



# 虚幻引擎GamePlay框架理解与应用

虚幻社区经理-大钊

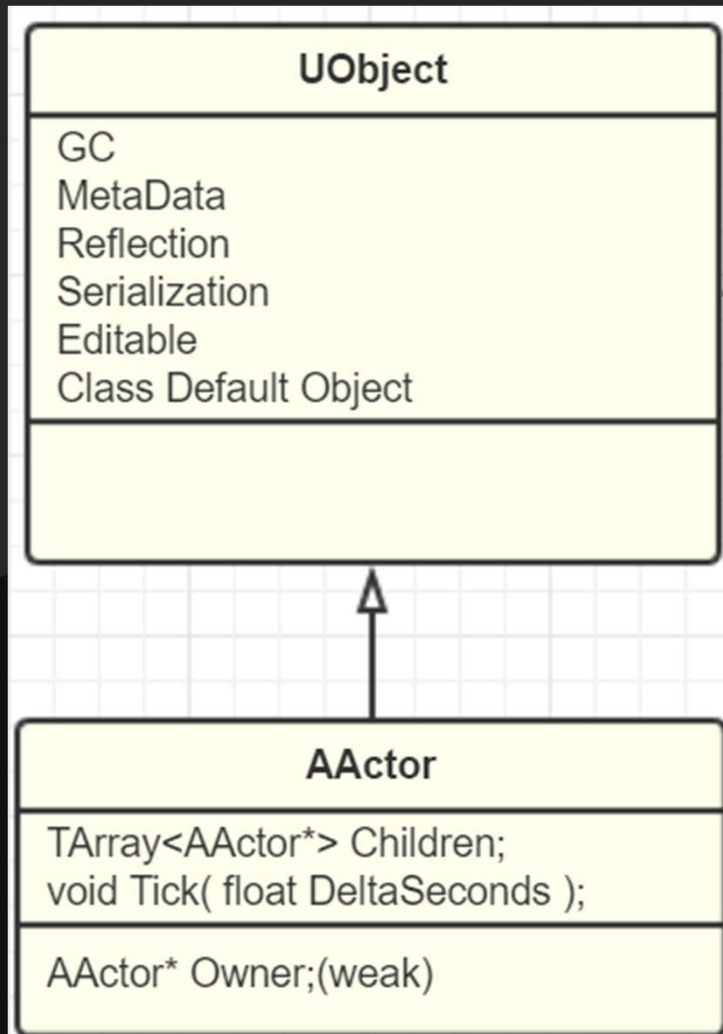
 #UE4 | @UNREALENGINE

## 什么逻辑写在哪里

- 各自的职责
- 从属交互关系
- 如何表现一个系统的多方面
- 哪一些写法是不太好的
- 常见的一些问题和误区

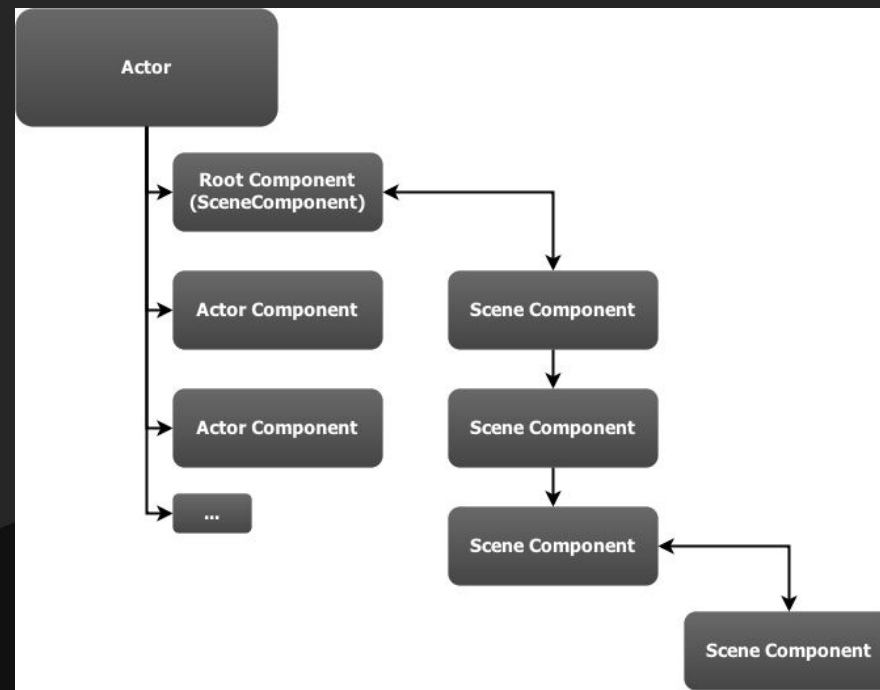
## Actor-演员/个体

- 组成世界的单位个体
- 可互相嵌套
- 组件的容器
