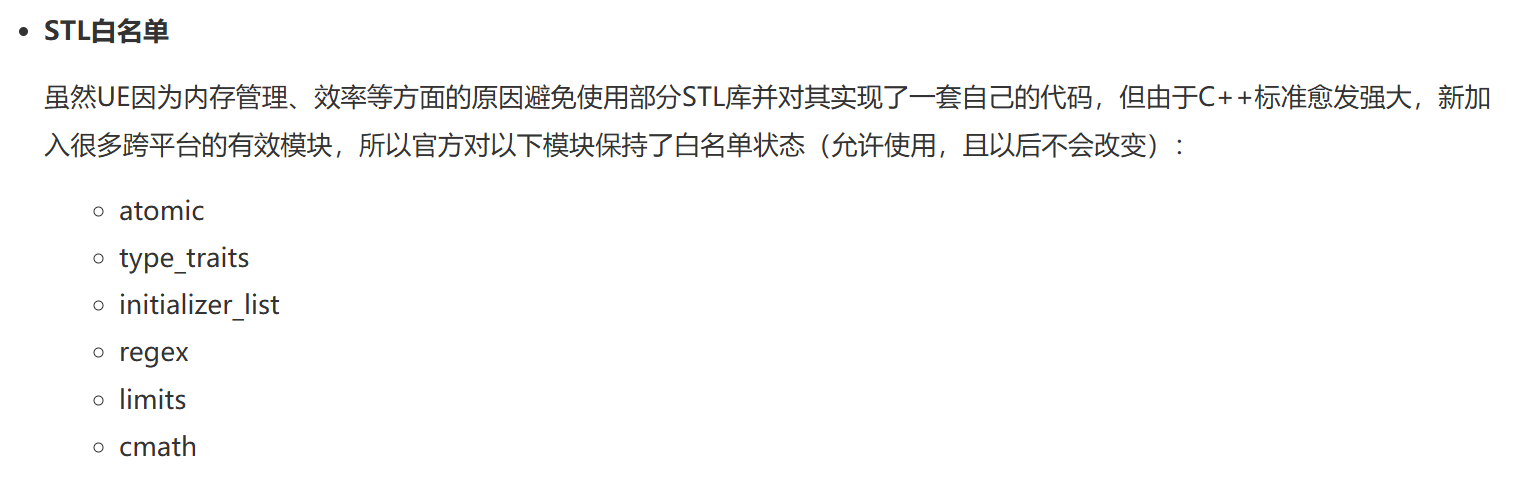
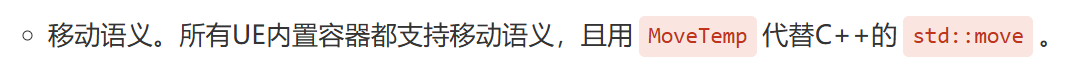


