开发新模块流程

1. 第一步根据需求定表(config/game下的配置xlsx(config/hotfix目录是用来做客户端代码热更用的，上线前无需))，然后执行

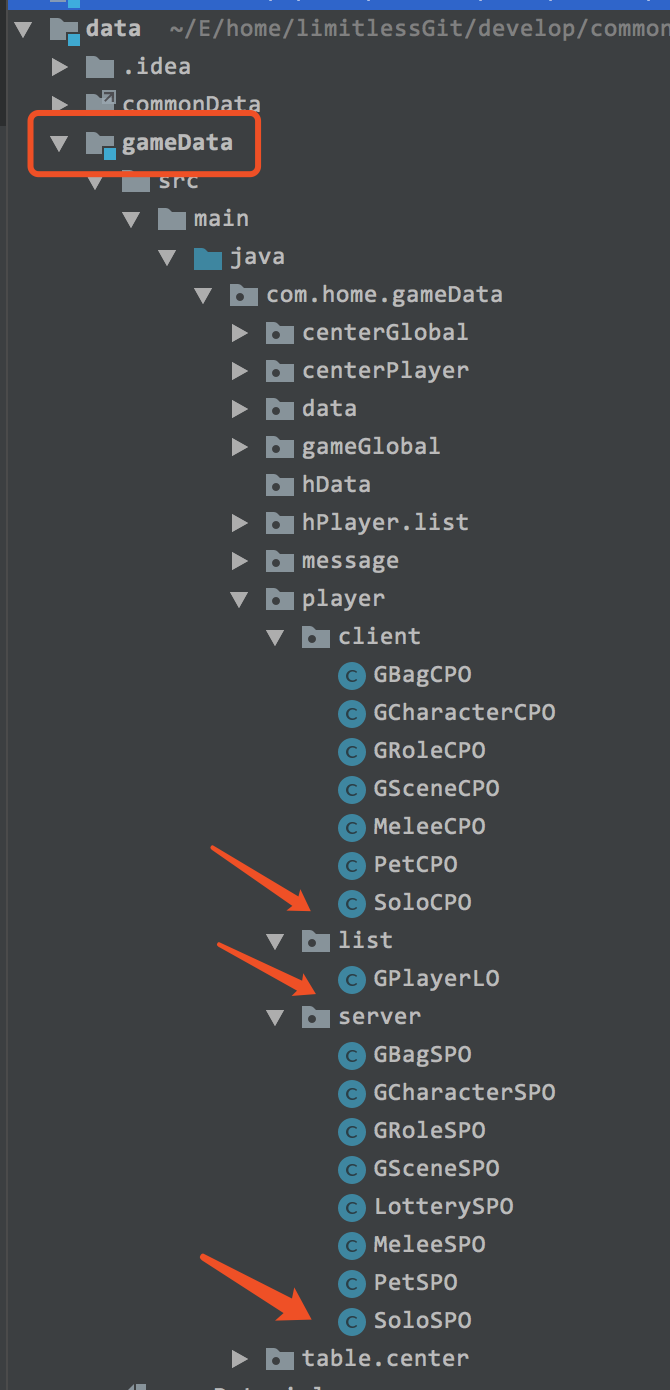
develop/server/tools/configExport.py,生成双端配置类

配置使用说明，参阅 工具类/《配置表说明.docx》

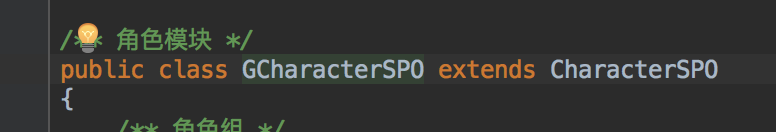
1. 然后创建双端模块数据，在数据工程(develop/common/data)中，

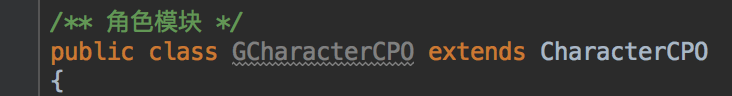
定义好模块的SPO,CPO,并加到对应的LO里(如GPlayerLO),然后运行 dataExport.py生成数据类和模块Part类。如下图