- Spawn & Tick
- 网络通信的通道
- 派生出各种不同职责的Actor

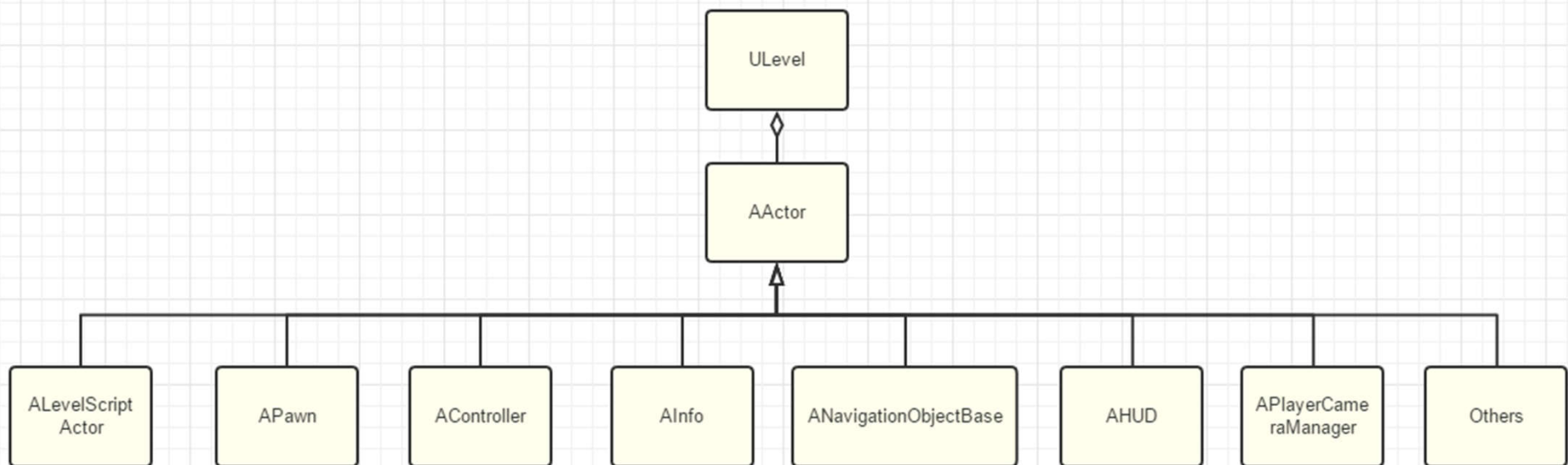


## Component-组件

- 组件是种能力
- 为Actor提供能力
- 能力一般跟业务逻辑无关
- 能力是跨游戏的



## 不同的Actor们



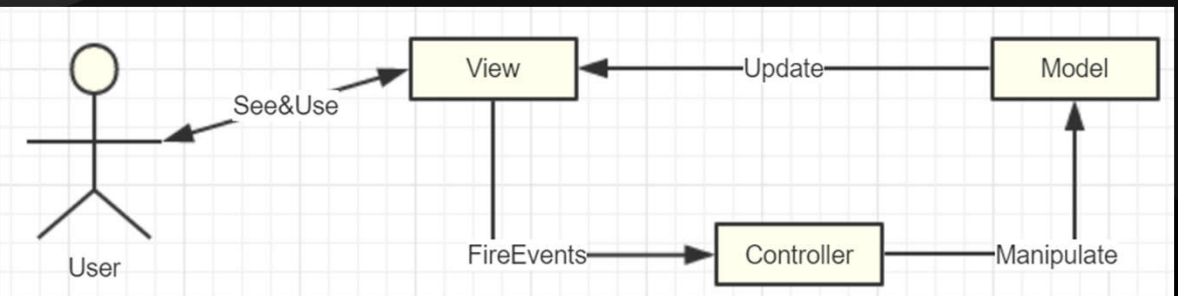
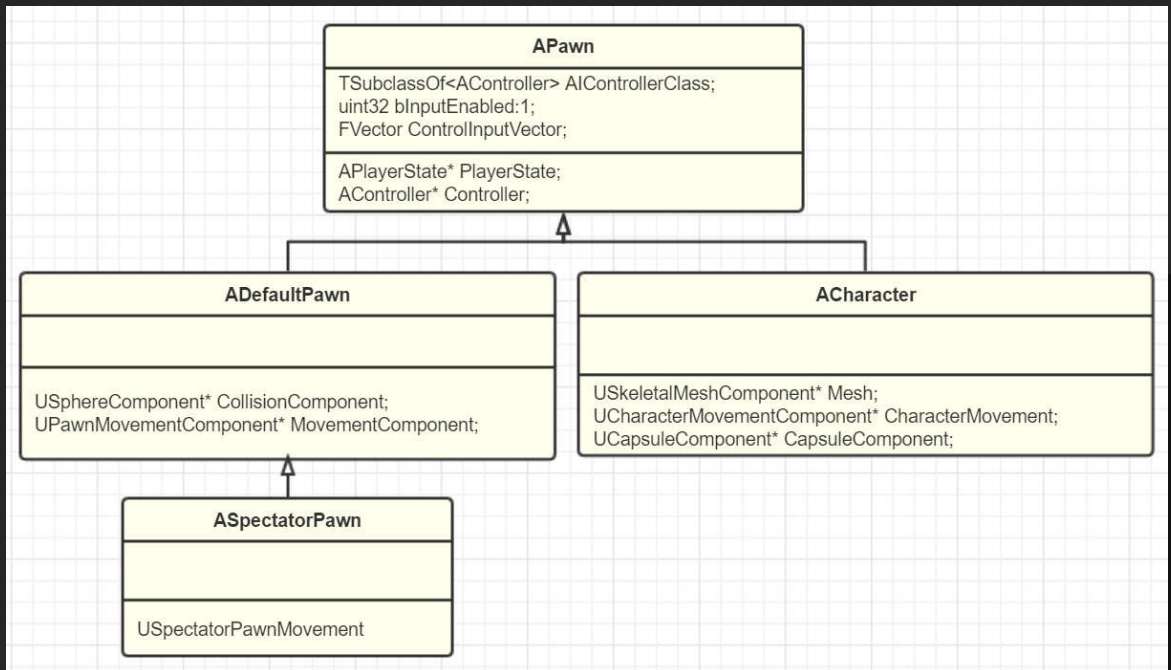
- 不是所有Actor都看得见

