**1、人物邮件**

人物邮件为人物下has\_email（和gold等字段同级，和gold等一样实时更新）

public bool has\_email；//bool值，true表示有邮件，具体邮件在点击邮件后向服务器获取

**2、获取人物信息协议修改**

NetworkMessage message = NetworkMessage.Create((ushort)Opcodes.Server\_Character\_Info, 20);

message.writeUInt64(this.guid);

message.writeString(this.name);//账号

message.writeString(this.otherName);//昵称

message.writeString(this.headId);//头像

message.writeInt64(this.card);//房卡

message.writeInt64(this.diamond);//钻石

message.writeInt64(this.gold);//金币

message.writeString(DataManager.Instance.weixinShareUrl);//分享链接

if (this.roomer.enterRoomTypeCodeInfo != null)

{

if (RoomManager.Instance.getRoom(this.roomer.enterRoomTypeCodeInfo.Item2) != null)

{

message.writeUInt32(this.roomer.enterRoomTypeCodeInfo.Item2);//房间号

}

else

{

this.roomer.createRoomCodes.Remove(this.roomer.enterRoomTypeCodeInfo.Item2);

this.roomer.enterRoomTypeCodeInfo = null;

message.writeUInt32(0);

}

}

else message.writeUInt32(0);

message.writeUInt8((byte)sex);

message.writeString(DataManager.Instance.soundUploadUrl);

message.writeString(DataManager.Instance.soundDownloadUrl);

message.writeBool(isDaiLi);//代理

PaoMaDingManager.Instance.packMessage(message);//跑马灯

message.writeString(DataManager.Instance.ShowMessage);//公告

message.writeString(DataManager.Instance.serverVersionCode);//服务器版本号

message.writeString(DataManager.Instance.serverVersionAndroidUrl);//安卓更新地址

message.writeString(DataManager.Instance.serverVersionIosUrl);//ios更新地址

message.writeString(DataManager.Instance.serverVersionUpdateInfo);//更新信息

message.writeBool(DataManager.Instance.loginWithServiceCode);//是否设备号登陆

message.writeUInt32(dayShareRewardCount);//每日分享次数

message.writeUInt64(playerBase.inviteGuid);//邀请人guid

//以上为原码，未改动，以下为改动部分

//发送玩家所有未处理的俱乐部邀请请求

message.writeBool(has\_email);//是否有邮件

message.writeInt32(clubList.Count);//玩家加入的俱乐部数量

foreach(var item in clubList)

{

ClubBase club = ClubManager.Instance.getClub(item);

if(club != null)

{

message.writeBool(true);//是否存在该俱乐部

message.writeUInt32(club\_id);//俱乐部id

message.writeString(club\_name);//俱乐部名称

message.writeInt32(room\_list.Count);//俱乐部房间数}

else

{

message.writeBool(false);//是否存在该俱乐部

} }

**3、查询俱乐部房间列表，点击俱乐部时候看到俱乐部开房信息时使用**

Handler.addPlayerHandler(Opcodes.Client\_Query\_Club\_RoomList, onClientQueryClubRoomList);//查询俱乐部房间列表

NetworkMessage message = NetworkMessage.Create((ushort)Opcodes.Client\_Query\_Club\_RoomList\_Act, 100);

message.writeInt32(room\_list.Count);

foreach(XYGQiPaiRoom item in room\_list.Values)

{

message.writeUInt32(item.codeId);//房间id

message.writeUInt8((byte)item.playType);//玩法类型，普通、包牌

message.writeUInt32(item.getRoomConfig().playerCount);//玩家人数

message.writeUInt32(item.getRoomConfig().gameCount);//局数

message.writeBool(item.getRoomConfig().wuZhaTp);//无炸摊牌

message.writeBool(item.getRoomConfig().faWangTp);//罚王摊牌

message.writeInt32(item.getRoomConfig().jiangMa.Count);//奖码数

foreach (var sub\_item in item.getRoomConfig().jiangMa) message.writeUInt32(sub\_item);//奖码

