# 群体AI系统设计实现

## 现状

当前群体AI实现了怪物AI联动的出版，主要是通过在怪物的逻辑内加入群体管理逻辑，可以管理成员，发送群体AI命令到成员实现控制，在群体AI成员内实现命令的执行逻辑，来实现boss对小怪的控制，细节上，boss寻找小怪逻辑是通过behaviour tree的子树实现的，群体AI的数据配置在使用群体AI逻辑的单位上，比如boss

## 缺点

群体AI逻辑需要依赖boss或者小怪实现，不能由其他逻辑触发

数据配置在群体AI的使用者上，不能灵活配置

怪物制作群体AI逻辑麻烦，需要把相关的数据，blackboard变量等都添加到对应的怪物数据中

调试麻烦，做群体AI相当于做怪物AI的一部分，实际与群体的所有者boss关系不是那么密切，很多是一个决策者的角色，当boss逻辑有bug的时候，会导致群体AI的调试不能进行

逻辑开发不集中，导致群体AI逻辑分散到很多实体中，比如队员寻找，事件响应（怪物生成，死亡等影响成员数量），具体逻辑（逃跑点，包围点，围攻目标选择）等具体逻辑，制作在群体AI使用者身上，命令执行者需要对指令做出具体响应

## 方案

1.把群体AI逻辑做成像怪物，场景物件等一样的一个实体，可以通过spawn逻辑触发，动态设置owner等属性

2.群体AI通过编辑器可以通过archetype的原理来预定义属性，而且可以拖入场景，不依赖于怪物触发，由于内部逻辑使用behaviour tree，可以通过BT editor的debug功能模拟命令者发送命令

3.通过属性不同决定该群体AI实体的功能，比如围攻，攻击控制(防止怪物一拥而上)，跟随，掩护，集体跑路等

3.解除群体AI中命令者和执行者的耦合，命令者需要创建群体AI和发布命令，执行者只需要有执行逻辑即可，例如队员选择，队员死亡处理，新的怪物产生处理（队员补充）等逻辑，都通过模块化写入代码内，封装在群体AI内部，boss完全不需要关注这些逻辑，只需要用逻辑创建对应的群体AI的archetype即可

4.采用commander方式，boss负责创建群体AI和发布命令，群体AI逻辑实体负责寻找成员，执行群体逻辑和给执行者发布具体行动指令，举例制作包围逻辑，boss只需要创建包围的群体AI，该实体自动寻找队员，计算包围点，然后发布根据成员id控制不同的成员向各自的包围点移动，命令执行者不需要有群体AI逻辑，比如计算包围点，逃跑方向等，只需要按照命令指定的方式，目标去行动即可



## 具体实施

CGroupAI类， 实现的功能包括

1. 整合behaviour tree功能，可以用archetype逻辑定制使用的behaviour tree
2. 群体AI的基础逻辑，例如事件响应等一些复杂逻辑，例如当一个GroupAI被删除，他的队员会被低优先级的GroupAI接管，当一个群体AI内队员不足，新生成的满足筛选条件的怪物会被自动加入到GroupAI中，当高优先级的GroupAI产生，会自动把低优先级的GroupAI中的队员加入到高优先级的GroupAI中，同时可以用behaviour tree来定制逻辑，比如群体AI的筛选逻辑，可能不同的群体AI需要高度定制的筛选逻辑，对怪物的类型，血量，甚至各种状态都可能进行处理，无法进行统一代码编写的会用behaviour tree来实现，本质工作主要是成员管理

CGroupAIManager类，负责管理GroupAI，便于通过ID迅速查找定位对应的GroupAI，可以在怪物，其他逻辑中对GroupAI进行操纵，发送消息等，同时也是消息的派发逻辑，比如当一个GroupAI销毁时候，Mananger会把队员按照优先级一次让其他GroupAI进行挑选和加入