## ALevelScriptActor-关卡蓝图

- 一个特殊的默认脚本Actor: ALevelScriptActor
- 正经项目-大部分情况下都不适合
- 适合Demo, 新手村
- 适合一些关卡本身的触发机械互动

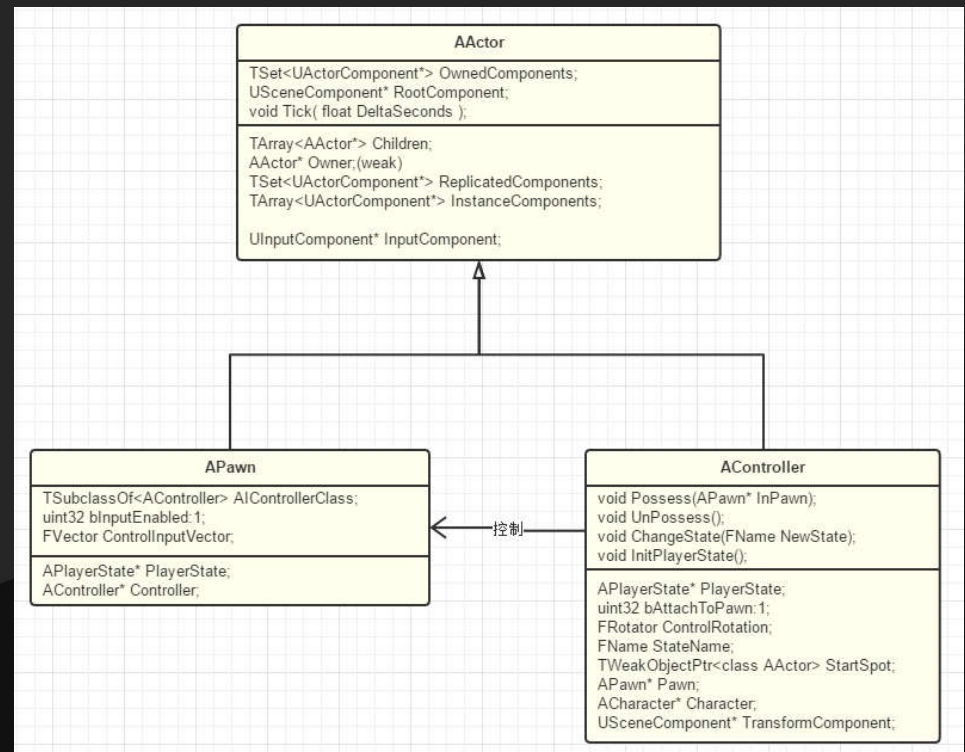
# APawn

- 可被控制
- 逻辑足够复杂MVC
- 肉体
- 人形角色



# AController & APlayerState

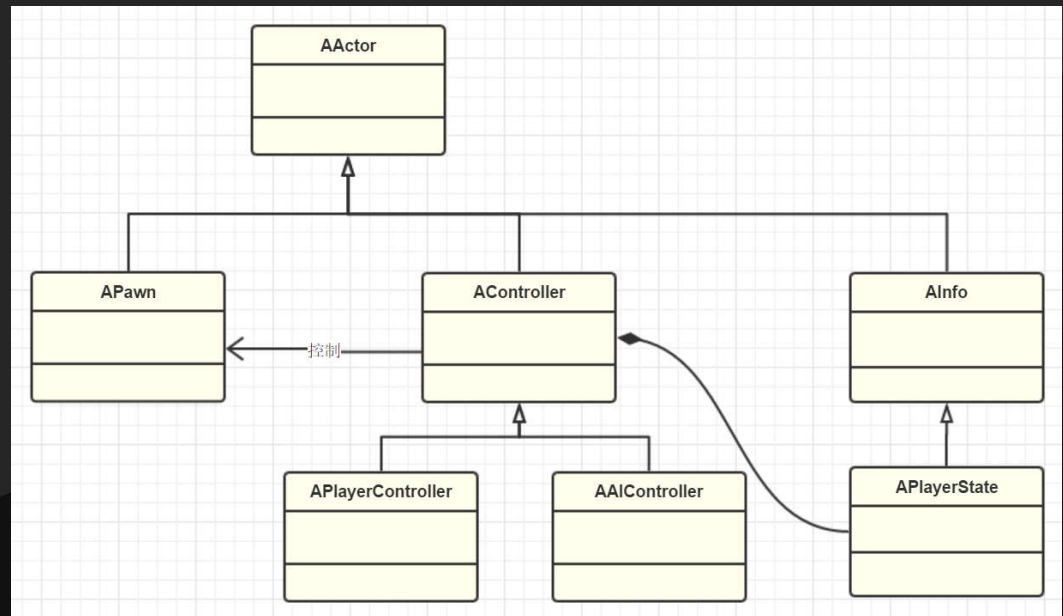
- 用来控制别的Actor的Actor
- 灵魂
- 肉体状态
- 抽离开是为了联机复制





