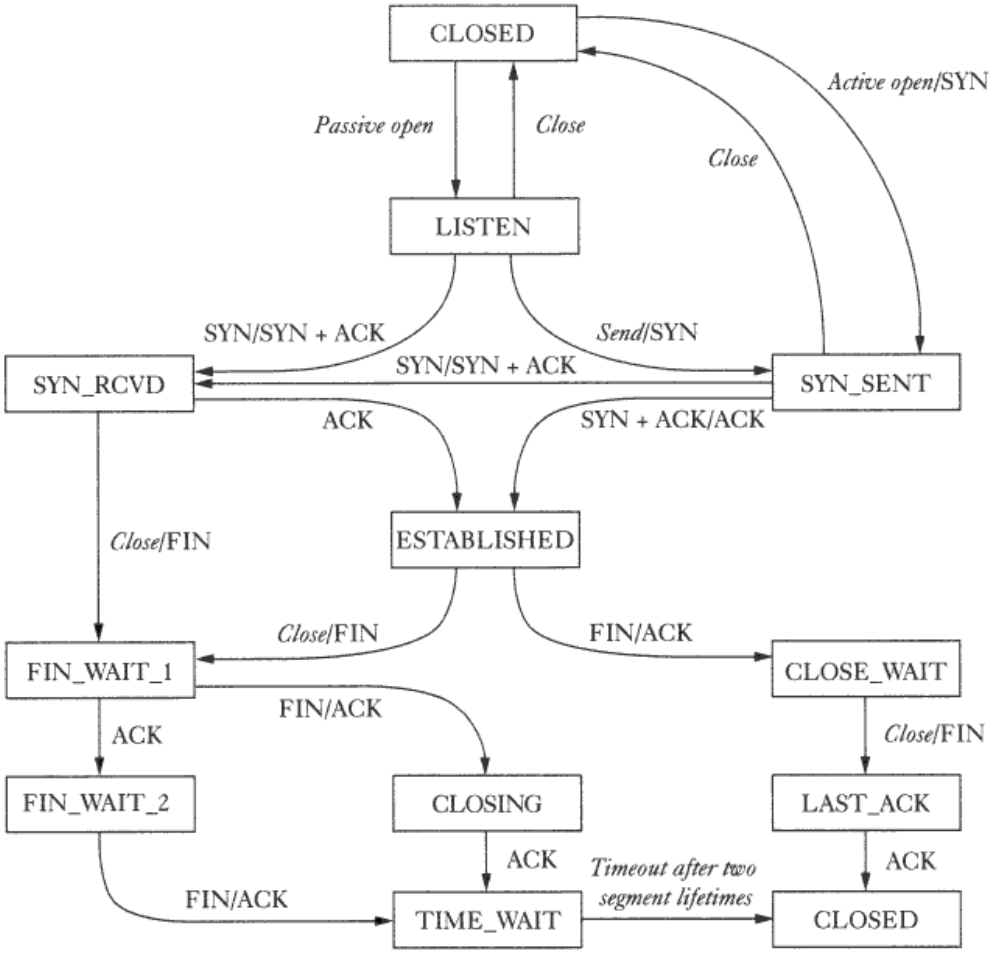
Adding reliability to UDP layer



[cite:http://www.google.com/imgres?imgurl=http://ssfnet.org/Exchange/tcp/Graphics/tcpStateDiagram1.gif&imgrefurl=http://ssfnet.org/Exchange/tcp/tcpTutorialNotes.html&h=654&w=679&sz=18&tbnid=lO5WuD_z4Ugd7M:&tbnh=89&tbnw=92&zoom=1&docid=OYQhrQmLYJVwNM&sa=X&ei=1hGcT6WKD-PY0QHyz9H7Dg&ved=0CDQQ9QEwAQ&dur=255]

The above diagram describes the state diagram of our TTP flow.
States:
Closed state: Neither side has initiated a connection.

1) **CONNECTION PHASE**: Three Way handshake protocol

Listen state:
i)
The server initially starts its receiver thread and waits for any SYN requests.
The Client initiates the connection and sends a SYN request to the server. This is the first step in the three way handshake.

ii)
The server's receiver thread acknowledges the SYN request with a SYN-ACK to the client. This is the second step in the handshake. Now the server is waiting for requests from the client.

iii)
The client knowing that the connection has ben established on getting a SYN-ACK, acknowledges the packet and then prepares to send its first request packet. Now the client and server have entered the **ESTABLISHED STATE.**

2) **DATA TRANSFER PHASE**
Client sends a data packet with filename.

Server checks for filename on it's side. If it exists, then first it reads the entire contents of the file into a byte array buffer and starts segmenting the data.
These segmented packets are then stored on the sender side in a queue.
It then sends over the size of the file to the client.

The client stores the file size and calculates the total number of expected segments.
It then sends an ACK back to the server.

On receiving an ack, the server sends all the packets at once using the go back n protocol.

The client receives all the segments in the correct expected order based on the expected sequence number. It the client does not receive it's expected segment, it just drops the packet and sends an acknowledgement of the highest received packet till then. The server on receiving this acknowledgement resends the segment with this sequence number. Thus in this manner the client ensures that even though WINDOWSIZE number of segments could be sent together, the packets are always received in order.
The client on receiving all the segments that it expected, it changes it's state to having received all data.

Flags being used :
1)SYN flag: Sent from the client end to inform the server that it wants to initiate a new connection with this server.
And other standard flags.
Extra flags:
FILESIZE:this tells the client that filesize is being sent in this data packet.
FILEPATH:This tells the server that the data packet contains file request.

3)CLOSE DOWN PHASE
Once the client has recived all its expected segments it initiates a close down by sending a packet containing the FIN flag.
Server responds with FIN_ACK.
Client sends acks ,waits for 2 MSL amount of time and then shuts down its connection.

TESTS conducted:
1)Stress test on large music files.Played on both sides.

2)Simulated dropping of packet and retransmission.

Adding reliability to UDP layer

Header format

SOURCE PORT		DESTINATION PORT
SEQUENCE NUMBER		
ACKNOWLEDGMENT NUMBER		
HEADER LENGTH	FLAG	WINDOW SIZE
DATA		

TTP LAYER Design

In this design ,both the sender and receiver each have 2 worker threads .One is for receiving the data and the other is to keep on sending data.