

水浒传

协议文件名

游戏通用结构

房间信息

玩家信息

debug玩家信息

滚轮中奖线信息

滚轮结算图标信息

结算图标信息

结算玩家信息

协议列表

检测游戏状态 //重连之后判断是否在桌子中

进入桌子（暂时不使用，游戏中使用通用进房间逻辑）

离开桌子（暂时不使用，游戏中使用通用出房间逻辑）

开始游戏（滚轮滚动）

请求一次小玛丽游戏

请求一次比倍游戏

获取游戏信息（目前客户端走配置，没有使用该协议）

桌子进玩家通知

桌子退玩家通知

玩家钱变化

广播玩家属性变更（例如金币变化，包括充值，任务等）

请求补充试玩金币

请求桌子内场景信息

请求结算（收分）

debug前端控制 获取房间信息

debug前端控制 执行杀分

协议ID枚举

水浒传

协议文件名

game_shuihz_protocol.proto

game_shuihz_def.proto

游戏通用结构

房间信息

msg_room_info

参数名	类型	描述	参数ID	默认值
roomid	int32	房间id	1	

玩家信息

msg_player_info

参数名	类型	描述	参数ID	默认值
player_id	int32	玩家ID	1	
player_nickname	string	昵称	2	
player_head_frame	int32	头像框	3	
player_head_custom	string	玩家头像	4	
player_gold	int64	玩家金币	5	
player_sex	int32	玩家性别	6	
player_vip_lv	int32	Vip等级	7	
seat	int32	座位号	8	

debug玩家信息

room_player

参数名	类型	描述	参数ID	默认值
player_id	int32	玩家ID	1	
gold	int64	玩家金币	2	
bet	int64	玩家下注	3	
profit_today	int64	当日获利	4	

滚轮中奖线信息

LineInfo

参数名	类型	描述	参数ID	默认值
lineId	int32	线ID 1-9，指向配置对应索引	1	
lightFlag	int32	左右灯标记 0 左灯亮 1 右灯亮 2 左右灯全亮	2	
leftLinkCount	int32	左边是几连 3~5，如果是5的话，则右边不统计	3	
rightLinkCount	int32	右边是几连	4	
leftIcon	int32	左边Icon 非0表示图标ID，否则表示没有	5	
rightIcon	int32	右边Icon 非0表示图标ID，否则表示没有	6	

滚轮结算图标信息

IconInfo

参数名	类型	描述	参数ID	默认值
iconId	int32	图标ID	1	
rate	int32	对应中奖倍率	2	

结算图标信息

GameInfo

参数名	类型	描述	参数ID	默认值
bankerSeat	int32	庄家所在座位	1	
smallBlindSeat	int32	小盲注所在座位	2	
bigBlindSeat	int32	大盲注所在座位	3	

结算玩家信息

BalancePlayerInfo

参数名	类型	描述	参数ID	默认值
playerId	int32	玩家ID	1	
seat	int32	玩家所在座位号,	2	
backPoker	PokerInfo[]	玩家底牌	3	
selPoker	PokerInfo[]	最终的5张牌（目前没有使用该字段）	4	
pokerType	int64	最终5张牌的牌型 EPokerType	5	
winReward	int32	赢得的奖励	6	
isAbandon	bool	是否弃牌	7	
isWinner	bool	是否赢家	8	
isShowPoker	bool	是否亮牌	9	
rank	string	排名	10	

协议列表

检测游戏状态 //重连之后判断是否在桌子中

说明：用于断线重连，登录游戏后发送，如果返回在房间中，则走重连逻辑，直接进房间

前端请求协议:packetc2l_check_state

参数名	类型	描述	参数ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_c2l_check_state10001

后端返回协议:packetc2l_check_state_result

参数名	类型	描述	参数ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_l2c_check_state_result15001
room_id	int32	大于0表示正在游戏中 0:不在游戏	2	

进入桌子（暂时不使用，游戏中使用通用进房间逻辑）

说明：暂时不使用，游戏中使用通用进房间逻辑

前端请求协议:packetc2l_enter_game

参数名	类型	描述	参数ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_c2l_enter_game 10002
room_id	int32	房间ID	2	

后端返回协议:packetc2l_enter_game_result

参数名	类型	描述	参数ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_l2c_enter_game_result 15002
result	int32	结果	2	1.成功 非1.失败
room_id	int32	房间ID		
player_list	msg_player_info	玩家列表		

离开桌子（暂时不使用，游戏中使用通用出房间逻辑）

说明：暂时不使用，游戏中使用通用出房间逻辑

前端请求协议:packetc2l_leave_game

参数名	类型	描述	参数ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_c2l_leave_game 10003
room_id	int32	房间ID		

后端返回协议:packetc2l_leave_game_result

参数名	类型	描述	参数ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_l2c_leave_game_result 15003
result	int32	房间列表	2	1.成功 非1.失败
playerGold	int32	玩家金币	3	

