Design of the Voice Conference Application

Data Serialization Format:-

In our project we have used 5 classes as following.

1).VoiceConference.java –

This is the main class we use to run the app.We have to compile this class to run the app.We have to give the ipaddress of the receiver as parameters.

2).RunClient.java -

This is to capture and send the voice to the receiver.

3).RunServer.java -

This is to receive the voice and to play the voice.

4).PacketData.java -

This is the packet format class.Serialization and Deserialization of the packet has done here.

5).Statistics.java -

Packet loss has calculated here.

Here Data serialization and deserialization has done in the PacketData class.Here what we have done is we have created an object from the PacketData class and that object is converted to a byte stream through static method in serialize method.That is the data serialization format.

In Data deserialization format, we get the bytestream from the received packet in the RunServer class and deserialize that byte stream through the deserialize method in the PacketData class and receive the object that the sender has created.

Handling Loss:-

Handling packet loss has done in the Statistics class.When serializing we have added the seq no to the packet object.After receiving the packet and deserialization happens we add the seq no of the received packet to an array list.Through this array list we sort the seq nos every minute and then calculate size of the total packets received and through that we get the packet loss.

Concurrency:-

We have created 2 threads as follows.

1.to capture and send the voice
2.to receive and playback the voice

They are in the RunServer and RunClient classes.

Tests

We have created unit tests to check the serialization and deserialization formats.

Performance Measurement

For testing, Linux netem tool was used.