UE code standard

****

**UE对以下STL没有单独写自己的封装**

****

****

**Data structure UE —— C++**

****

****

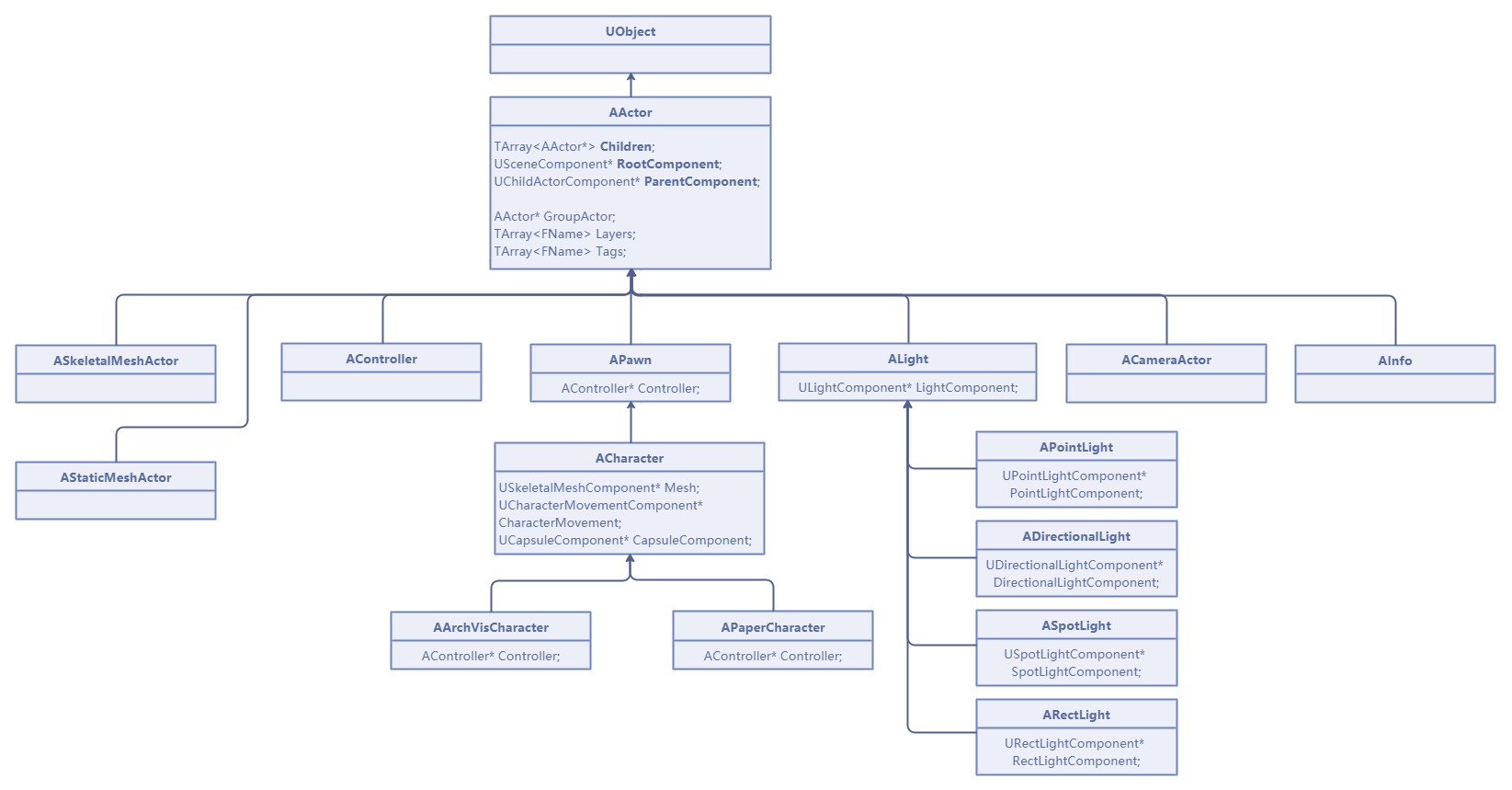