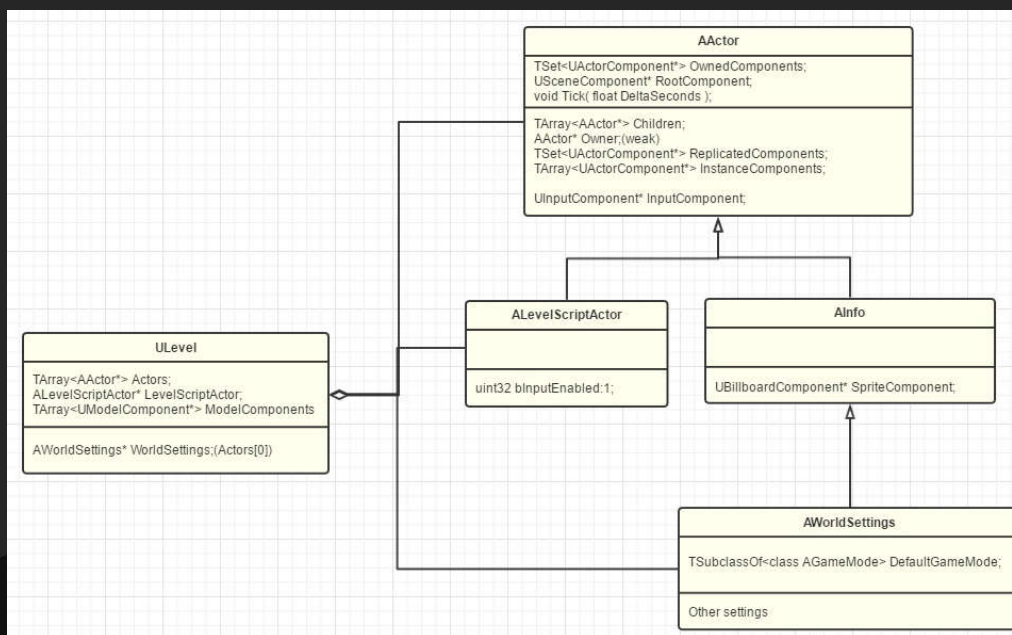
# APlayerController & AAIController

- 主角灵魂
- 和玩家沟通的渠道
- AI灵魂
- 行为树功能强大
- 黑板共享数据
- AI感知
- EQS环境查询



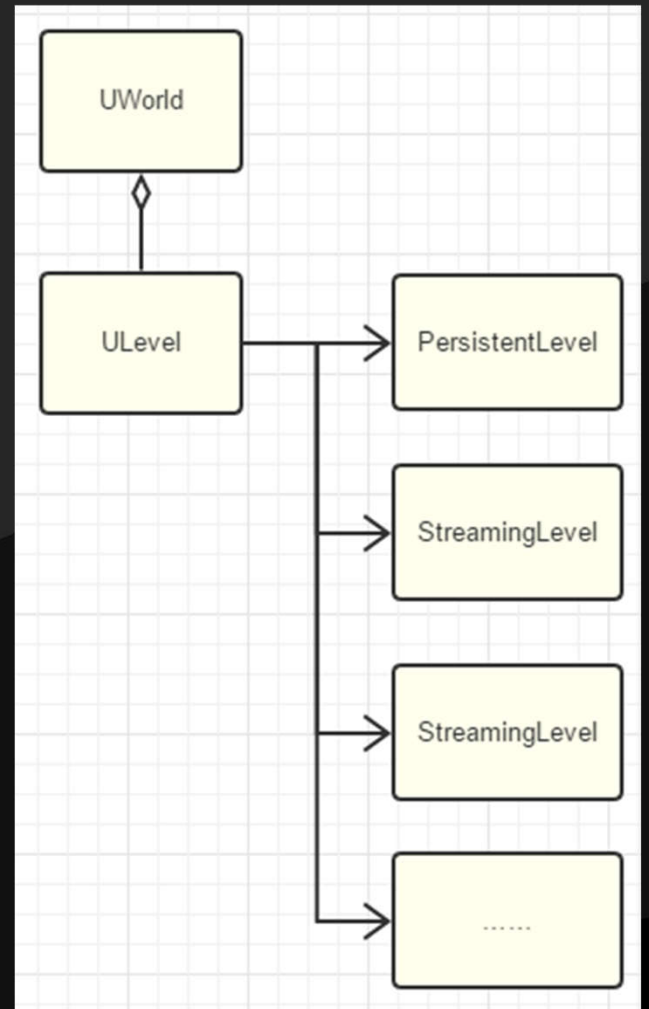
# ULevel 关卡

- Actor的集合
- World里的板块