开始游戏（滚轮滚动）

说明：开始游戏，点击按钮后调用，滚轮开始滚动，服务端返回滚轮结果

前端请求协议:packetc2l_star_game

参数名	类型	描述	参数ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_c2l_star_game 10004
rateIndex	int32	下注倍率 配置表中读取	2	

后端返回协议:packetl2c_req_declare_result

参数名	类型	描述	参数ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_l2c_star_game_result 15004
moneyType	int32	货币类型	2	1 金币类型，暂时没有其他类型
imageArr	string	15个滚轮icon图标，1~9，用半角，分隔	3	
infoList	LineInfo[]	滚轮中奖线信息	4	
winMoney	int32	赢得的钱	5	
deductMoney	int32	下注扣除的钱	6	
result	int32	结果	7	1.成功 非1.失败
curMoney	int32	当前的总钱数	8	
bonusGameCount	int32	可玩小玛丽游戏次数	9	
fullType	int32	全盘类型 0 没有全盘; 1 相同id的全盘; 2 武器全盘; 3 人物全盘	10	
fullId	int32	1 相同icon id的全盘 情况下的id	11	
icons	IconInfo[]	结算图标倍率信息（中奖图标）	12	

请求一次小玛丽游戏

说明：滚轮中奖后，如果中奖线中，有水浒传的线（从左或右开始有连续的3个龙，从中间开始不算），会赠送一定次数的小玛丽游戏，当次数没有小号完的时候，可以做次请求。小玛丽游戏中，只要外圈不转到exit就不扣次数，当次数扣完后，退出小玛丽

前端请求协议:packetc2l_req_bonus_game

参数名	类型	描述	参数ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_c2l_req_bonus_game 10005

后端返回协议:packetl2c_req_bonus_game_result

参数名	类型	描述	参 数 ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_l2c_req_bonus_game_result 15005
innerIconArr	string	内圈4个图标(从左到右) 与滚轮图标一样的格式，逗号分隔	2	
outerIcon	int32	外圈转到的图标结果 0-8 0表示要转到Exit,消耗一次次数	3	
winMoney	int32	本次奖励	4	
result	int32	请求结果	5	1.成功 非1.失败
showIcon	string	内圈高亮的位置，包括连续3个以上，以及和外圈一样的图标的位置用1,0,1,1这样代表4个位置是否高亮，1亮，0不亮，后端使用二进制位表示的，由于前端不方便转换，所以传字符串	6	

请求一次比倍游戏

说明：在滚轮和小玛丽状态之后，只要玩家是赢钱的，就可以进行比倍请求。此状态可以用收分请求中断，进入下一局游戏。或者输一次，也会自动中断。

前端请求协议:packetc2l_req_double_game

参数名	类型	描述	参数ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_c2l_req_double_game 10006
betpos	int32	押注位置： 0 小; 1 和; 2 大;	2	
betgold	int32	押注金额（校验用）	3	
betid	int32	押注轮数（校验用）	4	

后端返回协议:packetl2c_req_declare_result

参数名	类型	描述	参数ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_l2c_req_double_game_result 15006
dice1	int32	骰子1的 值	2	
dice2	int32	骰子2	3	
winMoney	int32	本次奖励	4	
result	int32	请求结果	5	1.请求成功 非1.请求失败

获取游戏信息（目前客户端走配置，没有使用该协议）

说明：获取房间列表

前端请求协议:packetc2l_get_room_info

参数名	类型	描述	参数ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_c2l_get_room_info 10007

后端返回协议:packetl2c_get_room_info_result

参数名	类型	描述	参数ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_l2c_get_room_info_result15011
rooms	msg_room_info[]	房间列表		

桌子进玩家通知

说明：当其他玩家进入时服务器通知

后端返回协议:packetl2c_enter_player_info

参数名	类型	描述	参数ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_l2c_enter_player_info 15008
player_info	msg_player_info	玩家信息	2	

桌子退玩家通知

说明：其他玩家退出时服务器通知

后端返回协议:packetl2c_leave_player_info

参数名	类型	描述	参数ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_l2c_leave_player_info 15009
player_id	int32	玩家id	2	

玩家钱变化

说明：其他玩家发生金币变化时服务端通知

现在每个房间都有配置进入的携带金币上限，设置自动补充后，每局结束后会自动补充金币

后端返回协议:packetl2c_player_change_money

参数名	类型	描述	参 数 ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_l2c_player_change_money 15010
player_id	int32	玩家id	2	
cur_money	int64	当前金币	3	1.成功 非1.失败
change_money	int64	变化金币	4	
bonus_state	int32	//触发此次金币变化的原因 小玛丽状态值 0.非小玛丽 1.进小玛丽 2.出小玛丽	5	

广播玩家属性变更（例如金币变化，包括充值，任务等）

说明：玩家属性发生变化时服务端通知

后端返回协议:packetl2c_bc_change_attr

参数名	类型	描述	参数ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_l2c_bc_change_attr 15011
player_id	int32	玩家id	2	
item_type	int32	//1.金币 非1.其他（金币以外的类型暂时没有用到）	3	1.成功 非1.失败
change_value	int64	变动值	4	