解释下关于SPO和CPO，SPO为存库的模块数据部分，CPO为推送客户 端 的部分(如果是启用离线模式的项目，CPO也作为客户端存本地数据)

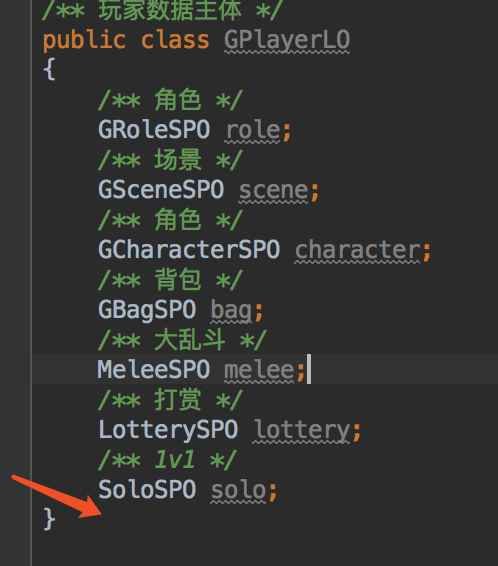


注意添加时，区分清楚是在gameData工程内，还是在commonData工程内，如果你要添加的模块，在C层已有，则分别在你新建的SPO和CPO中，继承对应的模块即可(并且命名必须为GXXXSPO和GXXXCPO，而且属性名也必须和C层的一致)，如

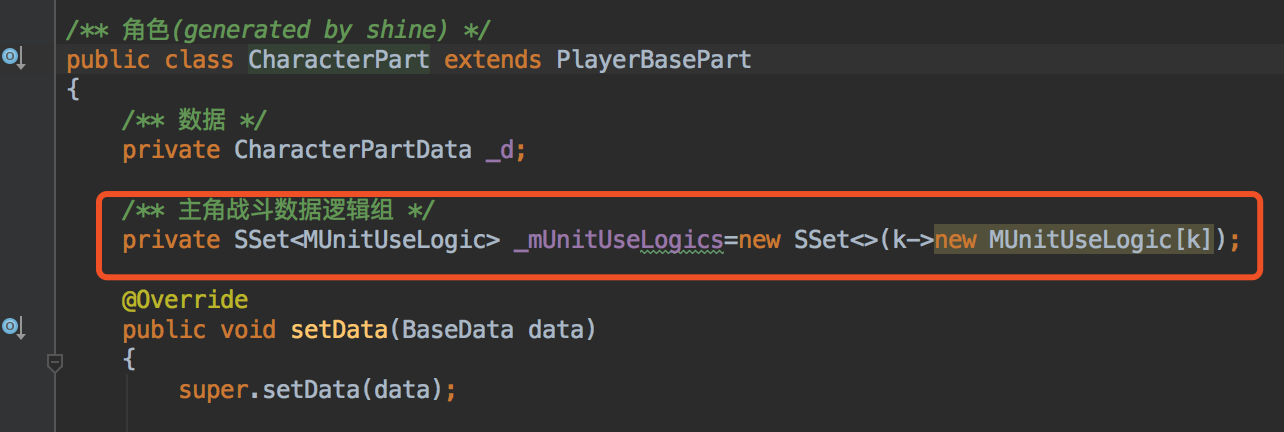




然后在LO中添加，如



1. 双端的Part类都会生成一些常用的接口方法，包括construct,init,dispose,afterReadData等，接口的含义可看方法注释，Part类的基类也会有一些不常用但存在的接口，如需要时可手动override使用，如beforeLogin,beforeEnter 等,可查看Part类的基类的内容
2. 创建完Part类后，在服务器Part的onNewCreate中写新模块创建数据赋值，然后在afterReadData中读数据库数据后的处理，然后在toWriteClientData写对客户端初始化数据的额外处理，客户端也是一样，在afterReadData中写收到服务器数据的解析。（过程中需要什么属性直接在此Part类上增加你的对象属性）。
3. 如果需要一些不参与序列化的数据,或是临时数据，直接在对应的Part类中创建private属性即可如

