**基于QT和Linux的在线聊天室**

项

目

文

档

**组 号： 五 组**

**组 长： 王陈友**

**小组成员： 王陈友、林兴宇、王 寰、王兵兵、唐新缘、郑焰秋**

**时 间： 2022年12月5日**

**目录**

[**一、项目背景与概述** 1](#_Toc121940127)

[**二、成员分工与描述** 1](#_Toc121940128)

[**三、整体功能描述** 1](#_Toc121940129)

[（1）服务器 1](#_Toc121940130)

[（2）客户机 1](#_Toc121940131)

[**四、总体设计描述** 2](#_Toc121940132)

[（1）服务器设计 2](#_Toc121940133)

[（2）客户机设计 2](#_Toc121940134)

[（3）QT总体设计 2](#_Toc121940135)

[**五、代码实现** 2](#_Toc121940136)

[（1）服务器设计 2](#_Toc121940137)

[（2）客户机设计 4](#_Toc121940138)

[（3）QT界面设计 5](#_Toc121940139)

[**六、成果展示** 6](#_Toc121940140)

[**七、项目总结** 8](#_Toc121940141)

[（1）QT设计总结 8](#_Toc121940142)

[（2）服务器搭建总结 8](#_Toc121940143)

[（3）客户机搭建总结 9](#_Toc121940144)

[（4）项目整体总结 9](#_Toc121940145)

[**八、参考资料** 9](#_Toc121940146)

**一、项目背景与概述**

在学习完Linux网络编程之后，利用多线程之间的的关系，将自己搭建的服务器作为中间站，转发来自各个客户机之间的消息，保证数据传输；与此同时，我们利用QT设计出在线聊天的界面，方便用户使用。在服务器与客户机的设计中，充分考虑到不同操作系统之间的差异性，所以，我们面向Linux有一个专属的客户机，在Windows系统下，利用QT设计出了另一个客户机，可在相同网段内实现不同主机之间的聊天通信。

**二、成员分工与描述**

王陈友：服务器搭建、代码修改、项目文档撰写；

王兵兵：编写QT接收服务器发送的图片代码

林兴宇：QT登录以及代码优化调试

郑焰秋：调试QT与服务器之间的通信，查找问题

唐新缘：QT界面和代码整合

王 寰：调试整合后的程序，提出改进问题

**三、整体功能描述**

（1）服务器

在服务器的搭建中采用了多线程，使得可以接入多个用户，利用系统时间头文件以及函数实现实时的消息传递时间显示，将消息整合后发送到各个客户机。

（2）客户机

利用QT设计，制作了一个简单的聊天界面，同时在QT中编写一个专属客户机，同Linux客户机原理相同，使得可以在同一个网段内不同操作系统的主机也可以进行相互通信，同时采用多线程的优势实现了客户机与服务器之间全双工通信，使得客户机与客户机之间进行间接通信。

**四、总体设计描述**

（1）服务器设计

服务器只作为一个中间站：通过客户端的连接申请，不断的在服务器中创建子线程，并广播新增成员消息，当服务器收到来自客户端的通信消息时，第一时间转发至所有客户端。

（2）客户机设计

在一个客户端中必须存在两个线程，一个用于接收，一个用于发送，当客户端接收到来自服务器的广播消息时，打印出来即可；也可以主动向服务器发送消息，等待服务器转发至别的客户端

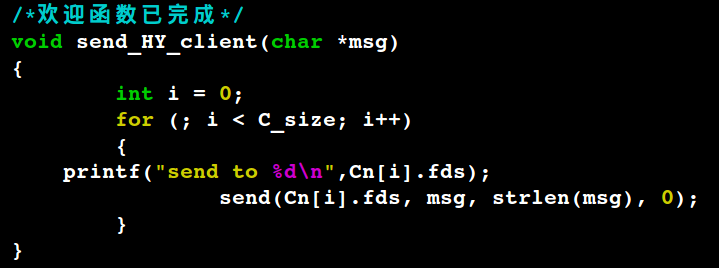
（3）QT总体设计

首先是设计简易的聊天界面，通过槽函数、单击事件出发信号，完成与服务器之间的连接，设置了发送按钮，以及显示聊天信息的文本框，用于接收服务器转发过来的消息，实现局域网多人聊天室的功能。

