Design and Implementation of a Conference Server for Multiple Users
Ganeshkumar Palaniappan¹, Mr. B. Pradeep²
¹ Student, M.Sc. (Engg.), ² Sr. Lecturer (SPC), Centre for Signal Processing and Communication, M.S. Ramaiah School of Advanced Studies, Bangalore

Abstract
Conference Server is a software/hardware combination that allows three or more users to participate in a real-time conference. One Conference Server can be shared across multiple global workgroups. Conference server can be deployed and managed as a single system, which can handle voice, video, and data all at the same time. The main reason for the conference server implementation is to provide a conference control interface and the data management facilities to manage the joining and leaving of participants, route data, audio from all participants to all other participants thus providing multipoint data. It also provides the facility to handle multiple conferences and users at the same time.

In the current work a high-level design for the conference server was analysed and created initially. In order to obtain the exact and the final design for the conference server the requirements were analysed and reviewed for various functional levels. The analysis were carried out for the server authentication for participants and management facilities for the server to hold the conference details along with the members allowed to take part in each conference. The conference and participants details are stored in the database. The implementation of the project was done in two different phases, the first phase contains the conference management services while the second phase, the call control services. The software was developed using 'C' language on Linux platform.

The centralized conference server was developed in two different phases and later combined together to work as a single functional conference server. The developed conference server, in addition to the basic features, offers richer functionality including simultaneous conferences, large scalable conferences, managed conferences and can support any subset of the advanced conferencing functions

Key Words: SIP, Conference Server, Conferencing, RTP, VoIP.

Abbreviations
AOR Address of Record
CSRC Contributing Source
HTTP Hyper Text Transfer Protocol
IETF Internet Engineering Task Force
ID Identification
IP Internet Protocol
M Marker
P Padding
PT Payload Type
RFC Request for Comments
RTP Real Time - Transport Protocol
SDP Session Description Protocol
SIP Session Initiation Protocol
SSRC Synchronization Source
TCP Transmission Control Protocol
UA User Agents
UAC User Agent Client
UAS User Agent Server
UDP User Datagram Protocol
URI Universal Resource Identifier
V Version
VoIP Voice Over Internet Protocol
QOS Quality of Service
X Extension

1. INTRODUCTION
A growing trend has emerged to use multimedia communications over IP-based networks including the global Internet. Voice over Internet Protocol (VoIP) is a new trend that is emerging in recent years to provide telephone service over IP telephony. VoIP technology enables traditional telephone services to operate over computer networks using packet-switched protocols. Packet-switched VoIP replaces voice signals into packets, similar to an electronic envelope that can carry, much more information over the network to support and enhance the communication needs when compared to traditional telephony methods. Jong Yul Kim et al [1] describes about the alternative methods of acquiring the physical location of an emergency caller.

VoIP enables transmission of audio on the regular Internet network, which till recently, has primarily been used to transmit data. This integration presents great opportunity for both service providers and consumers. VoIP has various advantages like lower costs, greater consumer control and location flexibility. Its typical problems have been quality of service (QOS) and standardization. The other contributor to this VoIP growth is the development and wide acceptance of the session initiation protocol (SIP). SIP and H.323 have been used for VoIP applications as shown in Fig 1 and SIP is gaining popularity as a flexible session oriented protocol approved by the IETF [2].
SIP protocol is used for establishment, modification and termination of sessions [4]. SIP is a popular choice for establishing media sessions and instant messaging. It is the most popular and widely accepted protocol for VoIP. There are several IP softphones and hardphones available in the market today that are SIP capable and are being used for VoIP. SIP is also used for multi-party conference calls.

SIP is a catalyst for the next phase of open communications using not only IP telephony and VoIP but also the suite of IP-related protocols. SIP is an interoperable suite of protocols designed to allow equipment from different vendors to communicate with each other. SIP brings about increased efficiency and productivity. In a VoIP converged network with SIP, organizations can choose from a variety of vendors to create a seamless converged communication network. The communication equipment manufacturers have the freedom to implement SIP. Implementation of SIP allows their equipment to communicate with equipment from other vendors. SIP has emerged at the forefront of most VoIP-related applications. SIP has been embraced by the leading VoIP telephony manufacturers and is being built into VoIP hardware and software, including IP-enabled telephones [5].

Multiparty conferences are becoming an important issue not only for Internet applications but also for data exchange. SIP is used to create the sessions for these conferences, which can be of open and closed type. In closed conference participants know the identity of each participant. Also the signalling protocol should manage participant identities and distribute them to all conference participants. Conference server needs to take care of these properties in a closed conference.

Conference server plays an important role in each and every conference as users dial up into these servers for conference participation. It also acts as the application to the available SIP stack in the industry and allows users to connect to the server. The server is capable of handling multiple users and multiple conferences at a single time. It is built up in such a manner that it plays a welcome message to the participant as he or she dials up and gets connected to the conference and allows only authorized persons to take part in the conference.