- 对应umap



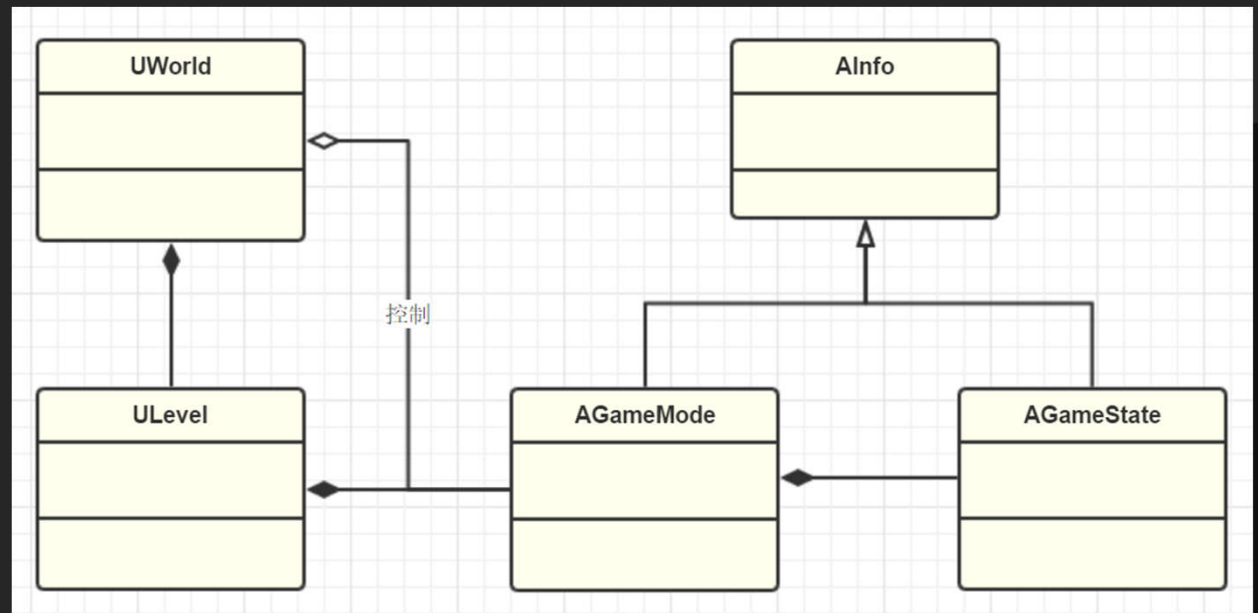
# UWorld

- Level的集合
- 运行时的真正的Actor集合
- “关卡”的真正单位
- 场景的真正表示对象



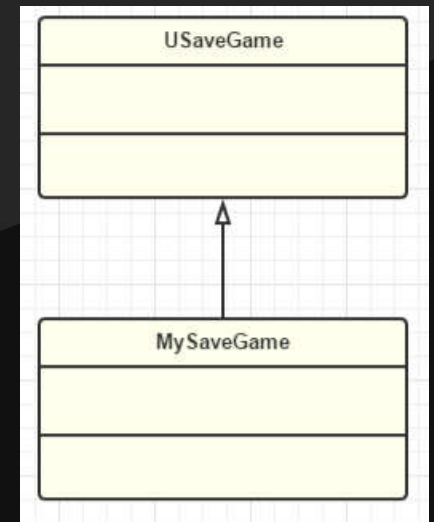
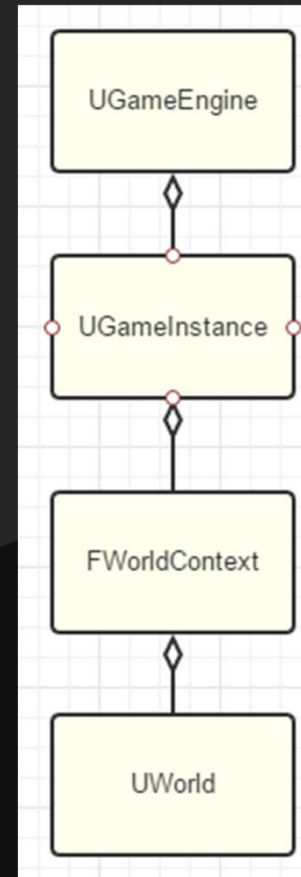
# AGameMode & AGameState