**五、代码实现**

（1）服务器设计

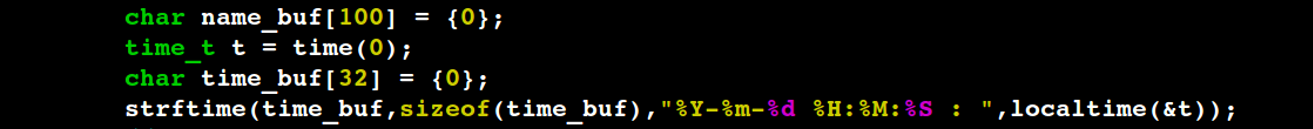
当客户机连接服务器之后，表示进入聊天室，系统发出欢迎提示



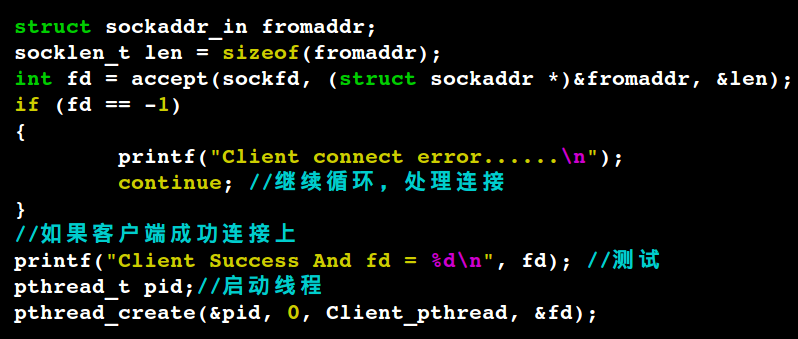
服务器转发来自客户机消息的具体实现



其中以下语句为获取系统时间，并进行格式化处理放进数组备用

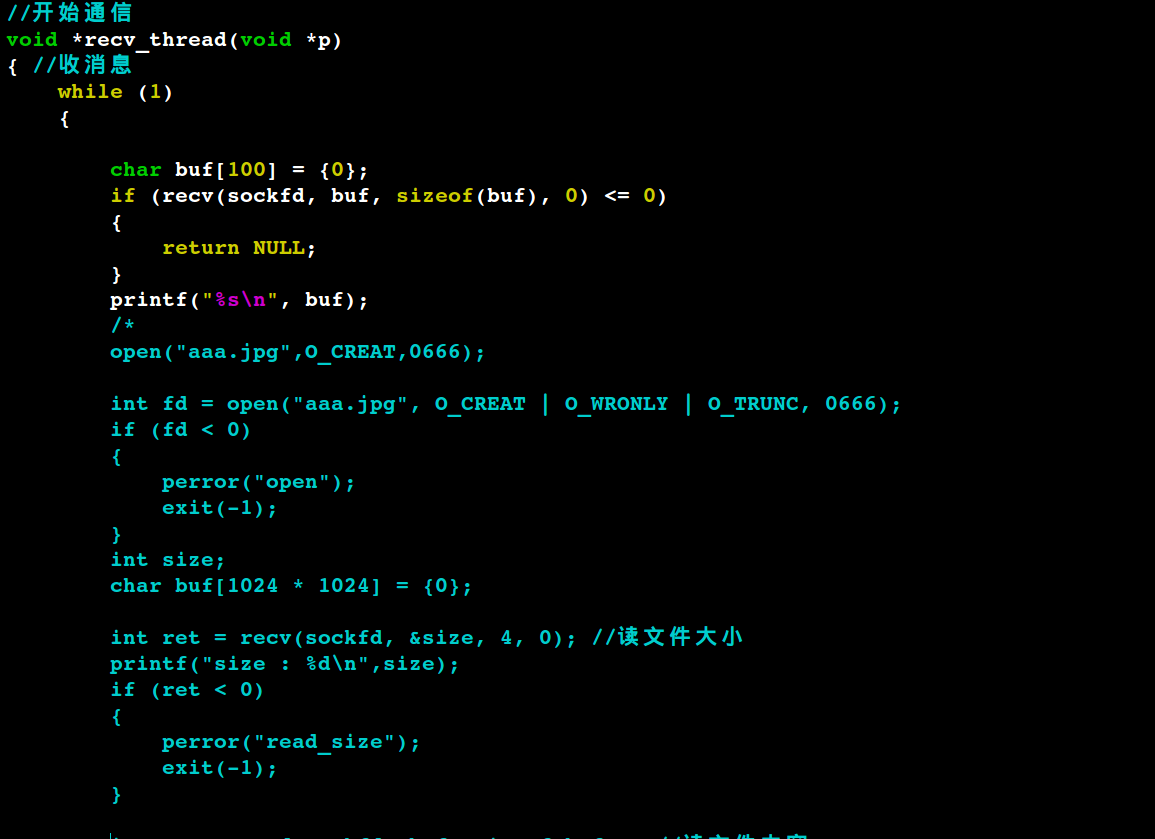