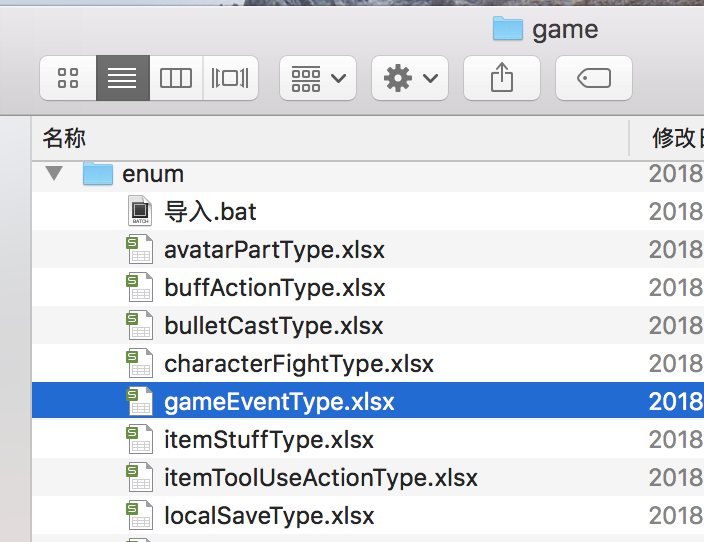


1. 每个Part内都有名为\_d的数据对象，该对象为你编辑的SPO/CPO的数据实体，可直接调用来，取值，或设置,如

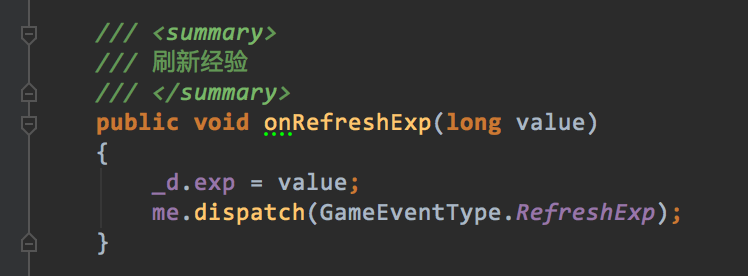


1. 接下来制定双端通信协议(消息)(详情见 工具类/ 《数据工程使用说明.docx》)

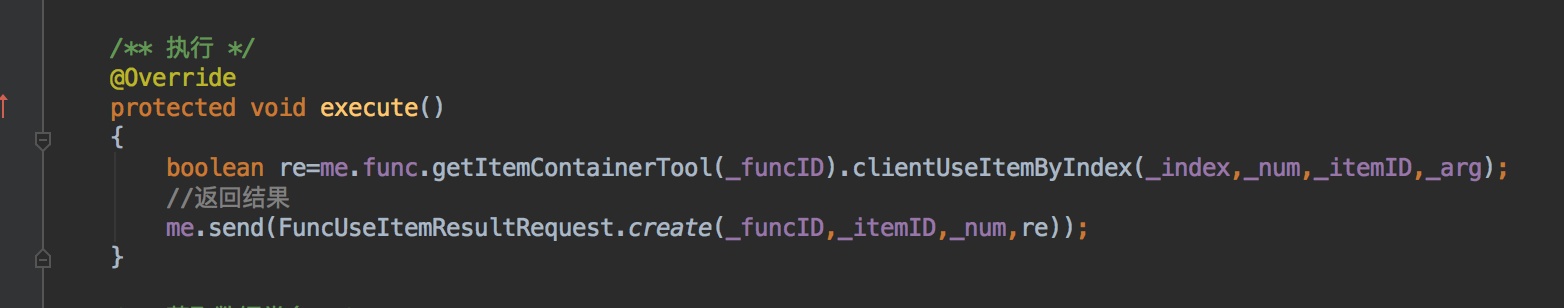
以及客户端内部gamEvent新增类型枚举,在config/game下



补充你需要的数据(描述中，最好注明参数类型)，以及dispatch



1. 开发逻辑功能，处理方法可以写在XXXResponse的execute里，也可写在Part里，然后exectue时调用之,如



1. 功能开发完后，同时补充clientProject的机器人逻辑，增加响应的机器人测试模式。
2. 接下来是客户端的显示的部分，参阅 使用部分/《UI制作流程.docx》