**UE坐标系**

****

**UE Macros**

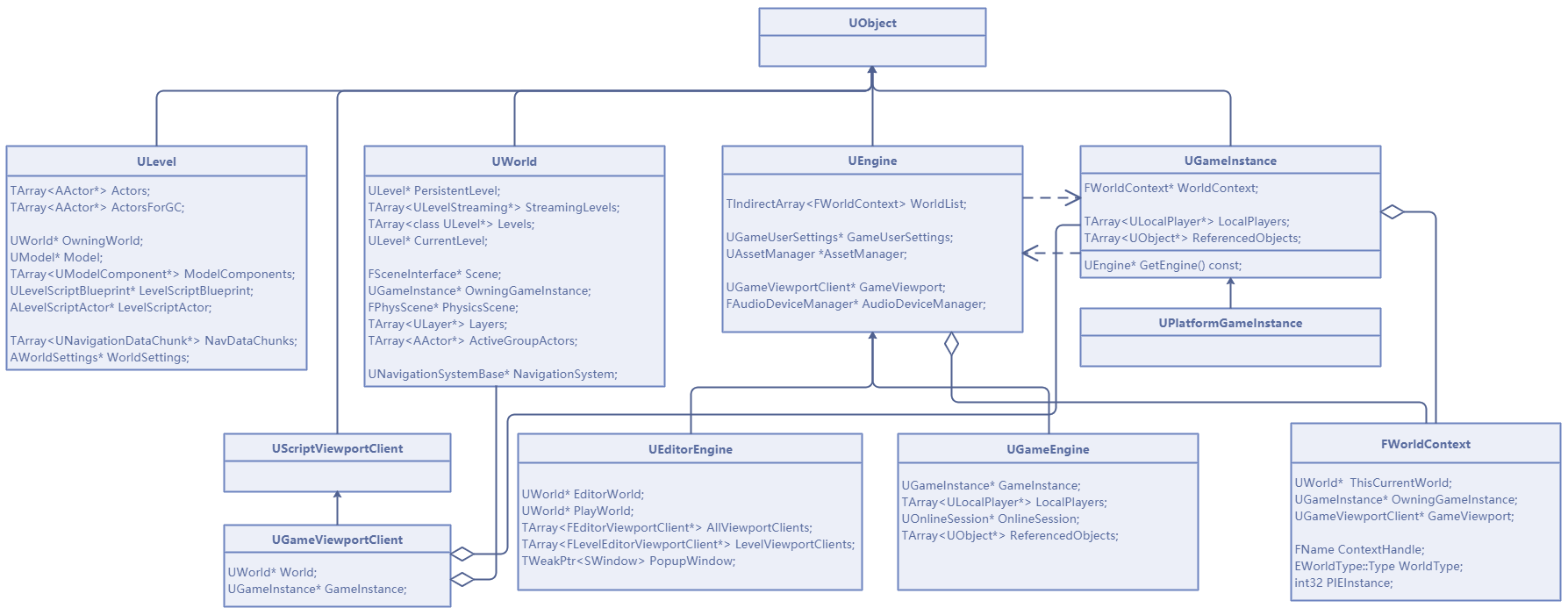
****

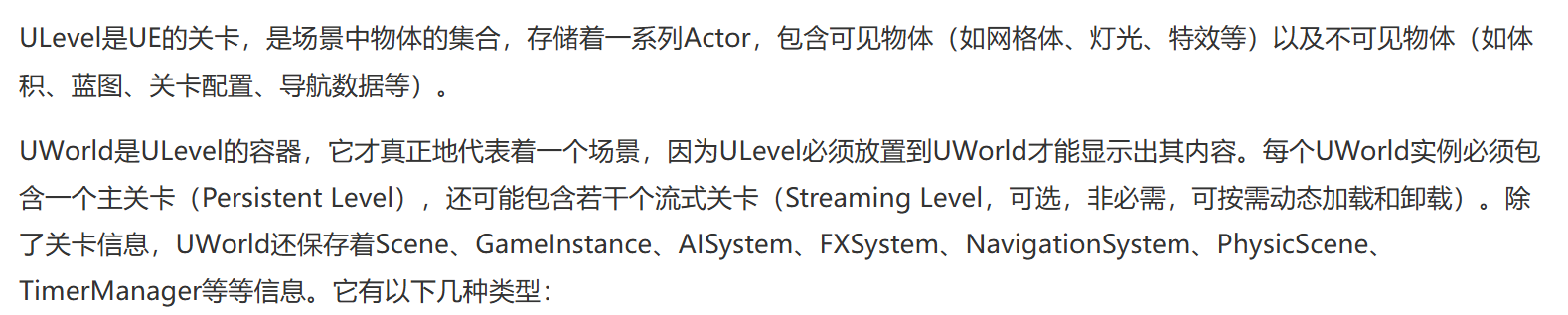
**AActor**



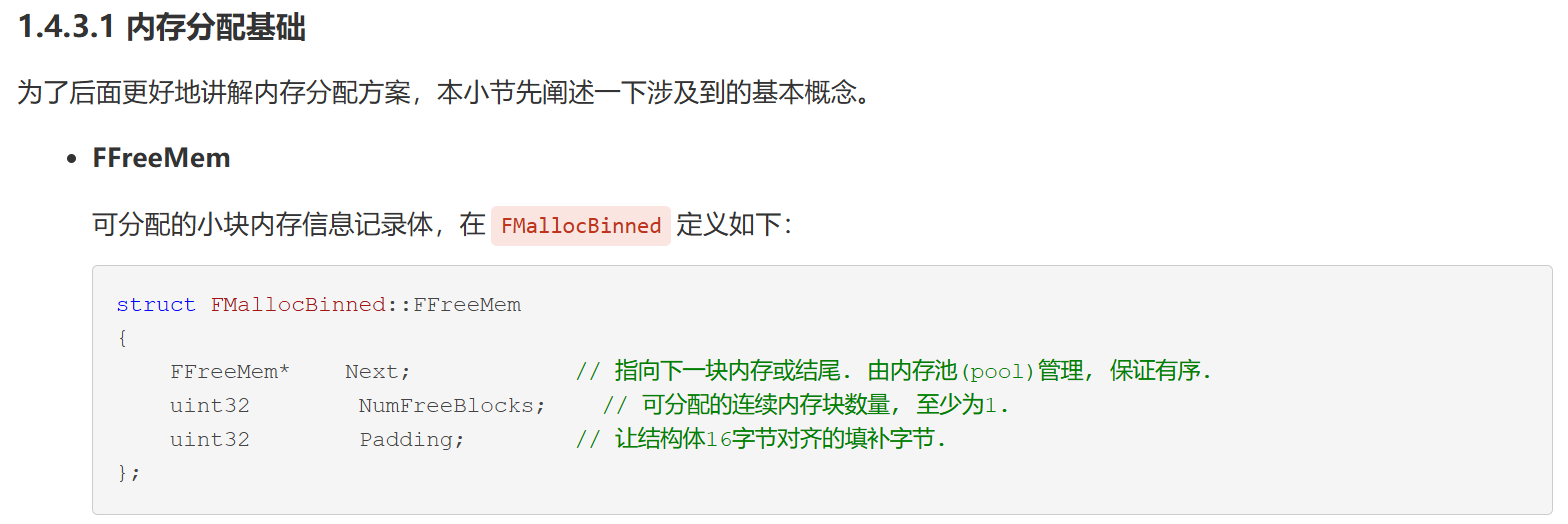