下面是客户端连接服务器之后，在服务器中打印输出，用作功能调试

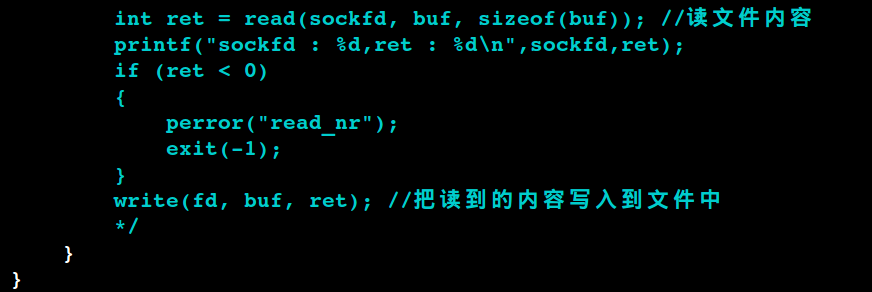


（2）客户机设计

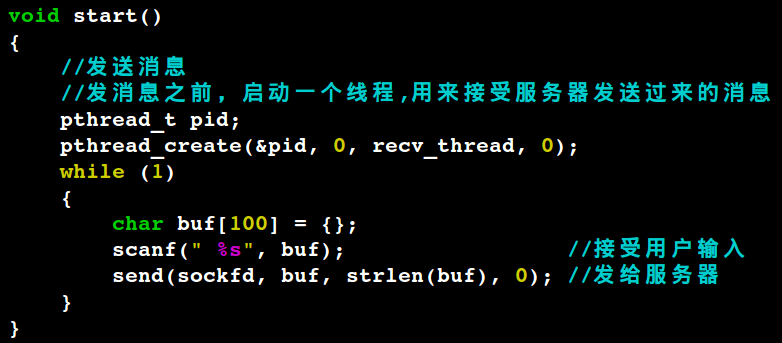
接收消息由很短的代码便可完成，如图所示



图中注释部分，为图片接收以及转发代码，但是由于时间有限，故该功能没有完美实现，只做到了客户机到客户机之间点对点连接的传送，当应用到服务器之后，有些BUG，所以废弃了图片接收与转发功能

