Mission Control Room Conferencing using Standard PABX Systems

Dona Liz Peter[1] *and Markus Töpfer*[2]
German Space Operations Center (GSOC) - German Aerospace Agency (DLR), Weßling, 82234, Germany

For Space Mission Control Room operations, voice communication has a high importance as it is used to transmit voice information, coordinate between different units and entities and to alert in case of unforeseen or critical events. The content of this paper is based on a GSOC study, which emphasizes on the core concept in these kind of environments, multi-conferencing. It discusses an implementation of this functionality within standard telephony systems. A multi-conferencing system is an environment where several voice conferences are running in parallel. It allows a user to participate in more than one conference at the same time. The multi-conferencing concept described here is not a commonly used public conferencing environment. It is most commonly used in control room conferencing for Space Mission Control and Air Traffic Control. The attempt is to implement multi-conferencing capabilities by reusing the basic features within a standard Private Automatic Branch Exchange (PABX) system. The basic call features of a standard telephony system are reused in order to achieve the goal.

The approach of this study is to analyze and prototype an alternative idea towards the switching mechanism for multi-conferencing capabilities in the Voice Conferencing Subsystems (VoCS) of the German Space Operations Center (GSOC). The VoCS is responsible for the voice communication with different space agencies. Embedded in a more advanced version around the multi-conferencing core implemented within PABX systems, this idea may be reused in any of VoCS applications like those with the Columbus module of the International Space Station, which aims at providing communication between ground, the astronauts on-board and between various ground centers all over Europe.

The work follows the concept of Role Based Access Control (RBAC). Clients attending the conferences are classified into different groups based on their roles. An authorized member in a multi-conferencing system can join or leave several dedicated conferences at the same time. Certain permission sets are predefined for each user in order to control his accessibility to conference states. The conference states include the listen/monitor state and the talk state. The user permission to access the state varies for each conference to which he is connected, depending on the purpose of the conference. Implementation of these kinds of permission sets is a critical functionality within operational mission control room environments. Applying this permission distribution within a standard PABX is a new way to approach this functional need.

Another important part of the environment is actual audio conferencing. An idea to attend multiple conferences at a time is achieved by certain internal call routing to various conference rooms. The audio traffic from different conferences attended by a user at a time are mixed and is sent back to the user by a single call back functionality through the voice gateway. This in turn results in reducing the bandwidth requirement of the whole system.

To prototype this concept, an open source software based switching platform called FreeSWITCH (www.freeswitch.org) was selected. FreeSWITCH has got both media gateway and switching functionalities. The voice gateway is implemented using a configuration, which reduces the complexity of the work. It provides complete flexibility in implementation and functionality and also allows extension for future demands in a rapidly evolving and ever changing aerospace research environment. The system is embedded in a higher order system, which operates within a web based framework.

With an entirely novel approach of reusing the default functionalities within standard PABX systems it was tested and proved that the multi-conferencing capabilities can be implemented. Prototyping of a multi-conferencing system was made possible with a standard open source telephony switch. With this concept full control over the system could be accomplished without actually implementing the whole system. Performance analysis and robustness analysis are the next tasks to be done in this context.

[1] Voice Systems Engineer, DLR RB-KOB Communications and Ground Stations, Münchener Straße 20 82234 Weßling Germany, Dona.Peter@DLR.DE
[2] Voice Systems Engineer, DLR RB-KOB Communications and Ground Stations, Münchener Straße 20 82234 Weßling Germany, Markus.Toepfer@DLR.DE

Successful analysis could lead to using this idea of multi-conferencing in the Voice Conferencing Subsystems as a potential alternative to currently used conferencing systems.

I. Introduction

VOICE communication and audio data exchange are essential services for supporting and coordinating space mission operations. This area is very important especially in cooperative international environments like satellite mission support and human space flight programs. Information exchange and coordination is a fundamental part of space missions. This paper provides a draft system design and prototype implementation for a voice communication model based on the functionality of the existing Voice Conference Subsystem [3,5,6] of the German Space Operations Center (GSOC) using a Private Automatic Branch Exchange (PABX) as basement for conferencing.

With this work, conferencing capabilities are enabled along with user permission control and dynamic access control. The main idea is to use the capabilities of a standard open source PABX system to prototype the system. In this way, full control over the system without implementing the whole system achieved. The question to be answered is if it is possible to reuse the basic functionality of a standard PABX system to implement multi-conferencing capabilities required for space mission control room conferencing. This work is part of a feasibility study on the replacement of the complex system which is presently in-use.