The authorization is done by the conference server by collecting the details such as conference ID, caller ID, and password from the participants. By playing corresponding messages with the help of RTP the details are collected from the participants. RTP protocol is a real time transmission protocol used for transmitting the data in real time applications. RTP plays an important role in the conference server implementation as it plays the part of playing the audio messages to the user through it. The conference server then verifies the collected details and validates it. If the validation is correct and each detail matches with the database then the participants are allowed to take part in the conference. If any of the entered detail mismatches with the corresponding detail in the database then the conference plays an invalid message to the users indicating that the entered details is not correct hence they are not allowed to take part in the conference.

2. FRAMEWORK FOR CONFERENCING WITH SIP

J. Rosenberg [6] has defined the framework for conferencing and described the overall architecture, terminology and protocol components needed for multi-party conferencing. SIP can support many models of mult-party communications like loosely coupled conferences, fully distributed multiparty conferences and tightly coupled conferences. The tightly coupled conferencing is suitable for multi-party communications as it provides a variety of conference functions, and may possibly perform media mixing functions as well.

2.1 Conferencing Architecture Overview

Conferencing is a specific term that has its own architecture and functionalities. The architecture consists of several blocks in it, which acts as interfaces and blocks in real time communications. The FOCUS is the central component in a SIP conference and maintains a SIP signaling relationship with each participant in the conference. FOCUS is responsible for the media streams with conference available to the participants.

![Conference Architecture Overview](image)

FOCUS uses the media policy to determine the proper configuration of the mixers. The result is a star topology, shown in Fig 2. It has access to the conference policy, an instance of which exists for each conference and can be a database that describes the way the conference should operate. It is the responsibility of the FOCUS to enforce those policies and also needs to know when it has changed. The conference is represented by a unique URI that identifies the FOCUS and are routed to the FOCUS for that conference. Participants join the conference by sending an INVITE to the conference URI. As long as the conference policy allows, the INVITE is accepted by the FOCUS and the user is brought into the conference [6].

Users can leave the conference by sending a BYE, as they would in a normal call. A FOCUS can also initiate an INVITE to bring a participant into the conference. The
participant can communicate with the conference policy server using some kind of non-SIP-specific mechanism by which it can affect the conference policy [6].

2.2 Physical Realization

Conference Server can be physically realized in five different ways to solve a variety of problems. All these realizations can be physically realized based on the required applications. The five different realizations are as follows,

- Centralized Server
- Endpoint Server
- Media Server Component
- Distributed Mixing
- Cascaded Mixers

Here the implementation of Centralized Server is presented.

3. MODEL CONSTRUCTION

The conference server is a device used to create, join and terminate conferences in real time applications. It uses many protocols for this purpose. The implementation is done with the help of SIP protocol to build up the conference server. SIP is an advanced protocol easy to construct in real time applications. The conference server is build up with SIP-controllable media mixer, which provides a network service for setting up conference and mixing audio streams. The conference server uses SIP signaling with the UA and the mixer to establish media between the two.

The supporting mobile Internet multimedia applications requires more than just the ability to maintain connectivity across subnet changes. SIP can help to provide terminal, personal, session and service mobility to applications ranging from Internet telephony to presence and instant messaging [7].

3.1 Components of Conference Server

The conference server includes three principles components in it in the form of Management Services, Call Control Services, and Multi-Media Services. SIP is used by the user agents to dial into the conference through this conference server to create a session between them. Conference server uses RTP to provide media functionalities for the user when he tries to connect to the conference through this server. Fig 3 shows the relationship between the conference server and the user agent (participant) and also gives a clear idea of the services provided by the conference server.

3.2 Design for Centralized Conference Server

It is the most simplistic realization. There is a single physical server in the network, which implements the FOCUS, the conference policy server, and the mixers. The design for the centralized server is shown in Fig 4.

This conference server is a unit that is able to administer many simultaneous connections to multiple UA and connect them all together. In order to establish a SIP session, a SIP-dialog should take place between each UA and the conference server.
3.2.1 FOCUS
As the name implies, the FOCUS is the centre of the conference. Conference participants are connected to it by a SIP dialog. The focus is responsible for maintaining the dialogs connected to it and ensures that the dialogs are connected to a set of participants who are allowed to participate in the conference, as defined by the membership policy. The focus also uses SIP to manipulate the media sessions, in order to make sure each participant obtains all the media for the conference. To do that, the focus makes use of mixers [6].

