Audio Streaming Project Description

The project I am working on is a Server-Client audio streaming system, the Server and client use multi-threading to allow the Network API to work independently of playback, and allow playback to work independently of any program using the overall API.

**Client**

The client starts by attempting to connect to the server, once connected it will request the format information of the song, this includes the size of the song data, the sample rate, sample size [add more here], once it collects this information it then sets up the two buffers and requests the data for the off buffer.

The data for the off buffer will be received in either as the whole requested chunk of data or parts depending on the size of the buffer, as the packet from the server will include the position and size of the data, the Client will simple have to copy the data into its respective position.

Once the buffer is filled it will switch the buffers and begin playing the current filled buffer, then requests the data for the off buffer.

This will be repeated until the song has been played through, if the Client plays the buffer and the off buffer is not filled then the song will be paused and will wait until the off buffer is filled and resume playback.

If the User of the Client sets the position of playback, the Client will check if the position is within the current buffer or off buffer, if it's inside the current buffer it will simply move to that position, if it's in the off buffer it will switch buffers, begin playback and request the off buffer be filled, if the position is outside both buffers, it will request data from the position and continue on like when it starts.

**Server**

The server once started will wait for a Client to connect, once the server connects it will load up the song and wait for the Client to request the song format and the send the information, from then on it will wait for the Client to request a chunk of the song, at which it will send the raw PCM data in either parts or who packets depending on the requested amount of samples.