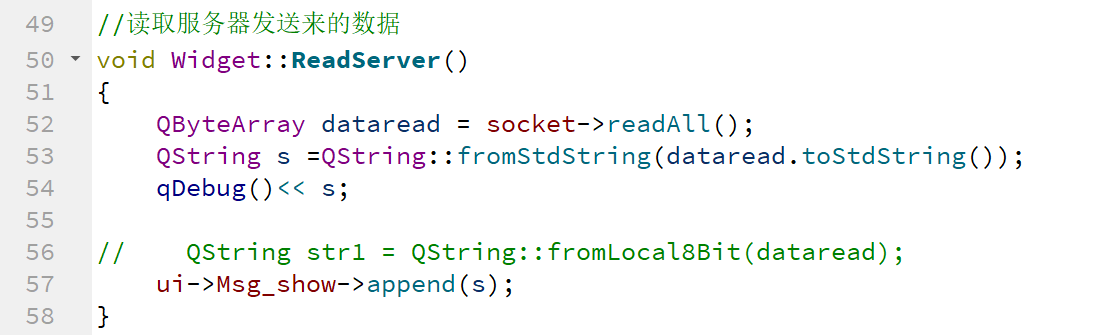


以下代码表示将用户输入的消息信息发送给服务器，由服务器再转发到群聊中

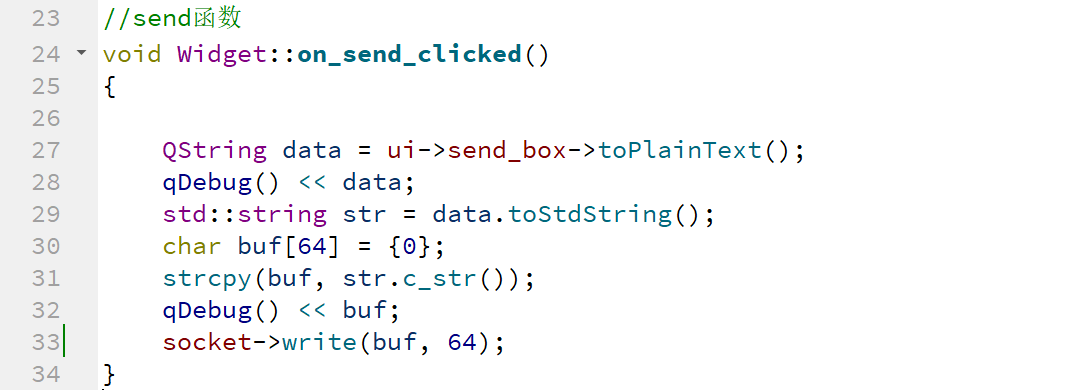


（3）QT界面设计

QT界面上接收服务器传递来的消息



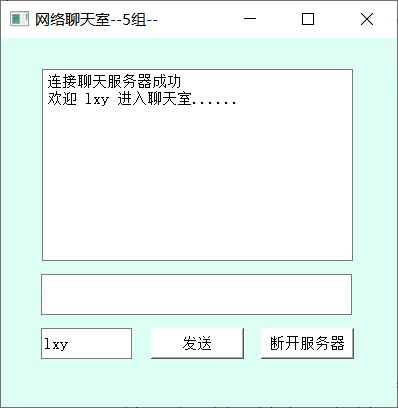
发送消息到服务器中，其中解决了不同操作系统之间传递中文字符出现乱码问题，乱码原因为对字符的编码方式不同，其中灵活运用QT中的toStdString()函数，再利用C++的string类型，巧妙处理中文数据消息

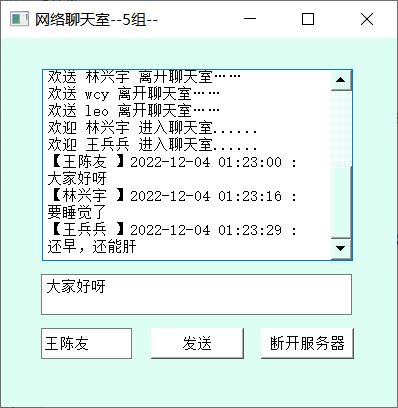


这里仅仅展示了单机按钮连接服务器的代码，而服务器的IP地址以及端口号我们嵌入到了代码当中，当我们连接成功之后，按钮变为断开连接，再次单机之后退出在线聊天室，或者关闭窗口后也会自动与服务器断开连接退出聊天室