- 世界规则
- 最权威
- 世界状态
- 通知到各端



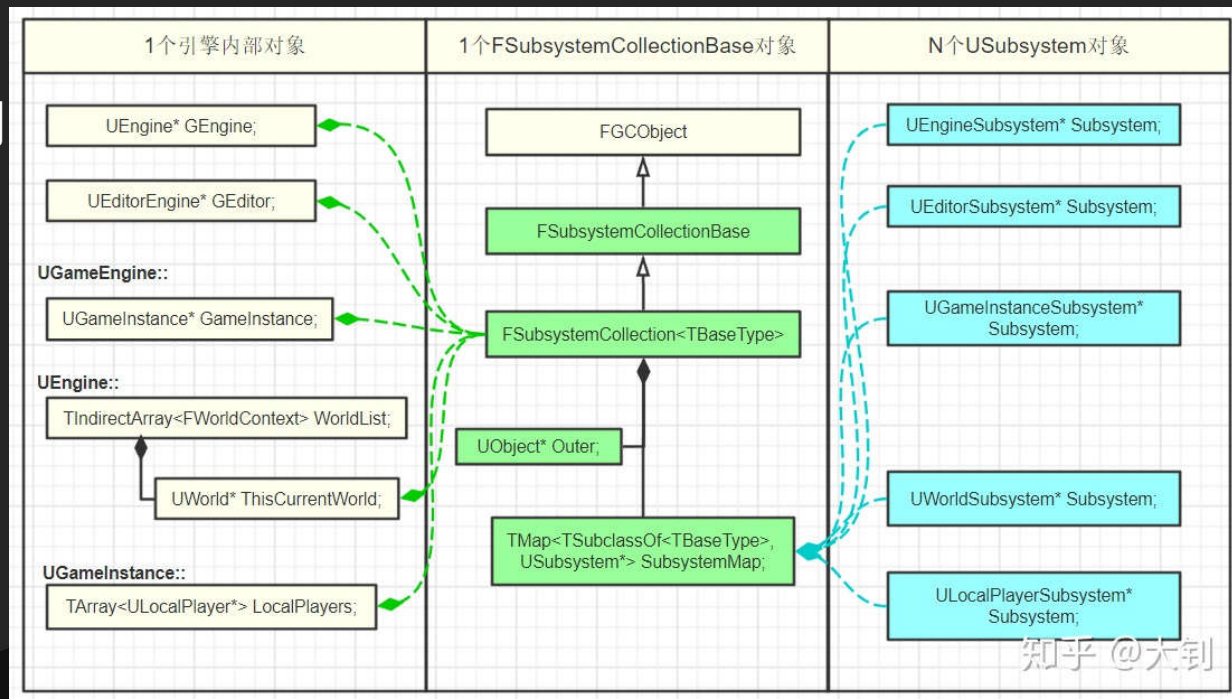
## UGameInstance - USaveGame

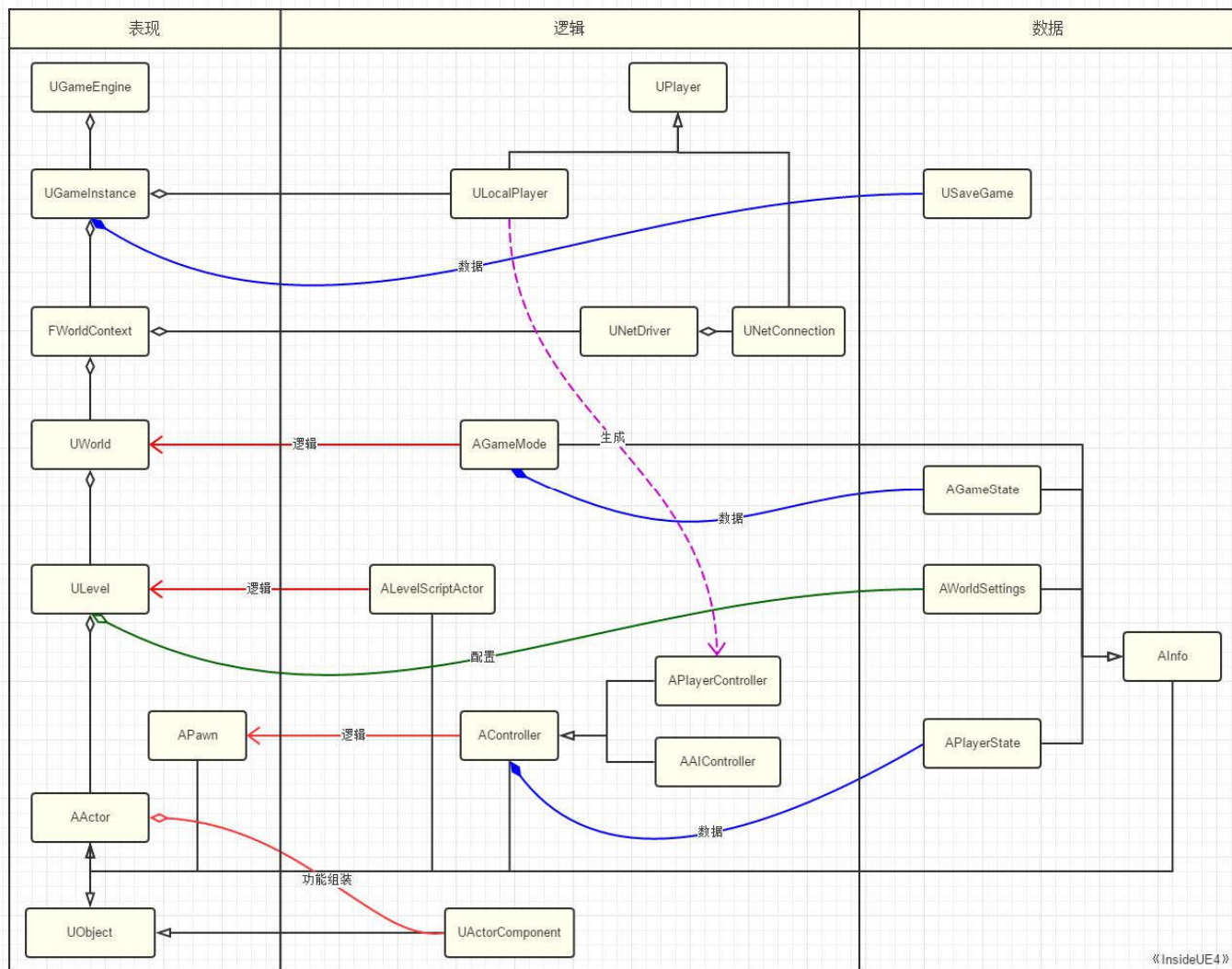
- 整场游戏的全局实例
- World的外围管理对象
- 跨关卡的逻辑
- 游戏存档



# USubsystem

- 业务逻辑的组件
- 5个生命周期粒度控制
- 自动生成对象
- 托管生命周期
- 替代Manager类





## 什么逻辑在哪里

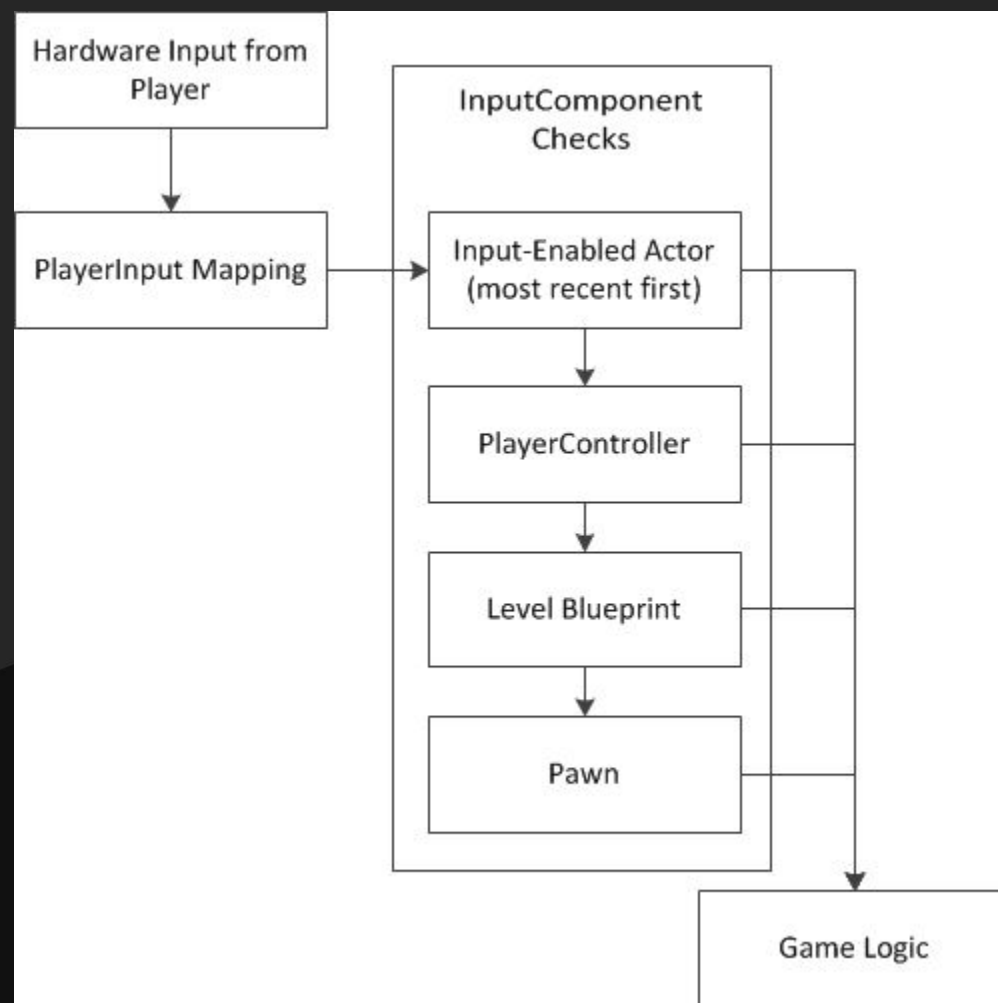
- Component: 功能、能力
- Actor: 个体本身的逻辑
- APawn-APlayerController-APlayerState 主角肉体-灵魂-状态
- APawn-AAIController-APlayerState AI肉体-灵魂-状态
- AAIController-BehaviorTree-Blackboard AI灵魂-行为树-数据
- AGameMode-AGameState 游戏规则-游戏状态
- UGameInstance-USaveGame 全局游戏实例-游戏存档
- USubsystem 游戏逻辑组件



