Designing the Protocol

1. What kinds of messages will be exchanged across the control channel?

At first the client will open a socket where it will initiate communications by attempting to establish a connection with the server.

1. How should the other side respond to the messages?

The server will also open a socket and determine if the connection is good or not and then wait for commands from the client. Once the server receives commands it will attempt to fulfill those tasks, send packets back to the client, and then continue to wait for commands until the client attempts to close the connection.

1. What sizes/formats will the messages have?

For our project the size of the messages will be 4096 bytes long and will have the format of UTF-8.

1. What message exchanges have to take place in order to setup a file transfer channel?

The client must initiate connection to the server, then both must reserve a port by which to send and accept messages, after this is done, the server is ready to accept requests from the client and begin file transferring.

1. How will the receiving side know when to start/stop receiving the file?

The receiving side will know when to start receiving once the server and client have both reserved a port each for communication and checked the buffer in order to prevent overflow; if both the connection is good and the buffer won’t overflow then it will begin receiving. The receiving side will know when to stop receiving the file once the complete file has finished transferring and the sender initializes the closing of the socket. Or the receiver will know to stop if it receives error whilst checking the buffer.

CODING DESIGN

SERVER AND CLIENT HAS 4 MAIN FUNTIONS:

1. Checkbuffer
2. tempSocket
3. downloadfile,
4. revFile

main:

1. while loop:

* makes a server running forever for the client to connect.

1. if else:

* check for ls , quit , get , and put file.( checking command between client and server and output the results for each of function.

Note: In server we also list off function for getting, putting, listing, and quitting for just simplify the main for more clean code.