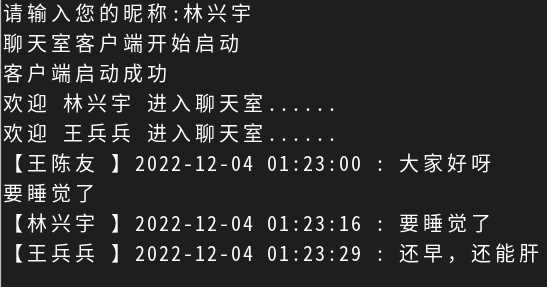


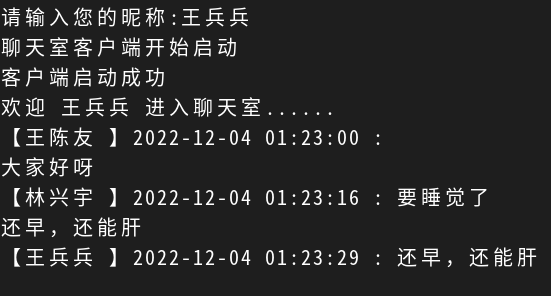
**六、成果展示**



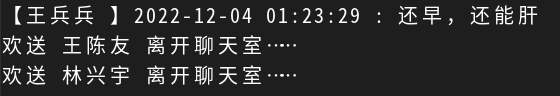


以上为QT端客户机聊天界面显示，下面为Linux终端中客户机演示





当用户断开连接后，服务器广播群消息，通知大家，谁退出了聊天室



**七、项目总结**

（1）QT设计总结

QT界面设计相对于其它来说还是比较简单，但是困难在于槽函数的绑定，以及事件信号的出发以及于Linux接口上的对接，其次是对于中文字符数据的接收。由于字符编码方式的不同，我们需要去转换成能够显示出来的字符，在代码实现中较为困难。

（2）服务器搭建总结

在搭建服务器之前，首先要明白服务器搭建的流程，如何去创建绑定套接字，怎么判断用户有没有接入成功，当客户机连接成功后怎么去广播消息；最重要的时候，当客户机连接成功后，如何去接收客户机发送过来的数据，再去做一个转发功能，以及怎么去实现显示实时的信息发送时间。

（3）客户机搭建总结

在客户机中，利用一个buf数组接收服务器转发过来的数据，然后将其打印输出，这部分内容相对比较简单；再加上运用多线程技术，实现了客户机与服务器之间的全双工通信。难点在于如何去接收服务器发送过来的图片消息，怎么讲图片消息显示出来，由于时间紧缺以及知识技术等因素，我们最终没有实现同时接收图片和文字消息的功能。

（4）项目整体总结

虽然整个小组在本次项目各个功能的开发中花费了大量的时间精力，但很多时候都是在纠结同一个功能，没有将任务具体化，导致了时间分配不均匀，以至于最后部分功能没有完全实现，这是一个遗憾。同时，整个小组虽然大家都有在努力的去实现自己的功能，但是没有考虑到将代码合并后的效果，以及合并代码后能否实现；所以在未来的学习生活中，我们要合理的分配时间，遇到不会的难题，不能独自一人去死抓，要从小组的角度去考虑问题，这样才能更有效率，更加可靠的完成小组项目内容。

**八、参考资料**

《Linux编程基础》人民邮电出版社

《C程序设计（第五版）》清华大学出版社

《QT界面设计》博客园网站

《Linux读取系统时间》CSDN网站