****

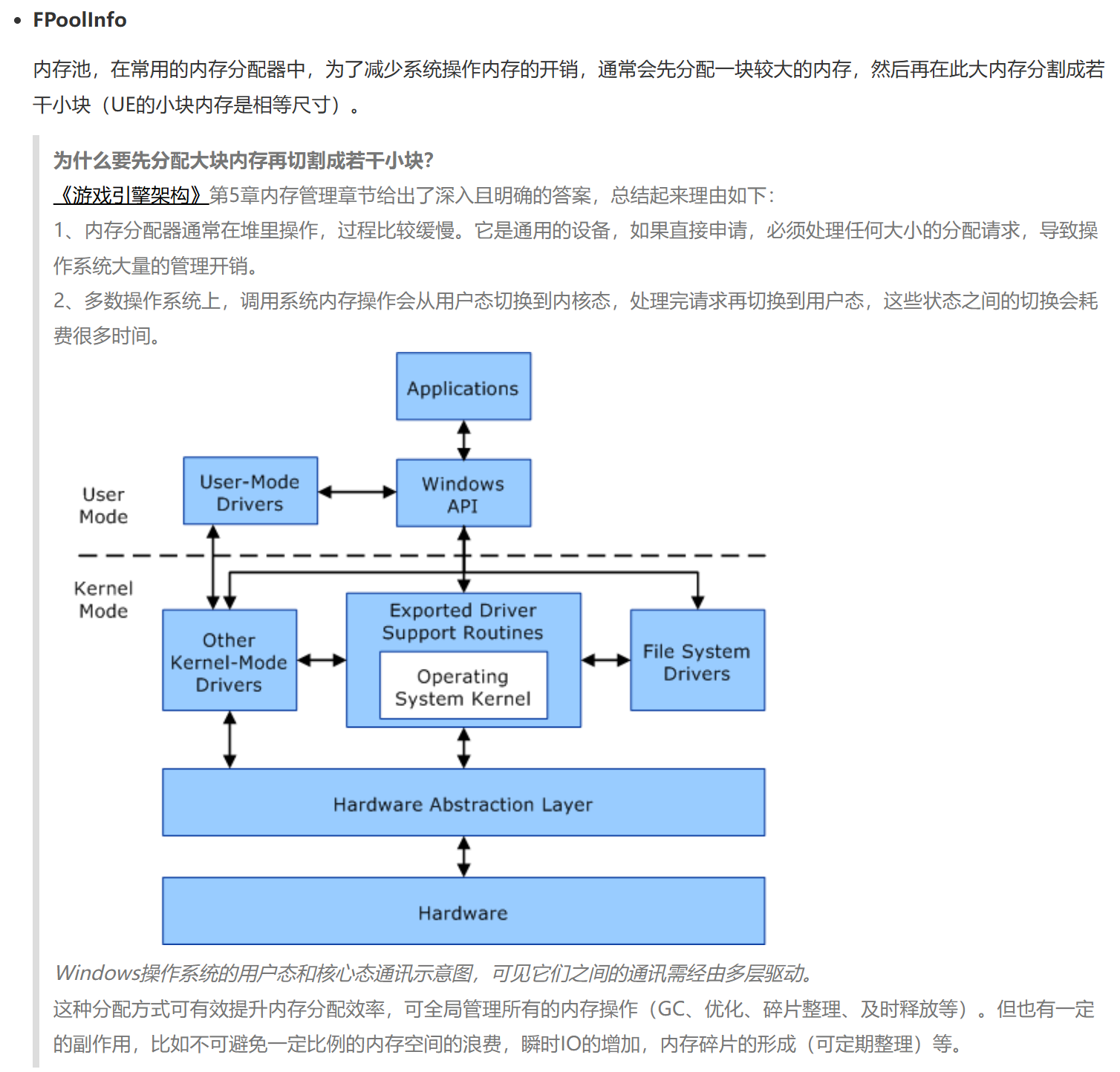
**ULevel UWorld ……**



****

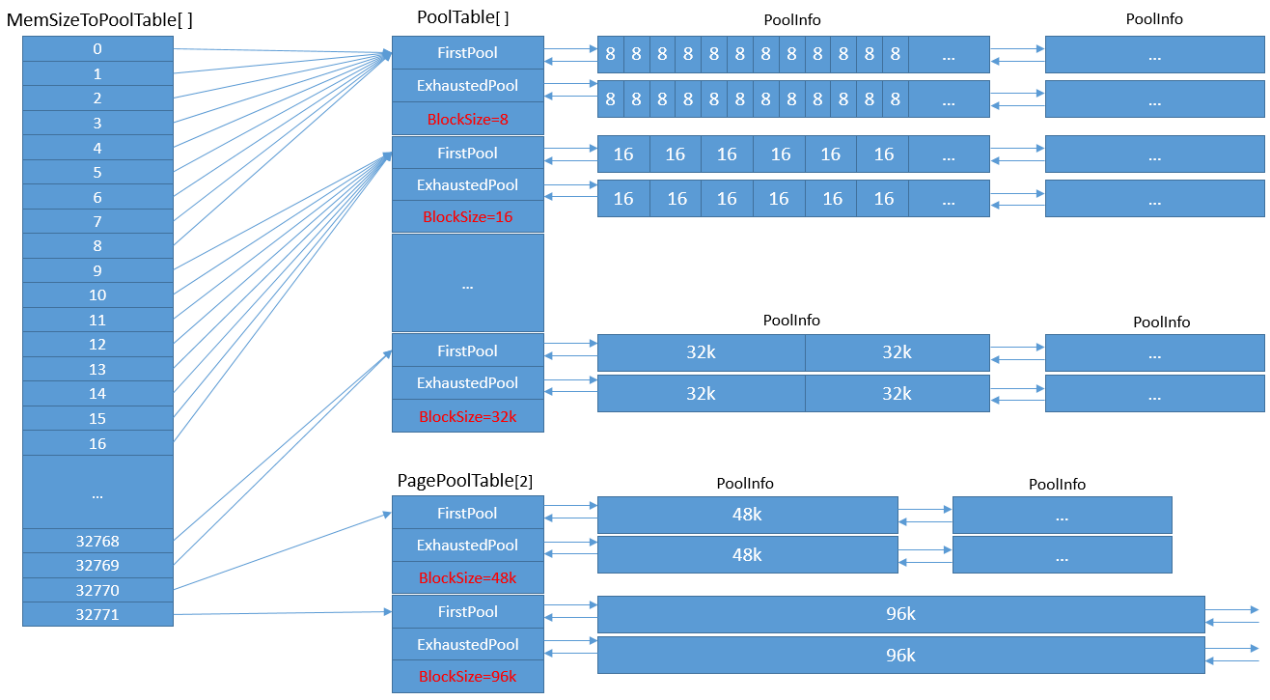
