# 主要程序说明：

A 服务器 （中心服务器）-- HNGameGate.bcf 主要功能为将基础配置发到客户端，通知客户端 M 服务器地址及其它配置。Cocos客户端未使用。

M 服务器 （登入服务器）-HNGameLogon.BCF 主要功能为负责玩家登入平台， 及道俱，玩家在大厅内玩家操作回应 （需要建立数据库联接，并操作数据库）

G 服务器 (游戏服务器) - HNGameLocal.BCF 服务游戏房间内操作。（需要建立数据库联接，并操作数据库）加载每个游戏 DLL

大厅.exe 登入玩家进入平台，并启动游戏启动器.exe

游戏启动器.exe 调用游戏 ico，与大厅使用管道通信。游戏启动器启动游戏。

# 平台启动运行流程：

玩家启动大厅，程序读取 HNgame.bcf A 服务器地址，与 A 建立联接，建立联接后，A 服务器将 M 服务器地址，端口及其它配置发到平台客户，客户端接收后，断开与 A 联接。创建 MainRoom，与 M 建立联接，弹出登入框。输入登入信息，点确定，将密码加密，发送到 M 服务器，服务器接收后，压入数据库处理队列，调用数据存诸过程判定登入合法。数据库处理过程完毕后，依结果向客户端

返回，正确登入 M 记录用户 I d 及 Socket 索引。M 写数据库标记玩家登入。客户端收到登入成功后界面表现，并向服务器发送请求游戏列表请求。M 服务器发送游戏列表。客户端接收，接收完成显示，隐藏正在登入提示框。当右边游戏列表 被占击时，向服务器发送请求房间列表。

点击进入房间，建立 GameRoom，GameRoom建立与 G 联接 Socket。联接成功，再将玩家信息，登入密码发送到 G。压入数据库处理队列处理玩家登入请求（存

储过程执行），存储过程会对密码，是否重复登入等进入检测。返回成功后，发送房间信息到玩家客户端。客户端表现。

玩家点击桌子坐到坐子上。向 G 服务器发送坐下消息，服务器返回，客户端建立 GameFrame，GameFrame建立 IPC管道通信;

GameFrame 启动游戏启动器进程。游戏启动器程序启动后建立 IPC ;IPC联通后，游戏启动器通过要加载的游戏是否百宝湾游戏来确立不同的初始化游戏； 加载游戏后，游戏与服务器的消息全部都是通过 ipc 发到平台再由平台通过

socket发到服务端，消息接收一样；

# 游戏运行游戏程：

玩家点击坐位，大厅向服务器发坐下消息，服务器验证返回,服务器玩家加入桌子玩家列表，大厅启动游戏启动器，游戏启动器，加载游戏，并向服务器发

送客户端自已信息（ASS\_GM\_GAME\_INFO），是否允许旁观。服务端接收后设置游戏 状 态 ， 调 用 游 戏 OnGetGameStation ； 正 式 进 入 游 戏 流 程 。 通 过

OnGetGameStation发送游戏状态到玩家客户端，客户端 SetGameStation接收并设置游戏场景，若是未开始游戏，玩家点击开始按钮，游戏向服务器发送举手消

息。服务器接收举手消息，依条件判定是否可以开始游戏，达到开始条件，调用

GameBegin虚函数。游戏执行游戏服务器的 GameBegin，游戏开始，进入玩游戏流程

# 游戏框架：服务端：

游戏逻辑类（即桌子，继承服务端 GameDesk 类）负责游戏的流程走向，与

客户端消息交互）重要虚函数：

GameBegin(BYTEbBeginFlag) 函数 游戏开始函数，一张桌子开始游戏，此函数为游戏入口，为平台调用

GameFinish函数已调用

游戏结束处理函数，一般结算在此函数中进行，游戏自

HandleNotifyMessage函数 游戏消息处理函数

OnGetGameStation函数 玩家进入游戏向客户端置场景功能，包括游戏没开始或者断线重回都从要进入这里，然后 消息到

IsPlayGame 判定当前是否在游戏中，可在游戏里重新实现，

UserLeftDesk 玩家离开桌子回调函数,

UserNetCut 玩家断线回调函数

# 客户端：

游戏逻辑（即现有 GameClientDlg，）负责客户端游戏流程处理，服务端消息接收，玩家 U I 操作处理发送

视图（GDI 游戏有，百宝湾游戏无此类,百宝湾与游戏逻辑在一块）负责界面显示

SetGameStation 函数 设置场景函数，相对应服务端 OnGetGameStation 的消息处理

HandleGameMessage函数 服务端消息处理函数

GameUserLeft 玩家退出

GameUserCome 玩家进入

# 其它：

GameMessage 公共消息定义，数据包协议定义，

游戏规则类，（牌型判定，是否可以出牌等一系列逻辑算法），一般可以与客

户端共用