

软件需求说明书

作者：王聪

1. 引言

1.1 编写目的

对整个项目做总体分析，参与开发的每个人都应该读一下。该文档为概要设计和详细设计提供直接参考，为后来的测试和用户手册提供直接参考。此份文档为客户端，服务器端和此可做互相参考。

1.2 背景

a. 项目名称：XYLFTP

b. 作品名称：xylftp

c.

指导老师：陈老师，王老师

客户端开发者：孔建军，金明洁，王聪，赵崇日，周晓炜

1.3 定义

FTP—File Transfer Protocol. See RFC 959.

JVM—Java Virtual Machine. See <http://java.sun.com>.

1.4 参考资料

a. *TCP/IP Illustrated, Vol I*, W. Richard Stevens

b. *Unix Network Programming, Vol I*, W. Richard Stevens, etc.

c. *The Art Of Unix Programming*, Eric S. Raymond

d. *The Mythical Man-Month: Essays on Software Engineering, Anniversary Edition (2nd Edition)*, Frederick Phillips Brooks, Jr

e. *Unix Systems Programming: Communication, Concurrency and Threads*, Robbins, Steve Robbins

f. *Java Network Programming*, Elliotte Rusty Harold

g. 软件文档国家标准

2. 任务概述

2.1 目标

开发一个功能完善的FTP客户端，初期实现基本功能，具体见[3.1]。如果初期顺利完成，则在此基础之上进行下一步的开发，即：完成更多命令的实现，添加更多命令行选项，添加图形界面。

2.2 用户特点

xy1FTP客户端是面向Linux/Windows普通用户。

2.3 假定和约束

- A. 一般约束：参与开发的人数不超过5人
- B. 经费：无
- C. 期限：两个月（2007年3月22日——2007年5月22日）
- D. 硬件环境：运行JVM的所有机器
- E. 软件环境：Windows, Linux 操作系统
- F. 相关关系：与本项目的服务器端xy1FTPd能够交互，也能够支持其他支持RFC959的FTP服务器的连接。
- G. 在管理方针、并行操作、安全与保密方面无特别约束。

3. 需求规定

3.1 对功能的规定

（注：[]里面的表示可选，不带[]的则是必选）

- (1) 使用命令行界面，用Java开发。
- (2) 能处理简单的命令行参数，两种格式大致如下：

```
xy1ftp [-h|--help] [-V|--version]
```

显示帮助信息，或版本信息。

```
xy1ftp [-v|--verbose][-u $USERNAME|--user=$USERNAME][-p $PASSWORD | --password=$PASSWORD][-d|--debug] [$HOST]
```

建立与\$HOST的连接。

-v 或--verbose: 显示较多的额外信息

-u \$USERNAME 或--user=\$USERNAME: 以\$USERNAME 用户身份连接, 如果无此选项, 则尝试默认的 anonymous。指定此选项时必须同时指定\$HOST。

-p \$PASSWORD 或--password=\$PASSWORD: 用\$PASSWORD 密码登录。如果此项未指定, 尝试默认的空密码。指定此选项时必须同时指定\$HOST。

-d 或--debug: 显示比--verbose 更多的信息, 供开发者使用。

如果后面没任何选项和参数, 应该直接进入交互模式。

(3) 交互式地接受简单的命令, 包括:

help [\$CMD] — 显示本客户端支持的命令列表。如果\$CMD 指定, 只显示\$CMD 的用法。

quit — 退出本程序。

bye — 等同于 quit。

open \$HOST [\$PORT]— 建立与\$HOST 的 FTP 连接, 如果启动此客户端时没有指定的话。如果\$PORT 未指定, 则尝试默认的 21 号端口, 否则尝试指定端口。

user \$USERNAME [\$PASSWORD] — 指定用\$USERNAME 用户名进行连接, 如果\$PASSWORD 指定, 尝试用指定密码, 否则尝试默认的空密码。

passive — 指定使用被动模式。

pwd — 列出服务器端的当前路径。

cwd [\$DIR] — 进入服务器\$DIR 目录, 如果\$DIR 没有指定, 则不做任何处理。

cd [\$DIR] — 等同于 cwd。

cdup — 等同于 cwd ..。

mkdir \$DIR — 在当前目录下新建一个名\$DIR 的目录。

rmdir \$DIR — 删除当前目录下的\$DIR 目录。注: 目录必须为空。

dir [\$DIR | \$FILE] — 列出服务器\$DIR 目录下的文件信息。如果后面指定的是\$FILE, 则列出\$FILE 的相关信息。如果\$DIR 和\$FILE 都没有指定, 则列出当前目录下的文件信息。

ls [\$DIR | \$FILE] — 与 dir 等同。

delete \$FILE — 删除一个文件。

size \$FILE — 获取\$FILE 文件的大小。

rename \$OLDNAME \$NEWNAME — 把名为\$OLDNAME 文件重新命名为\$NEWNAME。

chmod \$MODE \$FILE — 更改\$FILE 的权限为\$MODE。

get \$FILE [\$NEWFILENAME] -- 从服务器获取名为\$FILE 的文件, 保存至本地当前路径。如果\$NEWFILENAME 指定, 保存的文件应命名为\$NEWFILENAME, 否则和\$FILE 同名。