**UE Memory**

****

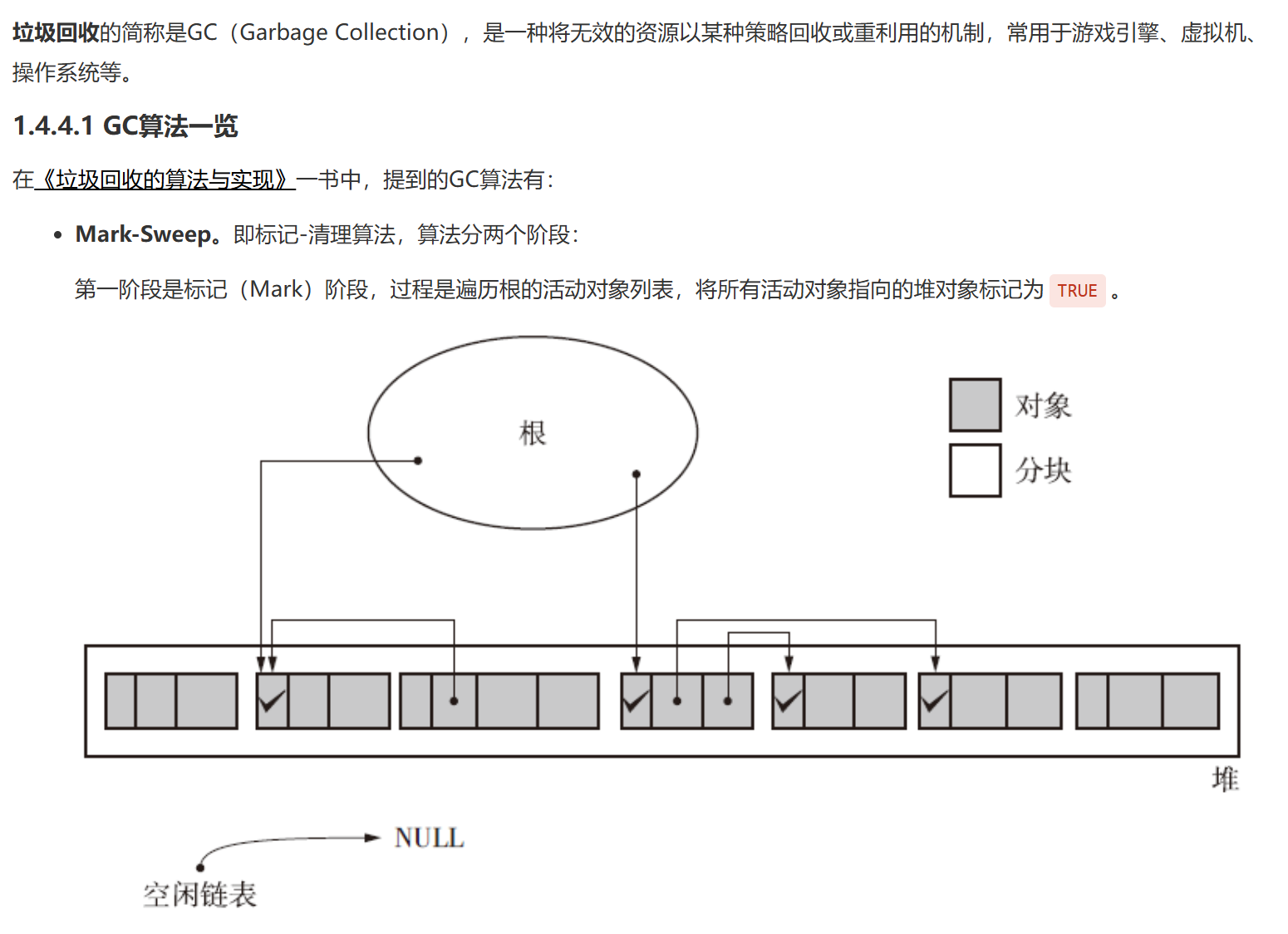
****

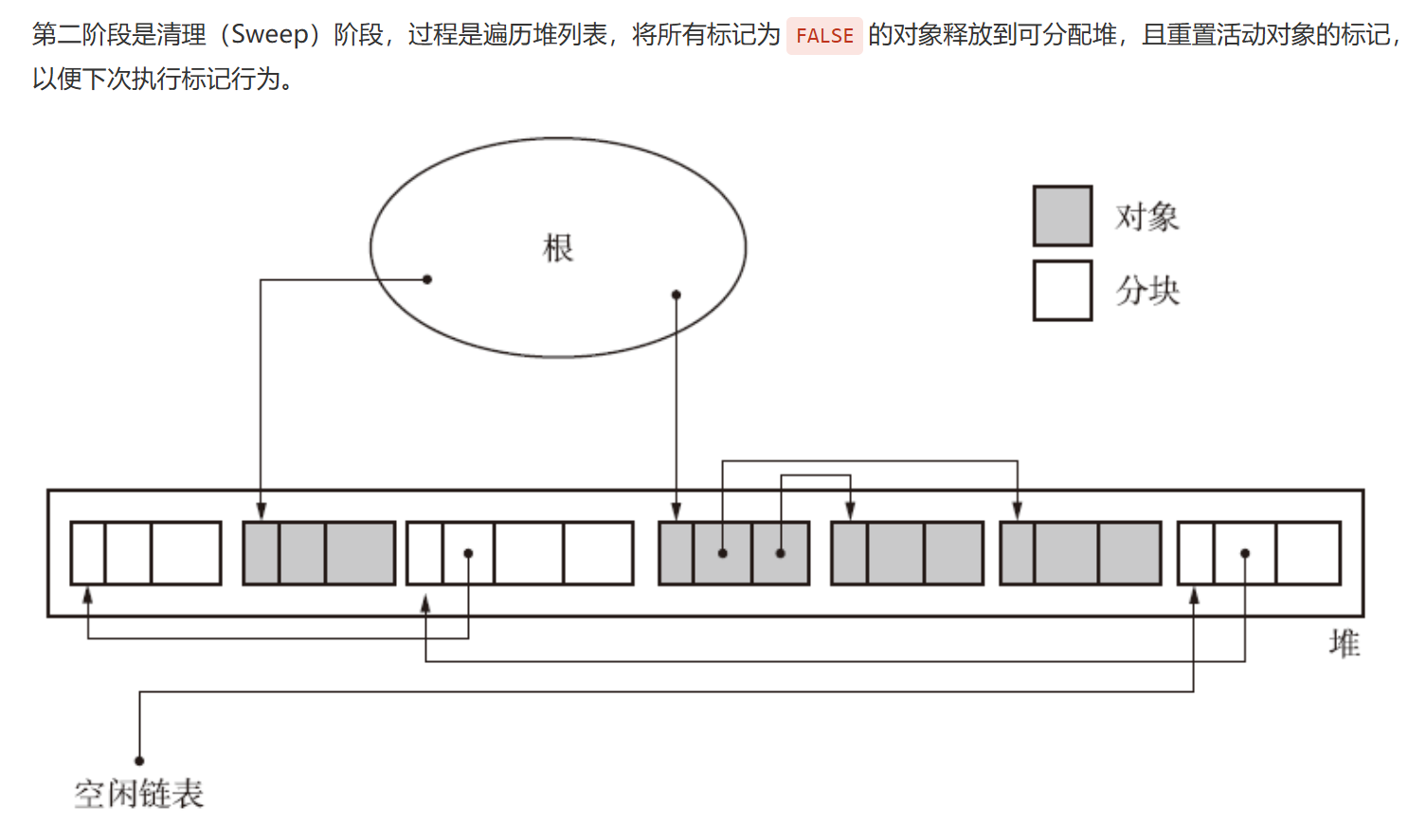
****

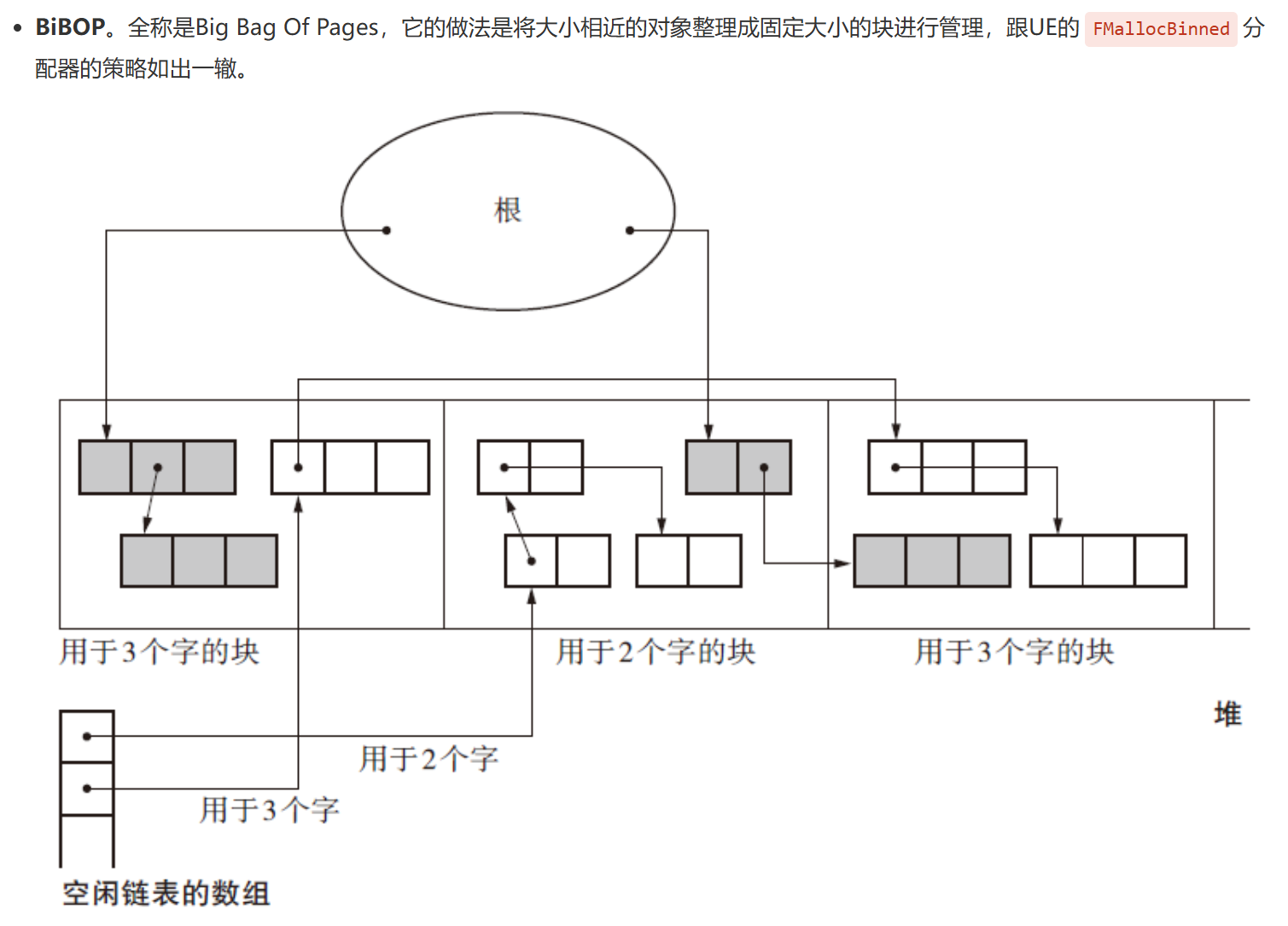
