### 存活检测【系统】

描述：后台系统检测服务器存活；

协议：{“cmd”:1}

服务器收到该协议后，会原样发回报文；

### 在线人数【系统】

描述：后台系统查询在线人数；

协议：{“cmd”:2}

服务器收到该协议后，发回响应，格式如下：

{“cmd”:2, “online”:100}

### 战绩上传【系统】

描述：游戏服务器通知WebService存储战绩；

总结算协议：

{

“cmd”:4,

“tid”:桌号,

“tidalias”:显示的桌号,

“scores”:

[

{“uid”:1, “name”:”xxx”, “seatid”:1, “score “:100},

{“uid”:2, “name”:”xxx”, “seatid”:2, “score “:200},

{“uid”:3, “name”:”xxx”, “seatid”:3, “score “:0},

{“uid”:4, “name”:”xxx”, “seatid”:4, “score “:-300}

]

}

每局结算协议：

{

“cmd”:5,

“tid”:桌号,

“tidalias”:显示的桌号,

“round”:当前回合,

“bankerseatid”:1,

“iswin”:1, // 是否胜利，0表示流局

“fromseatid”:1, // 流局为-1

“fromcard”:1, // 流局为-1

“zhaniao”:

[

{“card”:1,”seatid”:1,”iszhong”,1},

{“card”:2,”seatid”:2,”iszhong”,0}

]

“scores”:

[

{“uid”:1, “name”:”xxx”, “seatid”:1, “incscore“:3,”types”:[1]},

{“uid”:2, “name”:”xxx”, “seatid”:2, “incscore“:-1,”types”:[]},

{“uid”:3, “name”:”xxx”, “seatid”:3, “incscore“:-1,”types”:[]},

{“uid”:4, “name”:”xxx”, “seatid”:4, “incscore“:-1,”types”:[]}

]

}

### 牌局记录【系统】

描述：游戏服务器通知WebService记录牌局；

协议：

{

“cmd”:3,

“tid”:桌号,

“round”:第几局,

“records”:

[

{“type”:记录类型,”record”:”相应类型的base64:},

{“type”:记录类型,”record”:”相应类型的base64:}

]

}

牌局记录的类型和内容格式定义如下：

初使化 0

{

"players":

[

{"uid":1,"name":xxx,"seatid":1,"score":1000},

{"uid":2,"name":xxx,"seatid":2,"score":1000},

{"uid":3,"name":xxx,"seatid":3,"score":1000},

{"uid":4,"name":xxx,"seatid":4,"score":1000}

]

}

开始发牌 1

{

"banker\_seatid":1,

"banker\_type":0,

"decks\_count":55,

"player\_cards":

[

{"seatid":1, "hands":[1,2,3,4,5,6,7,8,9]},

{"seatid":2, "hands":[1,2,3,4,5,6,7,8,9]},

{"seatid":3, "hands":[1,2,3,4,5,6,7,8,9]},

{"seatid":4, "hands":[1,2,3,4,5,6,7,8,9]}

]

}

起牌胡 2

{

"qibaihus":

[

{

"seatid":1,

"qibaihu":

[

{

"type":2,

"eats":[ {"type":1,"first":1,"eat":1} ],

"hands":[1,2,3,4,5,6,7,8,9]

},

{

"type":3,

"hands":[1,2,3,4,5,6,7,8,9]

}

]

}

]

}

玩家出牌 3

{

"seatid":1,

"card":1,

"hands":[0,0,0,0,1,2,3,4,5,6,7,8,9,10]

}

玩家发牌 4

{

"seatid":1,

"card":1,

"decks\_count":30

}

玩家选择操作 5

{

"seatid":1,

"type":1 // 操作类型

}

玩家操作结果 6

{

"from\_seatid":出牌方座位ID,

"from\_card":2, // 出牌方被吃掉的牌,

"to\_seatid": 吃牌方座位ID,

"to\_eatcard\_add":{"type":牌型,"first":1,"eat":2,"isbu":0}, // 吃牌方牌型

"to\_eatcard\_del":{"type":牌型,"first":1,"eat":2}, // 吃牌方牌型

"to\_hands": [0,0,0,0,0,0,0,0,0,1,2,3,4,5] // 吃牌方当前手上的牌

}

杠上发牌 7

{

"seatid":1,

"cards":[1,2],

"decks\_count":30

}

海底牌 8

{

"seatid":1,

"card":1

}

结算 9

{

“banker\_seatid”:1

"iswin":1, // 是否胜利，表示流局

"fromseatid":1,

"fromcard":1

"zhaniao":