FOCUS receives an INVITE and checks it with the conference policy to decide whether the participant is allowed to join or not. If the policy does not allow the user in which case the call can be rejected. Conference participants can remove themselves from the conference by sending a BYE to it. The same process can also be done by the focus which is often referred to as “ejecting” a user from the conference, and is called “mass ejection” when done for many users.

Participants in the conference are the SIP UA’s that has a dialog with the focus. This SIP user agent can be a PC application (a softphone), a SIP hardphone, or a PSTN gateway. The participant can also be another focus. Participants play an important role in the conference and without them the conference are almost not possible.

3.2.2 Conference Policy Server
The conference policy server is a logical component of the system. It represents the interface between clients and the conference policy that governs the operation of the conference. Clients communicate with the conference policy server using a non-SIP-specific mechanism [6].

3.2.3 Mixer
The mixer is responsible for combining the media streams that make up the conference. The process of combining media is specific to the media type, and is directed by the focus, under the guidance of the rules described in the media policy. A mixer is not aware of a “conference” as an entity, per se. A mixer receives media streams as inputs, and based on directions provided by the focus, generates media streams as outputs. There is no grouping of media streams beyond the policies that describe the ways in which the streams are mixed [6].

Mixer, either directly or indirectly always acts under the control of a focus. The focus is responsible for interpreting the media policy, and then installing the appropriate rules in the mixer. If the focus is directly controlling a mixer, the mixer can either be co-resident with the focus, or can be controlled through some kind of protocol. If the focus is indirectly controlling a mixer, it delegates the mixing to the participants, each of which has its own mixer.

3.2.4 Conference Notification Service
The focus can provide a conference notification service. The service accepts subscriptions from clients for the conference URI, and generates notifications to them as the conference state changes. Conference state includes the participants connected to the focus, and also information about the dialogs associated with them. When a participants joins the conference and when someone leaves from the conference, this state changes, and is reported through the notification service.

Conference policies are the rules that guide the operation of the focus. These policies can be simple, such as an access list that defines the set of allowed participants in a conference or it can be complex, specifying time-­of-­day­-based rules on participation, conditional on the presence of other participants.

Here the policies are made in a simple way to authenticate the list of allowed participants. There are no restrictions to encapsulate on the type of rules in a conference policy. It is mainly the implementer’s wish that can decide upon these policies.

Participants in the conference are the SIP UA’s that has a dialog with the focus. This SIP user agent can be a PC application (a softphone), a SIP hardphone, or a PSTN gateway. The participant can also be another focus. Participants play an important role in the conference and without them the conference are almost not possible.

3.3 Centralized Server – Architecture
The architecture for the centralized conference server has been constructed based on the requirements needed for the implementation of the conference server. Aameek Singh and Arup Acharya [8] describes ways of implementing a conference server such as centralized server and decentralized server type of implementation. Fig 5 shows the architectural diagram for conference server.

Mixer, either directly or indirectly always acts under the control of a focus. The focus is responsible for interpreting the media policy, and then installing the appropriate rules in the mixer. If the focus is directly controlling a mixer, the mixer can either be co-resident with the focus, or can be controlled through some kind of protocol. If the focus is indirectly controlling a mixer, it delegates the mixing to the participants, each of which has its own mixer.

The main features in this architecture are that SIP is being used for Signalling and RTP is used for media. This conference server can support multiple simultaneous conferences. Audio mixing is done in
such a manner that participants do not get their own audio.

3.4 Conference Management Services
The conference management services provide various functionalities that make up the conference a much-stabilized one. The management service plays an important role in the conference because it is the one that manages the whole conference. Conference management service provides four basic services in the form of Call handling, Manage signalling, Scheduling conference, and other management services. In all these services the implementation part is the scheduling conference part. Fig 6 gives a clear idea of the conference management services used for the centralized conference server implementation.

Schedule conference includes collecting the data for the conference in terms of members list and conference list that are going to be active. These data will be useful while authenticating a user for a conference. For authentication these data are to be stored in one database, which will be taken care by the management services store data part. When the user tries to dial into the conference the conference policy will check this stored database for allowing them to participate in the conference.

Fig 6 – Conference Management Services
Distribute data of the conference management services will be useful for the conference server initiated conference. But this part is not taken care in the present implementation as it is mainly based on user joining conference. These three parts form the scheduling part in the management services which will be taken care some other external tool to write into the database. Currently the collecting and storing of data are performed on the basis of files operation done using ‘C’ program.

3.5 Call Control Services
Call Control Services are the services that form the entire flow of the call starting from the moment user dial up into the conference server till he gets connected to the actual conference. These services are built up to face the controlling action involved in the conference server. This service plays an important role in the conference server building block and functionality.

Users initially dial into the conference with an INVITE request sent to the conference server. The conference server then accepts the incoming signal and plays the welcome message to the participant. When the welcome message played successfully, the message for obtaining ConferenceID is played. Both the messages are played using RTP protocol.