put \$FILE [\$NEWFILENAME] -- 将本地的\$FILE 文件传送到服务器端的当前路径。如果\$NEWFILENAME 指定, 传上去的文件应命名为\$NEWFILENAME, 否则和\$FILE 同名。

close — 断开当前连接, 并返回交互式界面。

!! -- 开始执行本地 shell, 直到 exit 才返回。

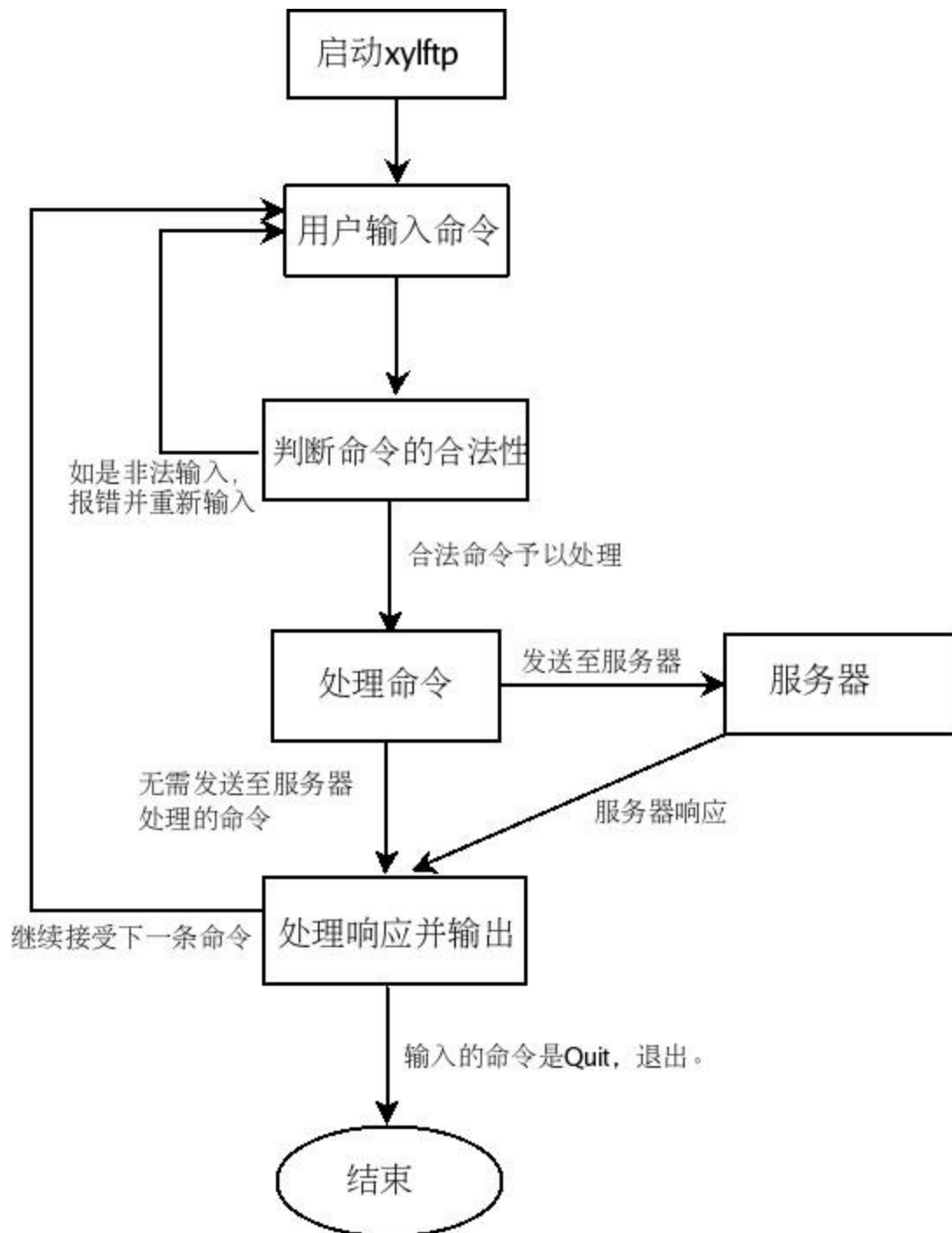
? [\$CMD] — 等同于 help。

3.2 对性能的规定

客户端应该尽可能快地对用户输入做出响应。

3.3 输入输出要求

客户端应该采用交互界面，因此还要能处理错误输入。交互输入的命令不应该对字母大小写敏感。



3.4 安全性要求

鉴于FTP协议并没要求加密传输，客户端可不必考虑太多安全性的问题。注意，传输数据时，客户端默认使用的应该是被动模式。

3.5 故障处理要求

下列错误应予以考虑，出错信息列表：

- 01-无法连接远程主机
- 02-上传文件出错
- 03-下载文件出错
- 04-远程主机关闭
- 05-未知命令
- 06-命令格式不正确
- 07-权限不足

4. 运行环境规定

4.1 设备

客户端使用 Java 编写，应该能在所有带 JVM 的机器上运行。

4.2 支持软件

编译器: `javac`

客户端应该在 Linux/Windows 上无差别地顺利运行。

5. 版权

采用 GPL v2 许可证发布，版权属于西邮 Linux 兴趣小组。