****

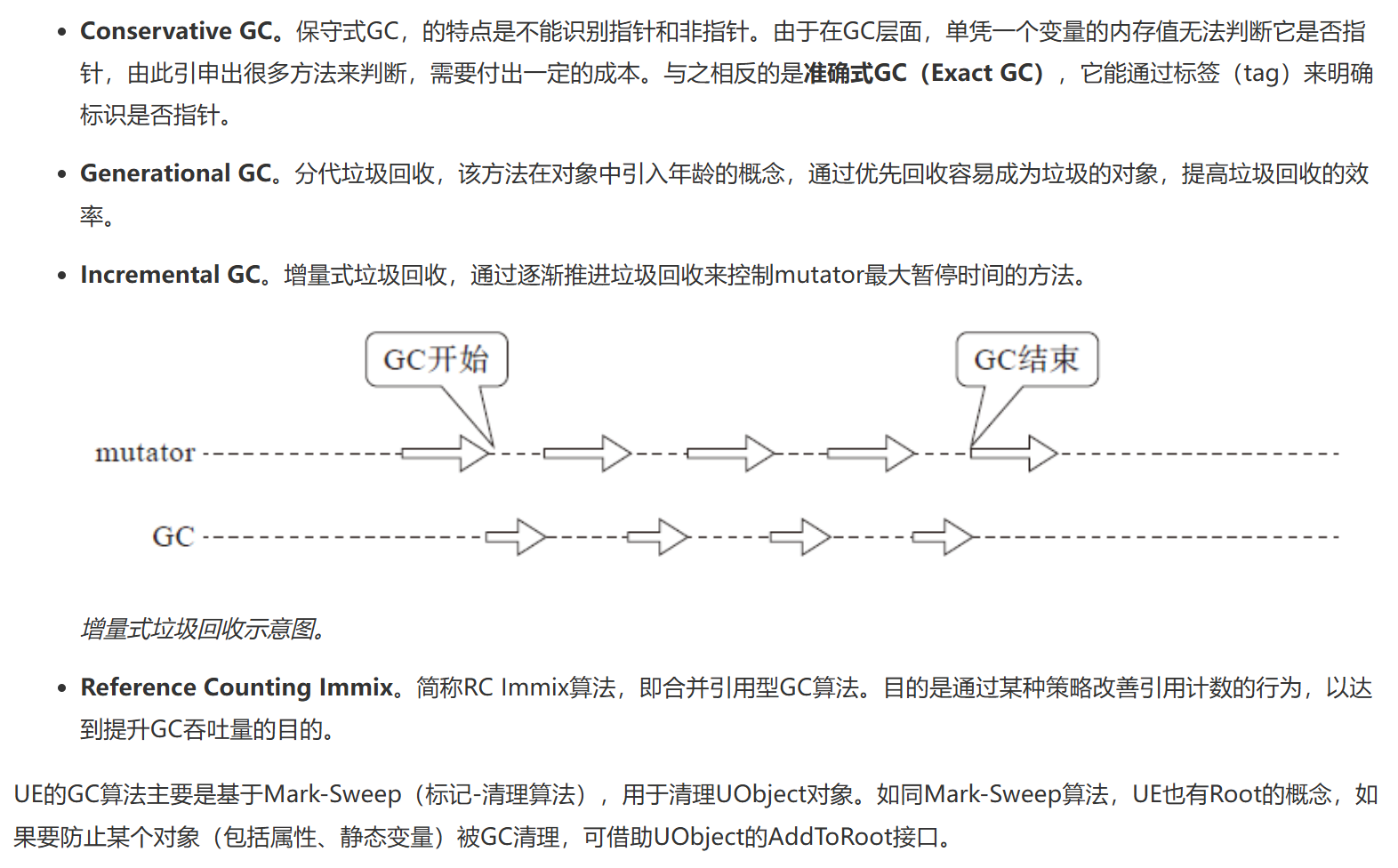


**常见垃圾回收GC （UE主要是基于Mark-Sweep的）**

****

****

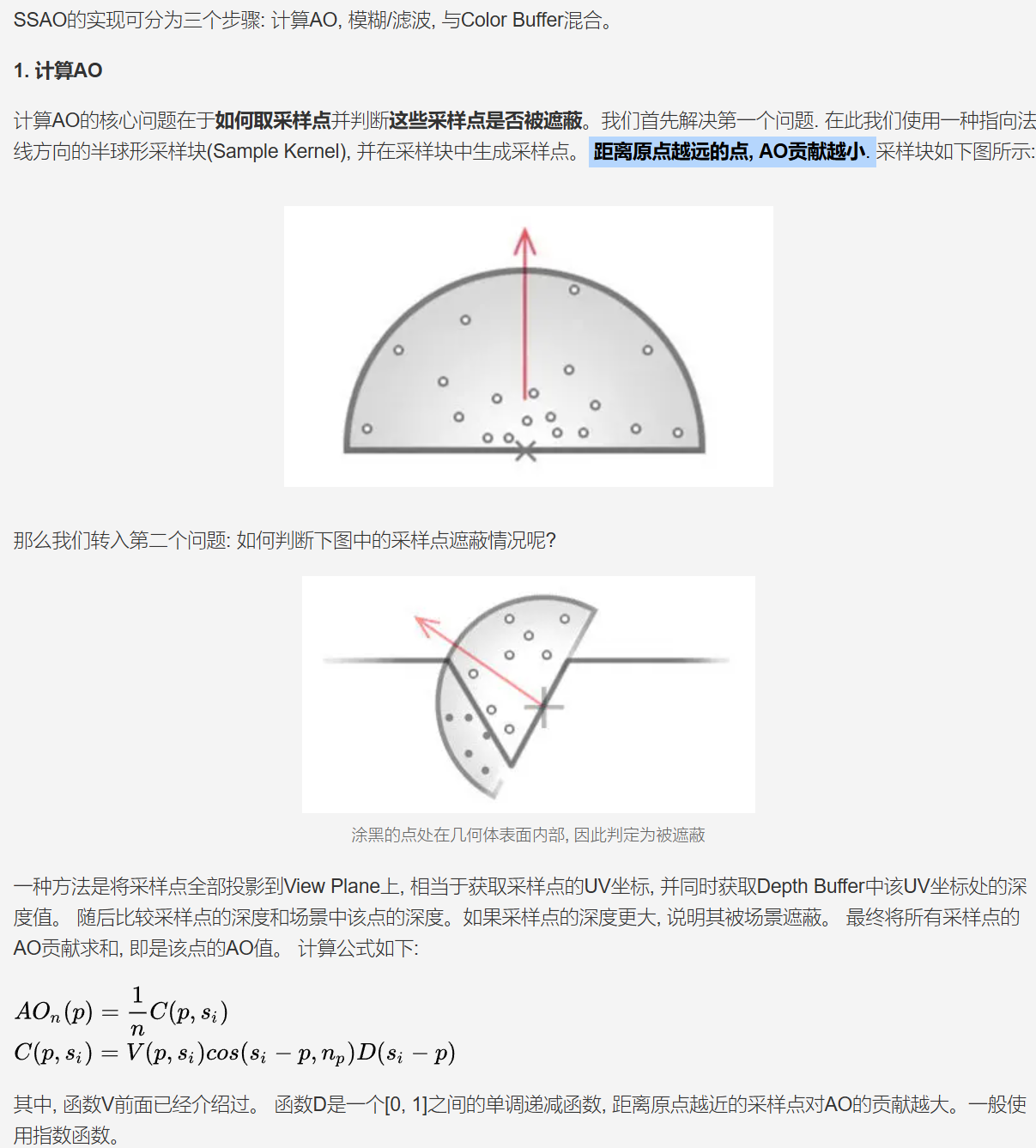
****

****

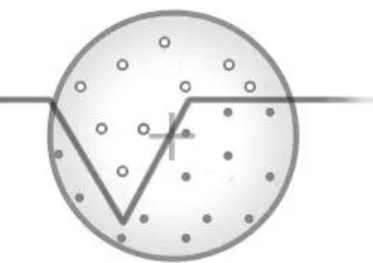
**SSAO**

[游戏后期特效#4： 屏幕空间环境光遮蔽(SSAO) | indienova 独立游戏](https://indienova.com/indie-game-development/unity-after-effect-04-ssao/)

