**机器人简单介绍文档**

[编译流程和运行流程](#_Toc12600)

[代码介绍](#_Toc7365)

[机器人主要流程](#_Toc11024)

游戏代码注意事项：

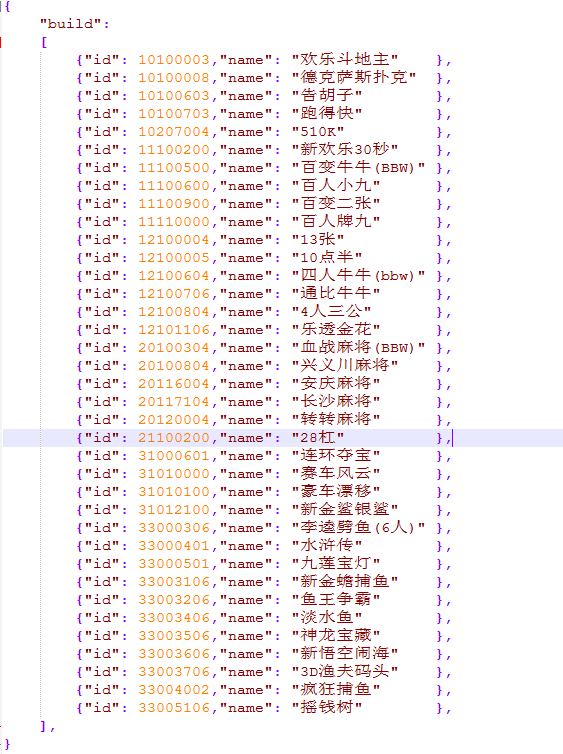
1. 捕鱼类游戏需要实时更新CloveSendClass中m\_pGameInfo 的 用户信息的i64Money字段
2. 文件编译不通过的地方注释即可
3. 游戏代码均需要放在ClientGameDlg.h，ClientGameDlg.cpp,UpgradeLogic.h, UpgradeLogic.cpp和UpgradeMessage.h, 通用函数可以考虑放到AIComFunction.h中
4. 游戏代码仅处理框架和游戏消息

编译流程和运行流程

拷贝机器人代码到相应的项目根目录下

1. 编译

a、在配置文件build\_config.json中配置需要编译的机器人，如下



b、执行编译脚本 ai\_build.bat

重新编译执行 ai\_rebuild.bat

1. 运行

第一步，将ini目录下的RobotSet.ini拷贝到生成的机器人目录下

第二步，打开Platform/client目录下的工程，生成AfcSocket.dll

第三步，拷贝AfcSocket.dll到生成的机器人目录下

第四步，对RobotSet.ini进行配置

第五步，运行机器人

# 代码介绍

1. Common目录

AIComFunction.h 一些实用的通用函数

PlatConfig.h 单实例全局配置项类,分为静态配置和动态配置，静态配置需要手动配置文件，动态配置通过UI修改

PlatglobalData.h 运行时的一些动态信息

PlatLog 和 PlatLock，日志和锁类

1. UI目录

UserStatusDialog.h 观察当前所有机器人的状态

DialogTaskList.h 查看和编辑所有任务

DialogTaskEdit.h 编辑单个任务

1. PlatformLogic

BaseRoom.h TCP客户端连接基类, 接收和发送数据

Auto\_Dlg.h 主对话框，继承自BaseRoom.h, 处理与AServer和Mserver相关的业务, 管理所有AI账号

GameListCtrl.h AI账号登陆的游戏信息管理类

Gameusermanage.h 在线账号管理类

GameRoomEx.h 房间类,继承自BaseRoom.h 每个AI账号一个房间, 处理存取钱，站起，坐下和配桌逻辑

1. GameLogic

具体游戏代码文件

# 机器人主要流程

从AServer获取MServer的登陆地址（只有第一个机器人会有此操作，后续机器人都沿用此信息）

1. 登陆流程

由定时器TIMER\_CHECK\_UNCONNECT\_ACCOUNT,每一秒钟检测是否有未登陆的账号，是否需要切换任务

2. 存取钱流程

登陆大厅成功之后检测是否需要存取钱，若需要则执行此操作；

游戏结束时检测是否需要存取钱，若需要离开房间，由下次登陆大厅存取钱；

捕鱼类游戏，通过定时器TIMER\_FISHGAME\_AIBANK检测是否需要存取钱，若需要离开房间，由下次登陆大厅存取钱；

3. 坐下和站起流程

坐下时和游戏结束后未站起启动状态检测定时器，TIMER\_CHECK\_USERSTATE防止账号一直坐桌;

捕鱼类游戏，由定时器TIMER\_FISHGAME\_LEAVE防止账号一直坐桌;

4. 配桌逻辑

桌子上机器人未达到配置的每桌限制的情况下:

若不允许与真人同桌，则选择机器人最多的桌子。

若允许与真人同桌，优先选择真人最多的桌子, 否则选择桌子上人数最多的桌子。