Player create\_player = Player.GetPlayer(item.creatorGuid);

if (create\_player != null)

{

message.writeString(create\_player.otherName);//开房人名称

message.writeUInt64(create\_player.guid);//开房人guid

}

else

{

message.writeString("无该人物数据");//开房人名称

message.writeUInt64(0);//开房人guid

}

message.writeInt32(item.sitPlayerDict.Count);//在座玩家个数

foreach(var player\_item in item.sitPlayerDict.Values)

{

message.writeUInt8((byte)player\_item.position);//玩家位置

message.writeUInt64(player\_item.guid);//玩家的guid

message.writeString(player\_item.playerBase.otherName);//玩家昵称

message.writeString(player\_item.playerBase.headId);//玩家头像

}

}

**4、俱乐部设置管理员**

Handler.addPlayerHandler(Opcodes.Club\_Set\_Admin, onClubSetAdmin);//俱乐部设置管理员

uint club\_id = message.readUInt32();//俱乐部id

ulong player\_guid = message.readUInt64();//要设置为管理员的玩家id

bool is\_set\_admin = message.readBool();//true为设置为管理员，false为取消管理员

(完成后会返回俱乐部信息，NetworkMessage message = NetworkMessage.Create((ushort)Opcodes.Client\_Query\_Club\_Info\_Ack, 100);)

**5、玩家申请俱乐部消息广播（只有管理员和会长会收到）**

Opcodes.Club\_New\_Application

收到这个协议再消息上打个红点，点击消息按钮后去向俱乐部发Client\_Query\_Club\_Player\_Info协议获取申请人列表

**6、俱乐部申请人列表**

NetworkMessage message = NetworkMessage.Create((ushort)Opcodes.Client\_Query\_Club\_Player\_Info\_Ack, 50);

message.writeInt32(applicationList.Count);//申请人数

foreach (var item in applicationList)

{

message.writeString(item.name);//申请人名字

message.writeUInt64(item.guid);//申请人id

message.writeString(item.head\_ip);//申请人头像

}

**7、俱乐部信息返回**

NetworkMessage message = NetworkMessage.Create((ushort)Opcodes.Client\_Query\_Club\_Info\_Ack, 100);

message.writeUInt32(club\_id);//俱乐部id

message.writeString(club\_name);//俱乐部名称

message.writeString(admin\_name);//俱乐部管理人名称

message.writeUInt64(admin\_guid);//俱乐部管理人id

message.writeString(admin\_tel);//俱乐部管理人电话

message.writeInt64(gold);//俱乐部资金

message.writeUInt8((byte)club\_create\_power);//开房权限，ClubCreatePower

message.writeInt32(PlayerCount);//俱乐部人数

message.writeInt32(playerList.Count);//俱乐部玩家信息

Player sub\_player;

foreach (var item in playerList)

{

message.writeString(item.name);//俱乐部成员名字

message.writeUInt64(item.guid);//俱乐部成员id

message.writeString(item.head\_ip);//俱乐部成员头像

sub\_player = Player.GetPlayer(item.guid);

if (sub\_player != null)

{

message.writeString(sub\_player.sessioner.clientRemoteEndPoint.Substring(0, player.sessioner.clientRemoteEndPoint.LastIndexOf(':'))); //ip地址

message.writeString(sub\_player.address);//地址

message.writeUInt32(sub\_player.play\_count);//玩的次数

}

}

message.writeInt32(admin\_list.Count);//管理员信息

foreach (var item in admin\_list)

{

message.writeString(item.name);//管理员名字

message.writeUInt64(item.guid);//管理员id

message.writeString(item.head\_ip);//管理员头像

}

**8、客户端发送所在地址（前端发给服务器）**

Handler.addPlayerHandler(Opcodes.Client\_Set\_Client\_Address, onClientSetAddress);//设置客户端所在地址

player.address = message.readString();//玩家所在地址