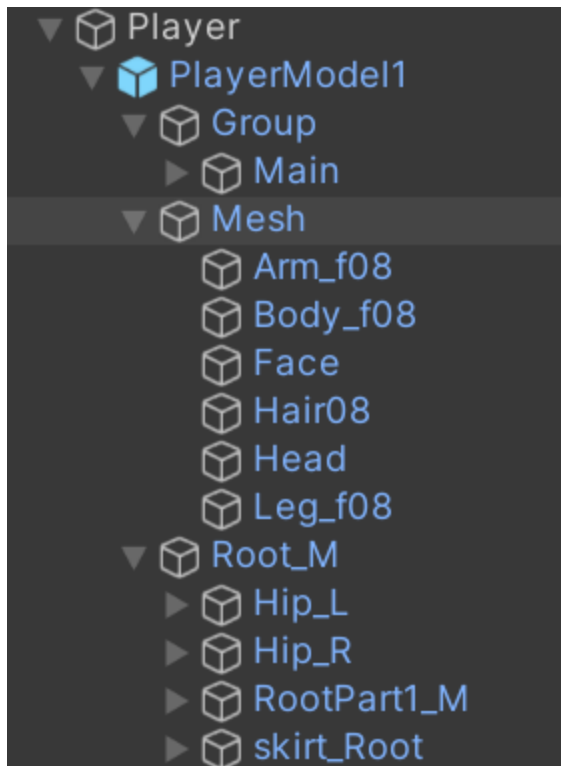
1. 玩家状态逻辑：玩家移动、动画切换、模型更改
2. 玩家血量/饱食度计算及同步UI显示
3. 玩家数据和模型的保存与恢复

核心脚本：

1. Player_Controller.cs：对外提供玩家模型、数据及一些方法的访问，起到一个承接转发的作用
2. Player_Model.cs
3. PlayerStateBase.cs：玩家状态基类, 抽象出所有玩家状态所需要的共同字段/函数
 - a. Player_Attack.cs
 - b. Player_Dead.cs
 - c. Player_Hurt.cs
 - d. Player_Idle.cs
 - e. Player_Mode.cs
4. PlayerData.cs
5. PlayerConfig.cs

玩家模型 PlayerModel.cs

玩家模型文件分析：



1. Group：不清楚用处
2. Mesh：玩家身体部位Skinned Mesh Renderer，与普通Mesh Renderer还是有区别的
3. Root_M：玩家身体部位，例如武器挂载位置就是在右手处（Wrist_R）

玩家模型脚本初始化

1. 初始化武器挂载位置：玩家模型右手处（Wrist_R）
2. 初始化模型在动画执行过程中触发的一些事件（委托/Action）例如移动时发出脚步声、攻击开始、停止攻击、攻击结束、玩家受伤、玩家死亡，执行方法方式为

`Action?.Invoke()`

玩家状态 PlayerStateBase.cs 及其子类

初始化方法：

1. 传入Player_Controller，目的是能在状态处理的时候快速找到玩家的一些数据和调用其他类的方法

玩家待机状态 PlayerIdle：

1. 进入状态：循环播放待机动画
2. 状态中：持续检测玩家是否按下水平/垂直方向按键
3. 退出状态：如果按下按键则切换到<玩家移动状态>状态退出待机

玩家移动状态 PlayerMove：

1. 初始化：需要传入 `CharacterController`，后续移动角色位置时需要用到 `CharacterController.Move`
2. 进入状态：播放移动动画
3. 状态中：
 - a. 检查玩家是否按下水平/垂直方向按键，如果没有持续按下，则切换到<玩家待机状态>
 - b. 通过Input得到水平/垂直方向按键值计算玩家朝向并使用插值计算提升转向流畅度

```
// 2. 查看是否进行朝向计算
Vector3 inputDir = new Vector3(h, 0, v);
Quaternion targetQua = Quaternion.LookRotation(inputDir);
player.playerTransform.localRotation = Quaternion.Lerp(
    player.playerTransform.localRotation,
    targetQua,
    Time.deltaTime * player.rotateSpeed
);
```

- c. 检查移动是否超过边界
 - d. 调用 `CharacterController.Move` 移动角色
4. 退出状态：无

玩家攻击状态 PlayerAttack：

1. 进入状态：确定攻击方向；播放攻击动画

2. 状态中：持续更新攻击方向，使用 `Quaternion.Slerp` 随时间平滑进行旋转
3. 退出状态：调用Player_Controller中的OnStopHit方法<停止攻击>