A. Motivation

The motivation of this work is mainly based on the need of a replacement of the current voice system in use for control room conferencing of the Multi-mission operation environment. The existing system is reaching end of its life and requires high costs for its operation and maintenance. In order to reduce the complexity of the system, a modularization approach is adopted. A modular system will make the system extendibility, interoperability and maintenance much easier. Simplification of the system is another important driving factor considered in the whole work.

B. Background description

Multi-mission operations are of a dynamic nature, where the system has to serve communication needs of dynamically changing endpoints. The operations involved in Multi-mission environments can be classified into two use cases; standard operations for maintaining the space crafts and special operation scenarios like Launch and Early Orbit Phases (LEOP).

Standard operations are regular operations supporting the satellites which are already in the functional state and in their final orbits. In this phase voice conferencing involves only a few voice-loops in parallel but with different external connections depending on the mission under support and the Ground Stations is use. This is related to a dependency on the mission characteristics, which use different ground terminals for data up and downlink e.g. for Geostationary or Low Earth Orbit satellites.

The use case of special operations like LEOP is different. A lot of parallel running conferences for a single mission are required. In this scenario the focus is set to parallel conferencing capabilities. LEOP phases include a lot of coordination works that need to be communicated between different groups in the mission control rooms. This involves a lot of voice-loops to enable parallel conferencing between the different controlling and co-coordinating entities for separation of communication groups.

The voice conferencing system, which is currently in use for Multi-mission operations, is a very complex proprietary system and handles the voice communication in a dynamic way. It is been intended to replace the voice systems for Multi-mission in the near future as it is reaching the end of its support period.

To question discussed in this paper is: Is it possible to replace such a complex proprietary system with a lightweight open source solution without losing required functionalities for both use cases.

III. System Description

The main intention of the work is to enable a user to participate in multiple conferences in parallel. The kind of conference which is most common in the field of conferencing is multi-party conferencing where multiple users can join a single conference. This is a one-to-many relation. What is needed for Multiparty Multi-conferencing is a single user attending different multi-party conferences at the same time, which is a many-to-many relation. Figure 1: Multi-conferencing scenario; illustrates the general scenario[9] of multi-conferencing. It shows 'N' number of parallel running multi-party conferences. The represented user is able to access as much of parallel running conferences as he wants at a time. He is therewith attending a multi-party multi-conference, a conference of conferences. Attending a conference is provided by means of call features. This way dynamic accessing capabilities are given to the user.

Each conference is a dedicated one, and named a voice-loop[4,5,6] in Space Mission Control Room environments. A voice-loop is a real-time auditory channel that connects physically distributed people. A user who speaks on a voice-loop broadcasts to all users who are listening to that loop. A user monitoring a voice-loop hears any communication among other users connected to that loop. Only an authorized and authenticated user is given talk or monitor access. All the members joining a particular voice-loop will have some related responsibilities. This implies that the loop to which a given user has access to, are those in which the user has got some tasks to perform.

User interaction is based on the permission set defined for every user. A user needs some predefined access rights to join a conference. Rights are defined in order to access the conference states. There are two conference states; one is to listen/ monitor the conference and the other is to talk in the conference. On the basis of the user's role state access rights are assigned for each conference.
In short the main intentions can be summarized as follows:

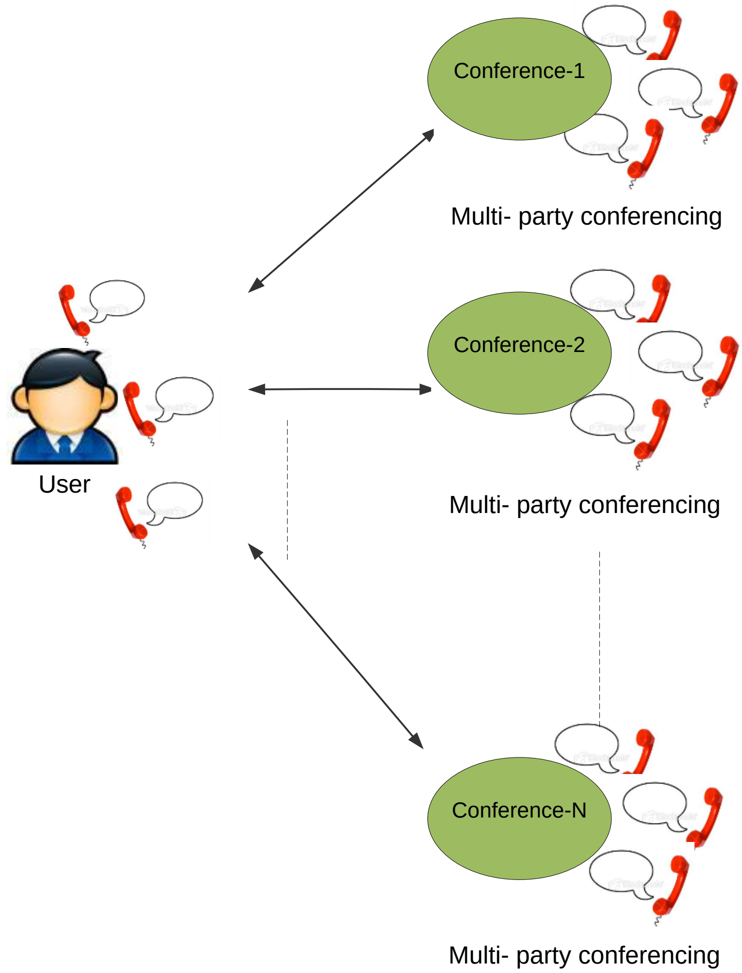

Figure 1: Multi-conferencing scenario

1. Enabling a single user to be capable of accessing multiple conferences at a time.
2. Assigning permissions for user interaction based on a user's role.
3. Dynamical accessing capabilities for the user.

The attempt is to realize these intentions by making use of some basic functionality within a standard open source tool.

A. High level component model

The whole system is defined in terms of a user-server relation model. A user is the one who participates in a multi-conference and a server enables the user to do so. The main functionalities necessary for the system is to receive and send audio signals as well as to control the audio which the user wants to receive. Even if both functionalities are interconnected, they don't necessarily need to be implemented on the same device.

For the conferencing part of the overall concept, a PABX environment shall be used. For this purpose a suitable protocol to split between controlling calls and transmitting audio need to be used. This is available with Session Initiation Protocol (SIP)[7,11] based VoIP calls. Using SIP as the signaling protocol and Realtime Transport Protocol (RTP)[8] as the transmission protocol offers capabilities to split control and media streams. At the same time it is implemented in each PABX system. Access to multiple conferences can be provided with multiple SIP calls and media transmissions are possible with different RTP streams. Implementation of the switching part within the PABX for SIP calls in the context of Multiparty Multi-conferencing is the central idea of the system design.

The user device can be a laptop or a webpage or a SIP telephone which has got the audio devices connected. The idea is to have a provision to send and receive the audio from the conference irrespective of the audio device used for the purpose and resultantly to make the system device independent. Any audio system that includes a microphone, speaker and a Push to Talk (PTT) button could be used for the purpose. The microphone enables the user to transmit the audio to the conference and the speaker enables the user to listen to the conference. A Push To Talk button will enable the channel when the user wants to talk. Only if the PTT button is pressed, an authenticated user can transmit audio into the channel. The next important section at the user side includes and presents available voice-loops for each user individually. Internal signaling is maintained by standard call functionality.

At the server side there are also independent but loosely coupled audio and control components.
The audio component is responsible for the audio transfer between the user and the server and for switching of audio. A user can send or receive audio to a conference with RTP streams. Users connect to a conference with a SIP call. This SIP session keep the signaling between the user and the conference. Once the user is connected to the conference, he can listen or talk accordingly to the rights he owns. Audio data transfer is carried out with RTP streams. The audio component enables audio transfer as well as switching of audio between conferences. To do that, the audio device obviously needs to be registered to the audio component of the server. All the access controls rights and user interaction rights will be dealt by the control component of the server. The control component is also responsible for controlling audio switching and audio transmission.

B. The switching part of the conferencing module

In order to enable a single user to access multiple conferences at a time there is a need for a switching mechanism which can connect the user to the particular loops and can get back the audio from all the loops he selected. The switching functionality is realized at the server itself. The switch connects the user to conferences to which he has access rights and sends back the audio from all the conferences the user decided to participate in. Conference selection, audio mixing and sending of audio back to the user are the main functionalities of the switch. To feedback audio the switch need to select audio from all the conferences a user connected to and need to mix it to user's audio stream.