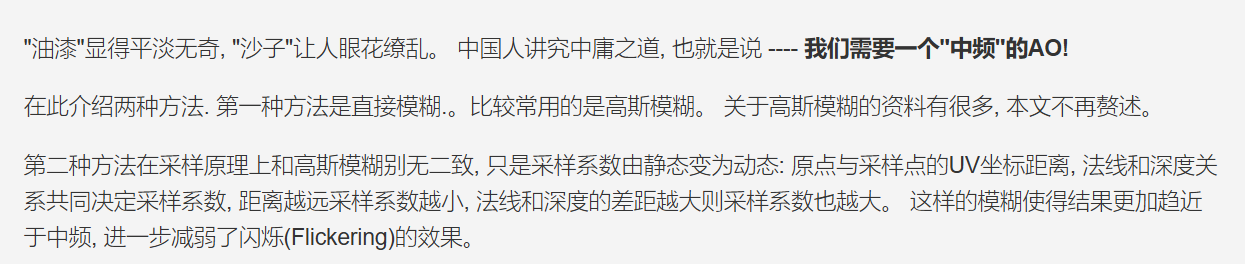
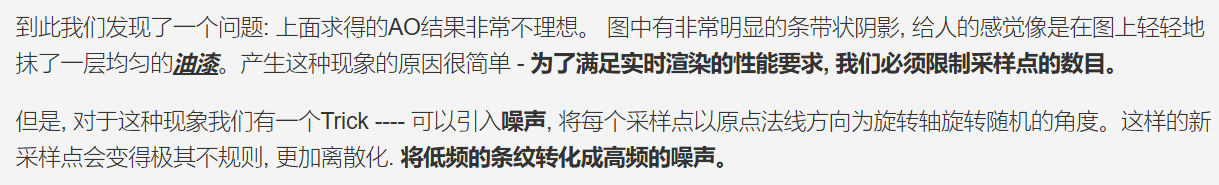
1. **采样半球，需要法线和深度**

****

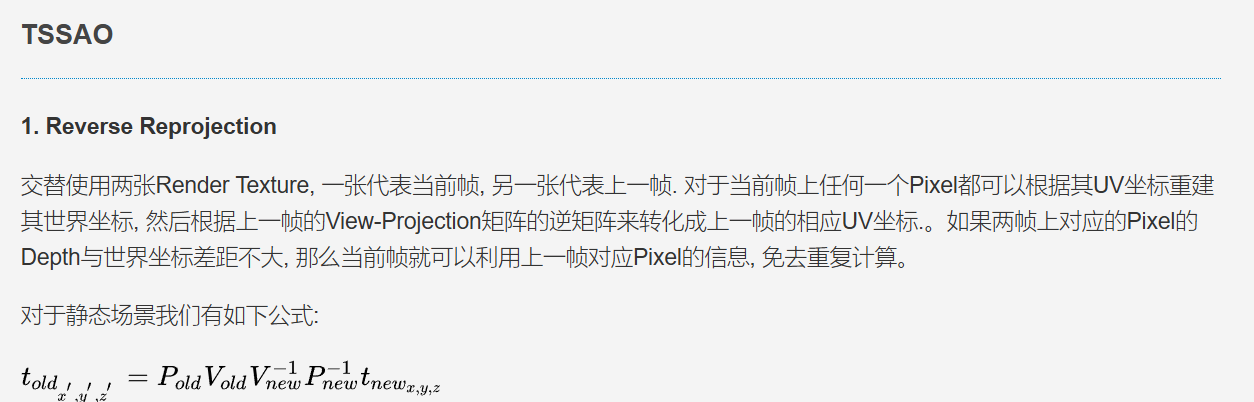
1. **只有深度，退化为最初的SSAO，采样球**

****

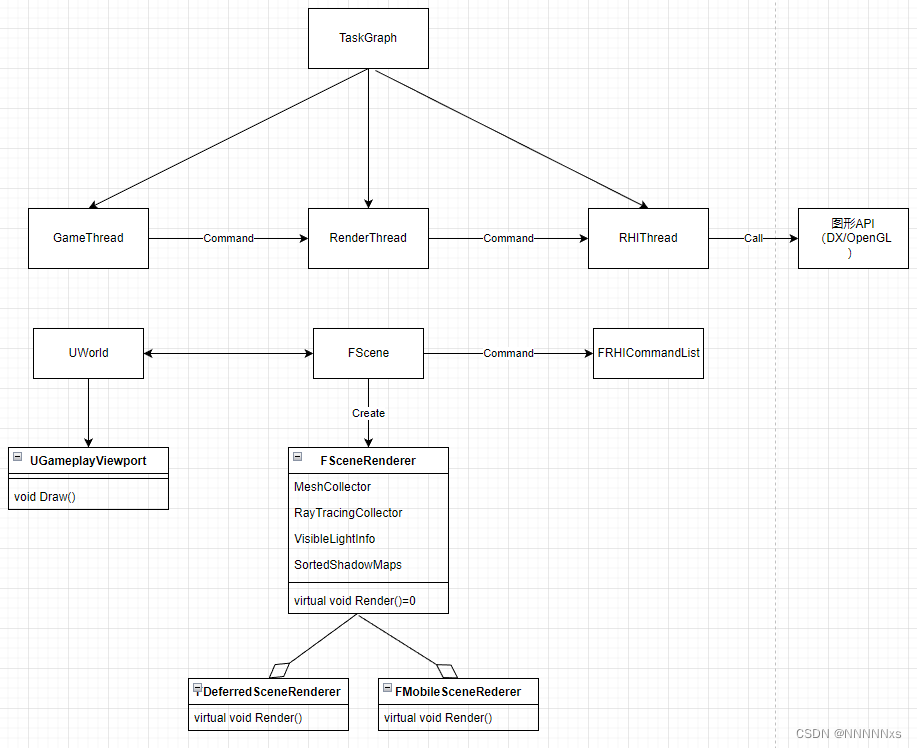
1. **由于随机的采样点如果过少会导致采样低频，采用trick，两种方式**

****

1. **TSSAO : 复用上一帧的AO**

****

场景渲染的基础是FScene，只有注册到FScene内的组件才会被渲染，其中包含了所有的场景渲染信息，并且它将根据渲染设置在每一帧中创建对应的SceneRenderer（每帧都重新创建的临时对象），其中Render为主要的渲染执行函数，Render中会调用渲染各个阶段的Pass具体执行函数。



**渲染全流程**