玩家受伤状态 PlayerHurt：

1. 进入状态：播放受伤动画
2. “状态中”和“退出状态”：无

玩家死亡状态 PlayerDead

1. 进入状态：关闭玩家身上的Collider防止播放死亡动画时被攻击进入受伤状态；播放死亡动画

玩家控制器 Player_Controller.cs

初始化方法

1. 读取角色配置和存档数据
2. 初始化角色模型相关的事件（实质是一些Action），详情见 (Player_Model.cs)<玩家模型脚本初始化>
3. 初始化玩家状态机并将玩家状态设置为待机状态
4. 初始化角色位置相关数据
5. 通过事件管理器触发生命值/饱食度UI更新

角色数据 Data/PlayerData.cs

```
[Serializable]
public class PlayerTransformData
{
    // 坐标: sv_position存档用, position外部调用用
    private Serialization_Vector3 sv_position;
    public Vector3 position {
        get => sv_position.ConverToVector3();
    }
}
```

```

        set => sv_position = value.ConvertToSVector3();
    }

    // 旋转
    private Serialization_Vector3 sv_rotation;
    public Vector3 rotation {
        get => sv_rotation.ConvertToVector3();
        set => sv_rotation = value.ConvertToSVector3();
    }
}

// 玩家主要数据：生命值、饱食度
[Serializable]
public class PlayerMainData
{
    public float hp;
    public float hungry;
}

```

角色配置 Config/PlayerConfig.cs

```

public class PlayerConfig : ConfigBase
{
    #region 角色属性配置
    [FoldoutGroup("角色配置"), LabelText("玩家转向速度")]
    public float rotateSpeed = 10; // 玩家转向速度
    [FoldoutGroup("角色配置"), LabelText("玩家移动速度")]
    public float moveSpeed = 4; // 玩家移动速度
    [FoldoutGroup("角色配置"), LabelText("玩家最大生命值")]
    public float maxHP = 100;
    [FoldoutGroup("角色配置"), LabelText("玩家最大饱食度")]
    public float maxHungry = 100;
    [FoldoutGroup("角色配置"), LabelText("玩家饱食度衰减速度")]
    public float hungryReduceSpeed = 0.2f; // 初设为5s掉1滴血
    [FoldoutGroup("角色配置"), LabelText("饱食度为0时玩家生命值衰减速度")]
    public float hpReduceSpeedOnHungryIsZero = 2.0f;
    [FoldoutGroup("角色配置"), LabelText("默认动画状态机")]
    public RuntimeAnimatorController normalAnimatorController;
    #endregion

    #region 角色音效资源
    [FoldoutGroup("角色音效资源"), LabelText("脚本音效")]
    public AudioClip[] footstepAudioClips; // 脚步音效
    [FoldoutGroup("角色音效资源"), LabelText("脚步音量")]
    public float footstepVolume = 0.5f; // 脚步音效音量
    #endregion

    #region 杂项
    [FoldoutGroup("杂项"), LabelText("音效配置")]

```


```

    public Dictionary<AudioType, AudioClip> AudioClipDict; // 音效配置字典
    #endregion
}

```




Script PlayerConfig

▼ 角色配置

| | |
|-----------------|--|
| 玩家转向速度 | 10 |
| 玩家移动速度 | 3 |
| 玩家最大生命值 | 100 |
| 玩家最大饱食度 | 100 |
| 玩家饱食度衰减速度 | 1 |
| 饱食度为0时玩家生命值衰减速度 | 0.5 |
| 默认动画状态机 | Player_Normal   |

▼ 角色音效资源















▼ 脚本音效 2 items 

| | |
|---|---|
|  footsteps grass 1 |    |
|  footsteps grass 2 |    |

脚步音量 0.2

▼ 杂项

▼ 音效配置 7 items 

| Key | Value |
|-----------|---|
| 玩家无法使用 ▼ |  失败    |
| 装备武器 ▼ |  物品装备    |
| 卸载武器 ▼ |  物品进背包    |
| 消耗品成功使用 ▼ |  使用消耗品    |
| 消耗品使用失败 ▼ |  失败    |
| 背包 ▼ |  物品进背包    |
| 通用失败音效 ▼ |  失败    |

