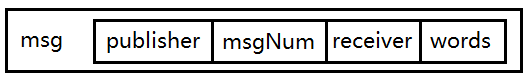
# Client -> Server



1. **msg:**

客户端的包头

1. **publisher**

消息的发送者

在用户登录之前（消息5、6），publisher使用服务器返回的连接名

用户登录周，publisher使用用户的真实名字

1. **msgNum**

消息编号

1：请求好友列表

2：私聊信息

3：群聊信息

4：用户下线

5：用户注册

6：用户登录

7：用户注销

8：发送文件

9：接收文件

1. **words**

消息内容

1：null

2：私聊信息

3：群聊信息

4：null

5：用户注册名

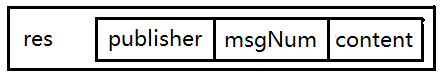
6：null

7：null

8：文件名+文件大小（filePart）

9：服务器文件名+是否接收

# Server -> Client



1. **res:**

服务端的包头

1. **publisher**

消息的发送者

私聊、群聊、转发文件时使用，其他情况设为null

1. **msgNum**

消息编号

0：回复客户端连接名字

1：回复好友列表

2：转发私聊信息

3：转发群聊信息

5：用户注册

6：用户登录

8：接收文件（服务器通知文件发送者开始发送文件）

9：转发文件（服务器通知文件接收者准备接收文件）

1. **words**

消息内容

0：回复客户端的连接名字

1：好友列表（以map<用户名，用户连接名>的形式）

2：私聊信息

3：群聊信息

5：用户注册是否成功

true：成功

false：失败，用户名已存在

6：用户登录是否成功

true: 成功

false: 失败，用户已经登陆

NOUSER：失败，没有注册的用户名

8：文件名

9：服务器文件名+原文件名+大小（filePart）

# 服务器内存中的连接、登录记录

1. **连接表：socketsMap<String, Socket>**

<连接名，Socket(8888)>

主要用来通信——CS之间发送请求

服务器在接收到客户端的连接后，将记录存入该表（S启动时）

1. **文件传输连接表：fileSocketMap<String, Socket>**

<连接名，Socket(9999)>

只有在传输文件时使用

服务器在接收到客户端的连接后，将记录存入该表（S启动时）

1. **在线用户表usersMap<String, String>**

<用户名， 连接名>

用户登录后，将记录存入该表

# 关键调用关系描述

1. **刷新好友列表**

ResDisplayFriendsHandler 受到 res 后，

调用 HomePage 中的 updateFriendList()，

updateFriendList() 中调用ResDisplayFriendsHandler 中的 updateFriendListStr()，

updateFriendListStr() 中调用 HomePage 中的 setFriendListStr()，

之后HomePage 中的updateFriendList() 根据自己的属性friendListStr 刷新列表。