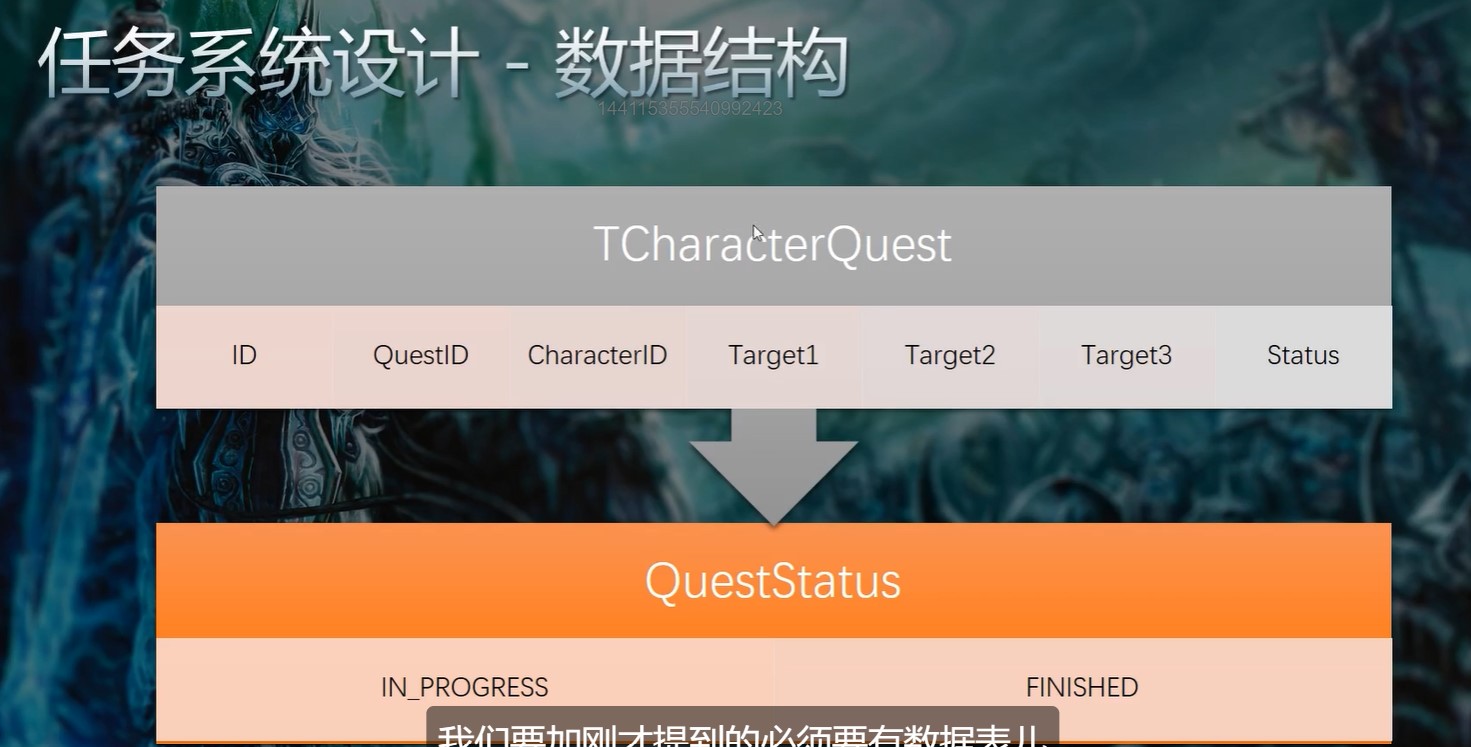
任务系统：

1. 数据结构：



TcharacterQuest：数据表

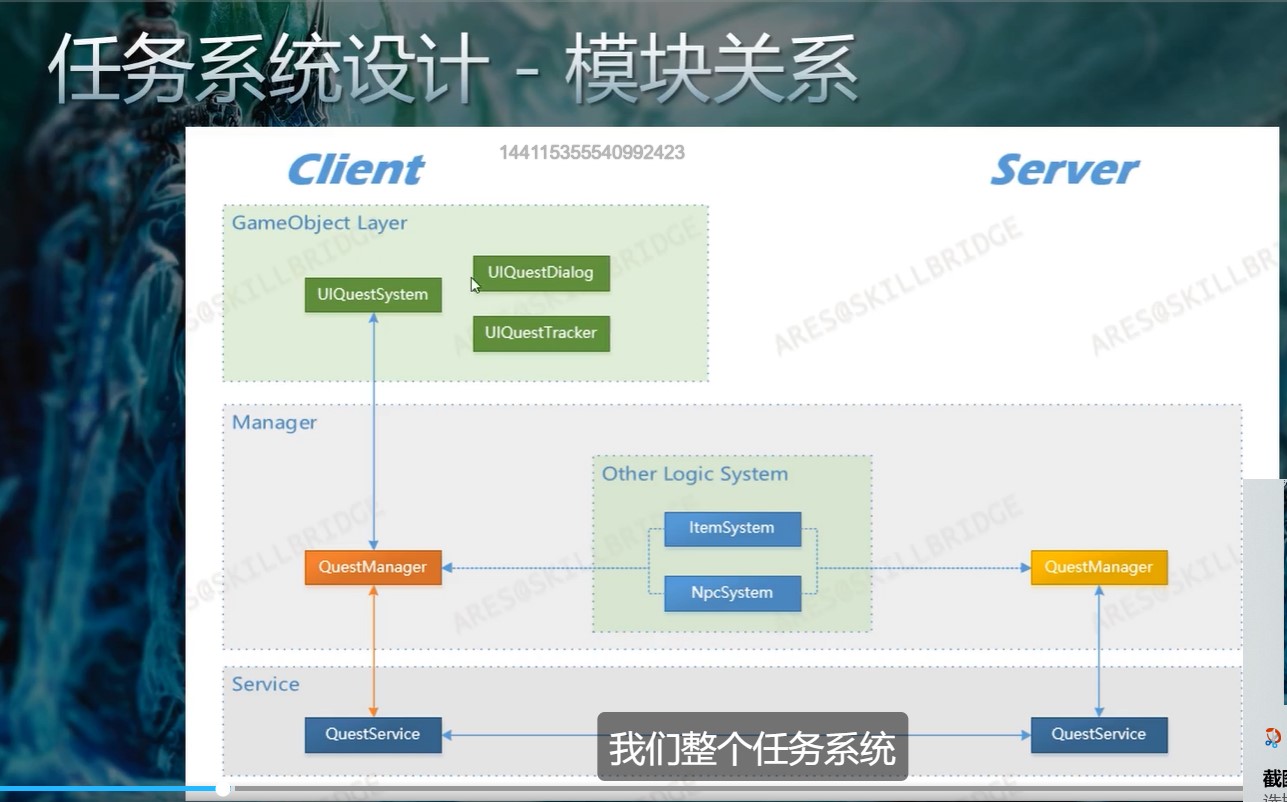
任务ID 接收玩家 任务目标 任务状态

QuestStatus：任务状态 进行中和已完成

玩家没接收这个任务服务器就不保存，只有当玩家接收任务后，服务器才会记录和保存，服务端同步记录客户端信息，比如客户端完成一个任务目标就下线了，但是没完成所有任务目标，服务端就记录目前状态。比如杀了只怪，服务端要先知道客户端杀了只怪，等客户端再发信息来时好效验，以此来防止外挂。