请求补充试玩金币

说明：在试玩场游戏时，试玩币不足的情况下可调用，会补充试玩币到指定值（配置）

前端请求协议:packetc2l_supply_chip

参数名	类型	描述	参数ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_c2l_supply_chip 10021

后端返回协议:packetl2c_supply_chip_result

参数名	类型	描述	参数ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_l2c_supply_chip_result 15021
result	msg_type_def.e_msg_result_def	结果	2	1.成功 非1.失败
gold	int64	补充到的值		

错误码

```
enum e_msg_result_def
{
    e_rmt_unknow = 0; //未知错误
    e_rmt_success = 1; //成功
    e_rmt_fail = 2; //失败
    //其他
}
```

####

请求桌子内场景信息

说明：进入游戏后调用，获取场景信息，用以客户端初始化展示

前端请求协议:packetc2l_get_room_scene_info

参数名	类型	描述	参数ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_c2l_get_room_scene_info 10022

后端返回协议:packetl2c_get_room_scene_info_result

参数名	类型	描述	参数ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_l2c_get_room_scene_info 15022
player_list	msg_player_info[]	玩家列表	2	

请求结算（收分）

说明：在游戏进到可选择比倍阶段的时候可调用，用来中断比倍，直接收分，准备进入下一局

前端请求协议:packetc2l_settlement

参数名	类型	描述	参数ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_c2l_settlement 10023

后端返回协议:packetl2c_settlement_result

参数名	类型	描述	参数ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_l2c_settlement_result 15023
result	int32	结果	2	1.成功 非1.失败
gold	int64	当前钱（无论结果是否成功，都返回）		

debug前端控制 获取房间信息

说明：有debug的权限可以调用，用来展示debug界面信息

前端请求协议:packetl2c_gm_get_room_info

参数名	类型	描述	参数ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_c2l_gm_get_room_info 10101

后端返回协议:packetl2c_settlement_result

参数名	类型	描述	参数ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_l2c_gm_get_room_info_result 15101
result	int32	结果	2	1.成功 非1.失败
killcount	int32	剩余GM控制杀分次数（人次）	3	
players	room_player	玩家列表	4	

debug前端控制 执行杀分

说明：拥有debug权限的账户可调用，用来执行房间杀分次数

前端请求协议:packetl2c_gm_get_room_info

参数名	类型	描述	参数ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_c2l_gm_do_kill 10102
killcount	int32	设置杀分次数（人次）	2	

后端返回协议:packetl2c_settlement_result

参数名	类型	描述	参数ID	默认值
packet_id	e_server_msg_type	协议Id	1	e_mst_l2c_gm_do_kill_result 15102
result	int32	结果	2	1.成功 非1.失败

协议ID枚举

```
enum e_server_msg_type
{
    //客户端到服务端 -----
    e_mst_start_c2l = 10000;
    e_mst_c2l_check_state = 10001; //是否在游戏
    e_mst_c2l_enter_game = 10002; //进入游戏
    e_mst_c2l_leave_game = 10003; //离开游戏
    e_mst_c2l_star_game = 10004; //开始游戏
    e_mst_c2l_req_bonus_game = 10005; // 请求一次小玛丽游戏
    e_mst_c2l_req_double_game = 10006; //请求一次比倍游戏
    e_mst_c2l_get_room_info = 10007; //获取房间列表

    e_mst_c2l_supply_chip = 10021; //补充试玩币
    e_mst_c2l_get_room_scene_info = 10022; //获取房间信息
    e_mst_c2l_settlement = 10023; //主动结算

    e_mst_c2l_gm_get_room_info = 10101; //gm请求房间信息
    e_mst_c2l_gm_do_kill = 10102; //gm执行杀分

    //服务端到客户端 -----
    e_mst_start_l2c = 15000;
    e_mst_l2c_check_state_result = 15001; //是否在游戏 - 回执
    e_mst_l2c_enter_game_result = 15002; //进入游戏 - 回执
    e_mst_l2c_leave_game_result = 15003; //离开游戏 - 回执
    e_mst_l2c_star_game_result = 15004; //开始游戏 - 回执
    e_mst_l2c_req_bonus_game_result = 15005; // 一次小玛丽游戏结果
    e_mst_l2c_req_double_game_result = 15006; //一次比倍游戏结果
    e_mst_l2c_get_room_info_result = 15007; //获取房间列表结果

    e_mst_l2c_enter_player_info = 15008; //玩家进入房间
    e_mst_l2c_leave_player_info = 15009; //玩家离开房间
    e_mst_l2c_player_change_money = 15010; //玩家(robot)钱变化
    e_mst_l2c_bc_change_attr = 15011; //玩家属性变化
}
```

```
e_mst_l2c_supply_chip_result = 15021;      //补充试玩币返回
e_mst_l2c_get_room_scene_info = 15022;     //获取房间信息返回
e_mst_l2c_settlement_result = 15023;       //主动结算返回

e_mst_l2c_gm_get_room_info_result = 15101; //gm请求房间信息返回
e_mst_l2c_gm_do_kill_result = 15102;      //gm执行杀分返回

e_mst_clend_index = 20000;
}
```