Figure2: Switching mechanism in conferencing component; shows the idea of the switching component. The switching center play its role between a user and N parallel running conferences

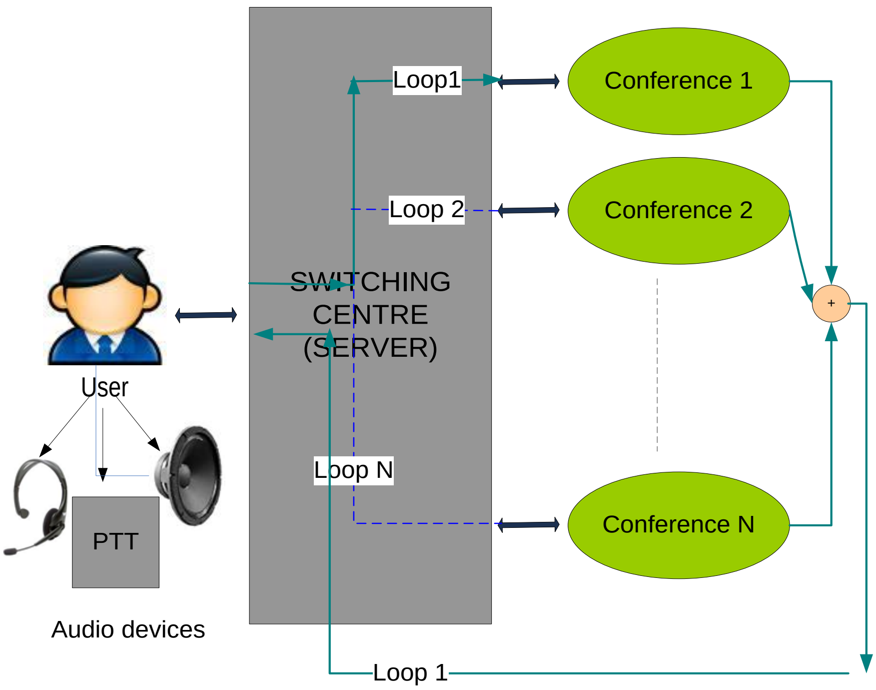

Figure 2: Switching mechanism in conferencing component

Given in the figure are N parallel running conferences to which the user has permission to access. These conferences are all classical multi-party conferences. The switching mechanism should be configured in a way, that allows a user to join a particular conference to interact with and the audio given back to the user contains the mixed audio from all the different conferences he selected. This will enable the user to take part in N number of different conferences at a time. All the internal connections are made using basic call features.

To enable the mentioned functionalities, the switch first needs to connect the user to a conference which he wants to join. The switching center connects the user to the conference which he wants to interact with, by a call made to the conference. Once the call is answered, the user joins the conference. During the call a session is established between user and conference, which acts as signaling channel. Immediately the audio from all the accessed conferences of a particular user will be mixed. The user will get back the mixed audio through a particular loop, which is enabled by the switching center. A single RTP stream of the mixed audio is sent back to the user.

a. Multi-conferencing idea

The user should be capable of attending multiple conferences at a time and he must get back the audio from all his accessed conferences. Figure 3: Multi-party multi-conference idea representation; is a diagrammatic representation of the idea to enable multi-conferencing.

The figure represents a user attending four different conferences at a time. There are four parallel running multi-party conferences, conference-I, conference-II, conference-III and conference-IV each represented by a block in the diagram. The user mentioned here has all the access rights predefined at the server, so that he can join all the given conferences. The user selects a conference by making a call. In order to give back audio to a user from all the conferences which he can access, audio mixing need to be done. For realizing the audio mixing, all the conferences are connected to a second conference called multi-conference. The members attending the multi-conference are other conferences; hence it is a conference of conferences. At the multi-conference the audio from all other conferences are automatically mixed and is given back to the user. All connections used in the system are one way connections. From user to conference, from conference to multi-conference and from multi-conference to user, the data is transmitted only in one direction.

Idea is to make use of the basic call features within a standard PABX to enable the switching mechanism, which includes the conference selection by the user, the audio mixing and to give back the mixed audio to the user. User interaction to the selected conference realized using the audio devices at his side. A specific user side device implementation is out of scope for this paper.