[

{"card":1,"seatid":1,"iszhong":1},

{"card":2,"seatid":2,"iszhong":0}

]

"scores":

[

{"seatid":1, "score":1000, "incsore":3, "types":[] },

{"seatid":2, "score":1000, "incsore":-1},

{"seatid":3, "score":1000, "incsore":-1},

{"seatid":4, "score":1000, "incsore":-1}

]

}

### 心跳

描述：客户端连接保活，客户端应定时发送该报文，服务器收到后原样发回；

协议：{“cmd”:1000}

### 登录请求

描述：登录认证请求；

协议：{“cmd”:1001,”uid”:111,”skey”:”xxxxxxxx”}

服务器会返回 **统一响应码**，表示成功或者失败，详细请参见下面的统一响应码CMD：4001。

### 退出请求

描述：玩家退出游戏请求；

协议：{“cmd”:1002}

服务器会返回 **统一响应码**，表示成功或者失败，详细请参见下面的统一响应码CMD：4001。

### 聊天请求

描述：玩家向同桌内其他玩家广播消息；

协议：

{

“cmd”:1003,

“type”:聊天类型,

“index”:已知聊天编号,

“text”:”自定义聊天内容”,

“chatid”:1

}

### 玩家通知信息变更

描述：玩家通知服务器自己的信息变更；

协议：

{

“cmd”:1004

}

### 玩家信息分享

描述：玩家通知服务器分享信息；

协议：

{

“cmd”:1005,

“data”:分享的字符串

}

### 玩家请求加入房间

描述：玩家在完成登录步骤后，请求进入房间；

协议：

{

“cmd”:1010,

“tid”:桌号

}

成功，返回tableinfo

失败，返回相关错误码，具体请参考“统一响应码”；

### 玩家请求解散房间

描述：玩家在登录房间成功后，可以申请解散所在的房间；

协议：

{

“cmd”:1011

}

### 玩家应答解散房间

描述：玩家对解散房间的应答；

协议：

{

“cmd”:1012,

“ack”:0/1

}

Ack取值：

0 表示拒绝

1 表示同意

### 玩家准备开始游戏

描述：游戏开始前，玩家进入房间，需要发起准备操作；

协议：

{

“cmd”:1013,

“ready”:0/1

}

Ready取值：

0 表示未准备

1 表示准备好

### 玩家出牌

描述：玩家打出一张牌；

协议：

{

“cmd”:1014,

“card”: 1

}

如果玩家当前处于听牌状态，每次拿牌后只能出刚拿的牌。

### 玩家执行某些操作

描述：玩家从可选的提示操作中，选择一个操作；

协议：

{

“cmd”:1015,

“type”:1,

“cards”:[1,2]