并且交任务是客户端向服务端发送任务目标，如果与服务端一致，服务端就标记已完成，并且发送给客户端

1. 模块关系

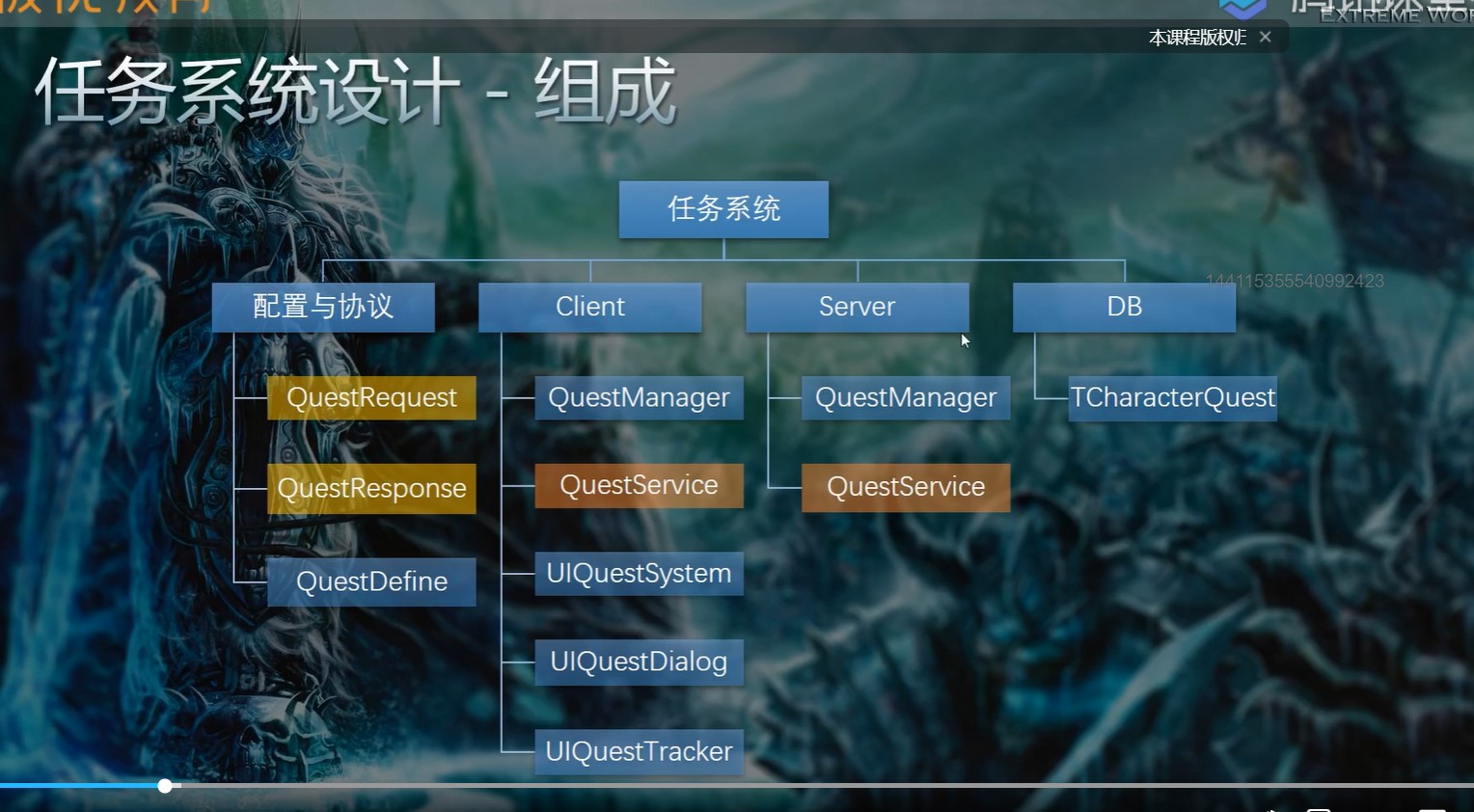


1. 任务系统分析

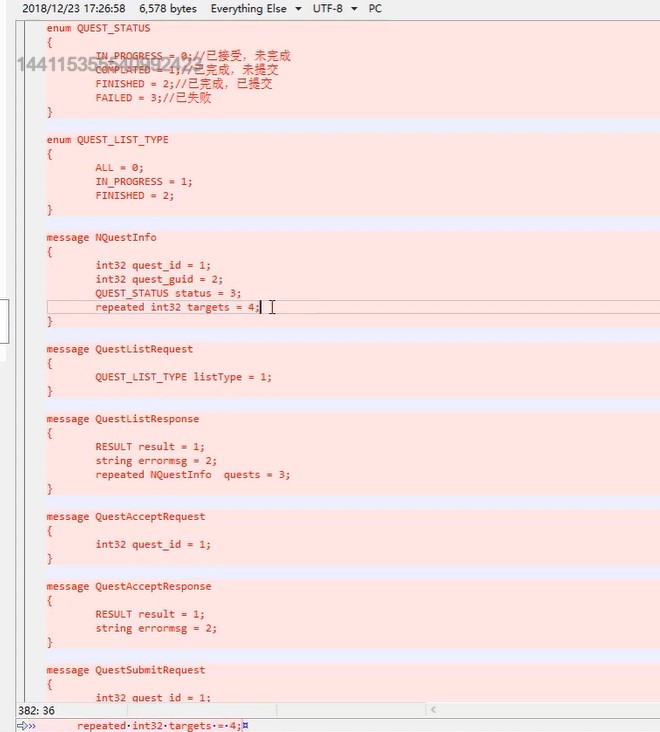


在处理任务奖励时，如果每个职业的奖励都不同，可以把每个任务做成不同id的几份来设置不同的职业只能领取不同的id的任务

1. 任务系统组成

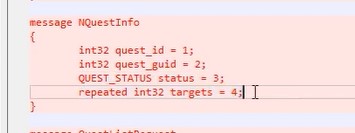


1. 协议



具体见具体协议

其中：



Quest\_id表示任务id

Quest\_guid表示唯一id或精准id，用于以后有能每天做一次或者一天做几次的任务时，防止数据库中一个任务id作为多个相同的任务id多次出现

1. 实现的细节
2. QuestManager中开放GetQuestStatusByNpc(int npcId)获取NPC任务状态和OpenNpcQuest(int npcId)打开NPC对话框于NPC系统交互
3. NPCManager中开放DoTaskInteractive(NpcDefine npc)与QuestManager交互，如果有任务就交给QuestManager。
4. 其中NPC系统中引用了任务系统而任务系统中没有引用NPC系统，引用是单向的