# 问题和误区

## 输入事件在哪里处理？

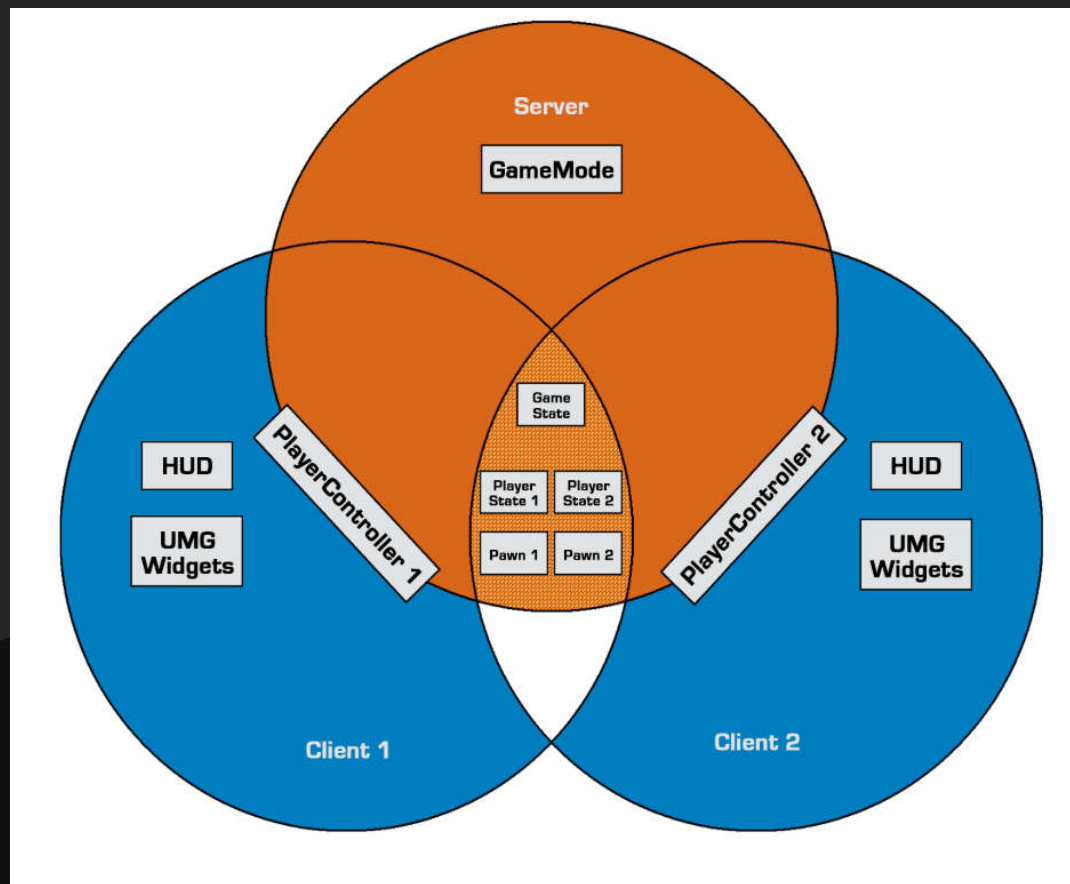
- 考虑联机扩展
- APlayerController 最优先
- APawn 其次
- AActor EnableInput
- 关卡蓝图随便测测 建钊1





## 联机的角色职责

- 谁在哪里
- 谁复制了



## 自带的不好？

- 什么都没有就好了吗？
- 多大程度上阻碍你？

## GamePlay框架里带Damage好不好?

- 统一的伤害接口
- 破碎, 游戏逻辑, GAS

# Q&A



微信公众号



UE需求反馈

知乎：虚幻引擎

B站：虚幻引擎官方

微信：unrealengine

QQ群2：630184107

邮箱：egc-community@epicgames.com

官网：unrealengine.com

社区经理-大钊 QQ：2722937652 微信：jack1046

邮箱：jack.fu@epicgames.com