}

Type表示操作类型；

### 玩家提求听牌提示

描述：玩家请求听牌提示；

协议：

{

“cmd”:1016

}

### 登录响应兼容协议

描述：为了兼容性，为某些牌本的客户端下发专门的登录响应；

协议：

{

“cmd”:4000,

“code”:0

}

### 统一响应码

描述：对于各种操作错误，服务器采用统一响应码的方式回应；

协议：

{

“cmd”: 4001,

“cmd\_req”:请求的CMD, 0表示服务器主动发起的错误

“code\_res”:错误码

}

Code\_res取值如下：

#define CODE\_SUCCESS 0 // 操作成功

#define CODE\_LOGIN\_AUTH 1 // 认证失败

#define CODE\_LOGIN\_INIT 2 // 玩家初使化失败

#define CODE\_OTHER\_LOGIN 3 // 异地登录被踢出

#define CODE\_NO\_PRIVILEGE 4 // 无操作权限

#define CODE\_GAMEING 5 // 游戏中，禁止操作

#define CODE\_TABLE\_NOTEXIST 6 // 桌子不存在

#define CODE\_SEAT\_NOFREE 7 // 无空闲的位子

#define CODE\_CARDS\_INVALID 8 // 牌型不合法

### 桌子信息响应

描述：玩家进入桌子后，服务器响应桌子信息；

协议：

{

“cmd”: 4002,

“reconnect”:0, // 玩家重连标记

“tid”: 桌子ID,

“myseatid”:1, // 自己的座位ID

“gamestate”:游戏状态, // 0 未开始,1 游戏中,2结算

“owner\_seatid”:房主座位号,

“total\_round”:10, // 总局数

“zhaniao\_count”:2, // 扎鸟数，取值0-6

“bankerwin\_score”:1, // 庄闲算分，1表示计算，0表示不计数

“play\_round”:1,

“decks\_count”:30, // 底牌数

“banker\_seatid”:1, // 庄家座位号

“sendcard\_seatid”:1, // 当前发牌方座位号

“sendcard\_card”:1, // 当前发牌方发的牌

“putcard\_seatid”:1, // 当前出牌方座位号

“putcard\_card”:1, // 当前出牌方打出的牌

“players”:

[

{

“uid”:玩家ID,

“seatid”:0,

“name”:xxx,

“clientip”:”1.1.1.1”, // 玩家外网IP

“offline”:1,

“ready”:0,

“score”:1000,

“hands”:{ 0,0,0,0,0,1,2,3,4,5,6,7,8,9},

“eats”:[{“type”:1,”first”:1,”eat”:2}],

“outcards”:[1,2,3,4,5,6],

},

]

}

### 服务器广播玩家进入房间

描述：玩家进入房间，服务器向其他玩家广播；

协议：

{

“cmd”: 4003,

“player”:

{

“uid”:玩家ID,

“seatid”:座位ID,

“name”:玩家名称，

……

}

}

说明：客户端收到该消息后，初使化玩家，如果之前只是离线，标识为上线；

### 服务器广播玩家退出房间

描述：玩家掉线、或者主动退出房间，服务器向其他玩家广播；

协议：

{

“cmd”:4004,

“uid”:玩家ID,

“seatid”:座位ID,

“binded”:0/1

}

说明：客户端收到本消息后，判断binded值，为1不能完全释放玩家，只能标示为离线；

### 聊天广播

描述：服务器广播聊天；

协议：

{

“cmd”:4005,

“uid”:玩家ID,

“seatid”:座位ID,

“type”:聊天类型,

“index”:已知聊天编号,

“text”:”自定义聊天内容”,

“chatid”:1

}

### 服务器广播玩家信息变更

描述：服务器广播玩家信息更新；

协议：

{

“cmd”:4006,

“uid”:玩家ID,

“seatid”:座位ID，

“money” :玩家最新金币，

“rmb”:钻石

}

### 服务器广播玩家信息分享

描述：服务器广播玩家信息分享；

协议：

{

“cmd”:4007,

“seatid”:座位ID,

“data”:分享的字符串

}

### 服务器询问解散房间

描述：玩家请求解散房间，服务器广播询问；

协议：

{

“cmd”:4010,

“sponsor\_seatid”: 1, // 发起者座位

“expire”:300, // 解散房间的有效期

“acks”:

[

{“seatid”:2, ”ack”:1},

{“seatid”:3, ”ack”:1},

{“seatid”:4, ”ack”:1}

]

}

### 服务器广播解散房间应答

描述：玩家对解散房间做出应答后，服务器向其他玩家广播；

协议：

{

“cmd”:4011,

“seatid”:座位ID,

“ack”:0/1

}

Ack取值：

0 表示拒绝

1 表示同意

### 服务器广播房间解散结果

描述：服务器广播房间解散结果；

协议：

{

“cmd”:4012,

“result”:1 // 1 表示解散成功，0表示解散失败；

}

### 服务器广播玩家准备

描述：玩家进入房间，点击准备操作后，服务器通知其他玩家；

协议：

{

“cmd”:4013,

“seatid”:座位ID,

“ready”:0/1

}

Ready取值：

0 表示未准备

1 表示准备好

### ~~服务器广播游戏开始~~

~~描述：游戏开始倒计时；~~

~~协议：~~

~~{~~

~~“cmd”:4014,~~

~~“time”:3~~

~~}~~

### 服务器广播游戏状态变更

描述：游戏状态；

协议：

{

“cmd”:4015,

“gamestate”:0

}

Gamestate取值如下：

1. 未开始
2. 游戏中
3. 结算

### ~~服务器广播当前庄家~~

~~描述：每局开始前，服务器向所有玩家广播庄家；~~

~~协议：~~

~~{~~

~~“cmd”:4016,~~

~~“seatid”:座位ID,~~

~~“select\_type”:0~~

~~}~~

~~Select\_type 表示选庄方式，取值如下：~~

~~0 随机选庄~~

1. ~~接上局轮庄~~

### 每局开始，服务器广播庄家、起牌

描述：服务器向每个玩家单独发起牌；

协议：

{

“cmd”:4017,

“play\_round”:1,

“banker\_seatid”:座位ID,

“banker\_type”:0,

“decks\_count”:55,

“player\_cards”:

[

{“seatid”:1, “cards”:[1,2,3,4,5,6,7,8,9,11,12,13,14]}, // 对自己发明牌

{“seatid”:2, “cards”:[0,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1]}, // 对其他玩家发暗牌

{“seatid”:3, “cards”:[0,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1]},

{“seatid”:4, “cards”:[0,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1]}

]

}

Round表示回合数；

Select\_type 表示选庄方式，取值如下：

0 随机选庄

1. 接上局轮庄

### 服务器广播起牌胡

描述：完成起牌后，若有玩家可以起手胡，服务器则广播胡牌；

协议：

{

“cmd”:4018,

“seatid”:座位ID,

“qibaihu”:

[

{

“type”:2,

“eats”: [{“type”:1,”first”:1,”eat”:1}],

“hands”:{ 0,0,0,0,0,-1,-1,-1,-1,-1,-1,-1,-1,-1},

},

{

“type”:3,

“hands”:{ 0,0,0,0,0,1,2,3,4,5,6,7,8,9},

}

]

}

Type表示胡牌类型

Cards表示玩家起牌胡部分牌型

### 服务器广播当前出牌方

描述：服务器广播当前归谁出牌；

协议：

{

“cmd”:4019,

“seatid”:座位ID

}

### 服务器提醒玩家出牌

描述：轮到玩家出牌后，取务器提醒玩家出牌；

协议：

{

“cmd”:4020

}

### 服务器广播玩家出牌

描述：玩家打出牌后，服务器广播；

协议：

{

“cmd”:4021,

“seatid”:座位ID,

“card”:1,

“hands”:[0,0,0,0,1,2,3,4,5,6,7,8,9,10]

}

Hands表示手牌，用于玩家出于牌后手牌修正，0表示填位，-1表示暗牌；

### 服务器广播向玩家发牌

描述：服务器广播向玩家发牌；

协议：

{

“cmd”:4022,

“seatid”:座位ID,

“card”:1,

“decks\_count”:30,

}

对发牌方来说，这张牌才可见，不可见发暗牌-1；

### 服务器提醒玩家执行某些操作

描述：需要玩家操作时，如吃、碰、杠、补、胡，服务器下发提醒；

协议：

{

“cmd”:4023,

“actions”:

[

{“type”:1,”options”:[{“cards”:[1,2]}, {“cards”:[4,5]}]},

{“type”:2,”options”:[{“cards”:[1,2]}, {“cards”:[4,5]}]}

]

}

### 服务器广播当前为海底牌

描述：如果当前是海底牌，服务器主动发起提醒；

协议：

{

“cmd”:4024

}

### 服务器广播玩家操作

描述：玩家选择某个操作，服务器广播；

协议：

{

“cmd”:4025

“from\_seatid”:出牌方座位ID,

“from\_card”:2, // 出牌方被吃掉的牌,

“to\_seatid”: 吃牌方座位ID,

“to\_eatcard\_add”:{“type”:牌型,”first”:1,”eat”:2,”isbu”:0}, // 吃牌方牌型

“to\_eatcard\_del”:{“type”:牌型,”first”:1,”eat”:2}, // 吃牌方牌型

“to\_hands”: [0,0,0,0,0,0,0,0,0,-1,-1,-1,-1,-1] // 吃牌方当前手上的牌，-1表示暗牌

}

如果from\_seatid为-1，表示是玩家自操作，只有吃牌方的牌发生变化；

To\_eatcard\_add下的isbu字段，当牌型为杠的时候，需要区分是否为补操作，值为1表示补，碰和吃牌型无此字段；

### 服务器广播全部翻牌

描述：每局结算时，广播所有玩家的牌型信息；

协议：

{

“cmd”:4026,

“huseatid”:[1,2], // 胡牌座位列表 为空或者不存在 表示流局

“iszimo”:1/0, // 1表示自摸，0表示放炮

“cards”:

[

{“seatid”:1, “hands”:[0,0,0,0,0,0,0,0,1,2,3,4,5,6], “eats”:[{“type”:1,”first”:1,”eat”:2}], “types”:[] },

{“seatid”:2, “hands”:[ 0,0,0,0,0,0,0,0,1,2,3,4,5,6], “eats”:[{“type”:1,”first”:1,”eat”:2}], “types”:[] },

{“seatid”:3, “hands”:[ 0,0,0,0,0,0,0,0,1,2,3,4,5,6], “eats”:[{“type”:1,”first”:1,”eat”:2}], “types”:[] },

{“seatid”:4, “hands”:[ 0,0,0,0,0,0,0,0,1,2,3,4,5,6], “eats”:[{“type”:1,”first”:1,”eat”:2}], “types”:[] }

]

}

### 服务器广播结算

描述：每局结算时，广播所有玩家的输赢信息；

协议：

{

“cmd”:4027,

“banker\_seatid”: 庄家座位号,

“iswin”:1, // 是否胜利，0表示流局

“fromseatid”:1,

“fromcard”:1,

“scores”:

[

{“seatid”:1, “score”:1000, “incsore”: 3, “types”:[] },

{“seatid”:2, “score”:1000, “incsore”: -1},

{“seatid”:3, “score”:1000, “incsore”: -1},

{“seatid”:4, “score”:1000, “incsore”:-1}

]

}

Fromseatid表示放炮方座位ID

Fromcard表示放炮方打出的牌

Scores下的type表示胡牌类型

### 服务器广播房间次数用玩

描述：房间次数用完，服务器广播房间即将收回；

协议：

{

“cmd”:4028

}

### 服务器广播杠上发牌

描述：服务器广播向玩家发杠上牌；

协议：

{

“cmd”:4029,

“seatid”:座位ID,

“cards”:[1,2],

“decks\_count”:30

}

Cards对所有玩家明牌，如不够2张，用0表示无牌；

### 服务器广播扎鸟

描述：胡牌后，服务器广播扎鸟结果；

协议：

{

“cmd”:4030,

“zhaniao”:

[

{“card”:1,”seatid”:1,”iszhong”,1},

{“card”:2,”seatid”:2,”iszhong”,0},

{“card”:3,”seatid”:2,”iszhong”,0},

{“card”:4,”seatid”:2,”iszhong”,0},

{“card”:5,”seatid”:2,”iszhong”,0},

{“card”:6,”seatid”:2,”iszhong”,0}

]

}

Card表示扎鸟牌

Seatid表示该牌中的座位

Iszhong表示该座位是否中鸟

### 服务器广播海底牌要不要询问

描述：广播正在进行玩家海底牌询问；

协议：

{

“cmd”:4031,

“seatid”:座位ID

}

### 服务器广播海底牌要不要回答

描述：广播玩家要不要回答；

协议：

{

“cmd”:4032,

“seatid”:座位ID,

“ack”:0/1, // 1表示要,0表示不要

“force”:0/1 // 1表示强制玩家要海底，0表示不强

}

### 服务器广播发海底牌

描述：服务器广播发海底牌；

协议：

{

“cmd”:4033,

“seatid”:座位ID,

“card”:1

}

### 服务器广播总结算战绩

描述：8局打完或者解散房间，服务器广播总结算成绩；

协议：

{

“cmd”:4034,

“scores”:

[

{“seatid”:1, “score “:100,

"paocount":

[

{"paotype":1, "count":1},

{"paotype":2, "count":1}

]

},

{“seatid”:2, “score “:200,

"paocount":

[

{"paotype":1, "count":1},

{"paotype":2, "count":1}

]

},

{“seatid”:3, “score “:0,

"paocount":

[

{"paotype":1, "count":1},

{"paotype":2, "count":1}

]

},

{“seatid”:4, “score “:-300,

"paocount":

[

{"paotype":1, "count":1},

{"paotype":2, "count":1}

]

}

]

}

### 服务器通知玩家听牌

描述：玩家听牌后，服务器通知玩家听牌；

协议：

{

“cmd”:4035,

“seatid”:座位ID

}

### 服务器响应玩家听牌提示

描述：

协议：

{

“cmd”:4036,

“seatid”:座位ID,

“tingcards”:[1,2,3]

}