The implementation part for this part is to create a buffer so that media data are read from the stored path with duration of 80ms. This is because RTP supports 80 bytes of data at a time; hence a delay is also created to read the data continuously till the end. When messages are played the conference server waits for the signal from the SIP stack. This signal is to indicate the conference server to collect the ConferenceID in the form of digits. The ConferenceID is collected and validate with the list of available active ConferenceIDs in the database.

Fig 7 explains the entire flow of the operations performed in the call control services. These are the operations involved in the conference server as the user dial up into the conference server to take part in the conference.

Fig 7 – Call Control Services
If the ConferenceID matches then the CallerID will be collected from the INVITE request. The same process of validation is done for the CallerID. If the CallerID matches with the corresponding conference then the message for UserPassword is played to the participant. Then the process of collecting the ConferenceID is repeated for collecting the UserPassword form the participant. The validation check for the collected UserPassword is done. If the UserPassword matches with the corresponding password of the CallerID then the participant is added to the conference.
allowed to take part in the corresponding conference in which the participant wants to join.

When in the validation check if there is any mismatch with the active data then the message Invalid is played to the participant indicating that the entered detail does not matches with the active conference details. This forms the major working of the conference server and this part acts as the call control services for the conference.

3.6 Media Management Services

Media Management Services are the services that are useful for the conference server for playing the media messages. In order to use these services buffers were created. The media services are explained in Fig 8.

- Create Buffers
- Split The Audio Data Into Smaller Junks
- Read The Smaller Junks of Data With Delays
- Pass The Read Data in RTP To The Participants

Fig 8– Media Services

These buffers have the ability to split the data into smaller junks of data (i.e.) 80 bytes. The data are made into smaller parts because the RTP layer used for transmission of data supports only 80 bytes of data to be passed through it at a given specific time. A delay of 10 msec is required for the RTP to send the next packet through it.

D. Clark, and D. Tennenhouse [10] describe the functionality of RTP. The data to send is split into smaller parts, covered by a RTP header. The most important parts of this header are the sequence number and the timestamp. RTP does not provide any mechanism to ensure timely delivery or provide other Quality-Of-Service (QOS) guarantees, but relies on lower-layer services to do so. It does not guarantee delivery, nor does it assume that the underlying network is reliable and delivers packets in sequence. Sequence numbers in RTP allow the receiver to reconstruct the sender’s packet sequence.

RTP is primarily designed to satisfy the needs of multi-participant multimedia conferences, it is not limited to that particular application. RTP consists of two closely linked parts:

- RTP – To carry data that has real-time properties.
- RTP Control Protocol (RTCP) – To monitor the QOS and to convey information about the participants in an ongoing session.

The media services are controlled by the FOCUS that takes control of it. RTP layer has the ability to pass the data from one end to another end through it without any difficulty as buffers take care of splitting the data and sending it to the RTP layer.

3.7 Characteristics of Centralized Conference Server

The following are the characteristics of the centralized server that provides additional functionality for the conference server.

- Maintains point-to-point and media relationship to all participants
- Conference Users are identified by request URI
- Anyone who has the conference URI may join the conference provided the conference policy permits the user
- Conference participants may send a BYE request to the server to terminate from the conference
- Authentication is done for each and every user who joins the conference
- FOCUS has the ability to connect all users simultaneously so that no user is made to wait in the queue
- RTP performs the media functionalities and helps participants to hear the announcements provided by the conference server before allowing the participant to join the conference

3.8 Conference Management Services

The Conference Management Services consists of three modules to be designed and implemented in it. The first part of this service is to collect the required data for the conference and the members allowed to take part in it in the database. The process of collecting data is done. The second part of this service is to store the data in the database in either of the format that will be accessed during the validation period. The third part of this service is to distribute the stored data at the time of the conference. This service will be used for conference server invited conference.

3.9 Call Control Services

The Call Control Services consist of the flow of the conference server application from the participant dials into the conference till he gets connected to the conference. Fig 9 shows the information box that will be displayed in the system of both conference server and the dialed user system.

The ‘Title Box’ consists of the title in which it operates. The titles are as follows.

- Originating a Basic Call
- Accepting an Incoming Call
- Rejecting an Incoming Call

When a participant dialed into the conference server, system has already “Originating a basic call”. In the mean time when there is an incoming call the system
Final design for the conference server in the form of Centralized Server was finalized due to its extensibility, reliability and adaptability to various situations.

The unit testing for the modules has been successfully made and the integration of the created sub modules has been done and tested for the whole conference server application.

The design was coded in two different integral parts using ‘C’ Language.

The header files for the conference server application was created and it can be made programmable depending upon the application and the availability for the users.

The implementation of these codes has been carried out on Linux platform.

REFERENCES