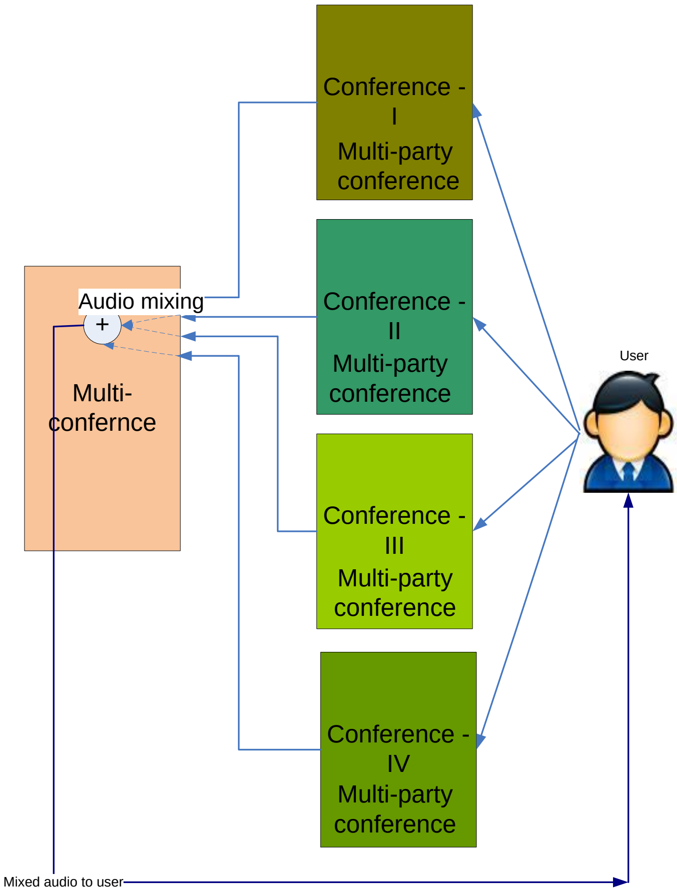

Figure 3: Multi-party multi-conference idea representation

b. Access control for user interaction

The control component at server side deals with the access control rights for user interaction. Once the permission is assigned, a user can access the conference states (listen/talk) by making a call to the conference. SIP is used for signaling between the server and the client. Once a call is made to the conference, a channel will be established between the client and the server. Audio transfer through the channel depends on the rights assigned to the user for accessing conference states. If the user has the right to talk, then the audio transfer is possible from user side to the conference. In other cases, the user will be automatically muted in the conference.

C. Scalability of the proposed system

Implement a system which is highly scalable to the increasing demand from different users is highly important as the system need to be used in standard operations as well as during LEOP phases. The proposed system can make use of its basic call functionality to extend the conferencing capabilities to any number of users. Extendibility is easily possible by coupling of conferences between different system instances. Conference coupling or bridging is done by calls which are made from one system to another. A system can support a particular number of users and conferences, which is based on the capabilities of the systems in use. The load is dependent on the amount of conferences in parallel and the amount of participants in parallel and either conferences or users could be scaled out.

IV. Implementation

For implementation, an open source soft-switch called FreeSWITCH [1,2] was selected. FreeSWITCH is an open source soft-switch [10] which satisfies the required gateway and switching functionalities. This open source tool has well maintained documentation. In addition a soft-phone called Ekiga is used as the user side device during prototyping.

A. The basic Implementation Setup

The basic implementation setup consists of a SIP server and a SIP client. In this study the FreeSWITCH Server is the SIP Server and an Ekiga soft-phone is used as SIP client. Figure 4: The basic Implementation setup; shows the basic setup which is reused in a way to attain the final goal.

The user who has been already registered with the Ekiga soft-phone to the FreeSWITCH is able to make a call to the server and to receive a call from the server. Basic protocols supporting the communication are Session Initiation Protocol (SIP) and Realtime Transport Protocol (RTP)

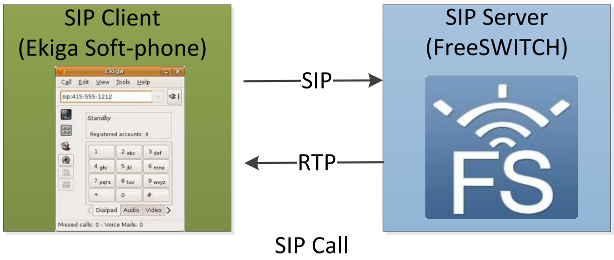

Figure 4: The basic Implementation setup

B. Implementation of the conferencing concepts

Users are created and registered over the Ekiga SIP phone to the FreeSWITCH server. Basic call features of FreeSWITCH are reused for the work in a way to enable multi-conferencing capabilities to users. The first step in this direction was to create a conference configuration [12,14]. Configuration includes conference profiles, groups and caller-controls. Once a configuration is defined for a conference, an authenticated user can join the conference.

Once a conference name is used in the dial-plan [12,13] for the first time, this conference will be created on demand and will be bound to a particular profile. The logical space to which all the members of a conference are connected together to communicate is called a conference room. The dial plan is made in such a way that the client will be connected to a conference when a call is made to a particular conference room. In this work each conference room is represented by a four digit number. So the client needs to dial that four digit number to join the conference. Each conference will be connected to a particular profile which is already defined in the conference configuration. Also the user in a conference is assigned certain caller-controls to control the conference features at the member side. These caller controls are defined in the conference configuration. Members who are allowed to join the conference are grouped into different categories. This grouping is based on their roles.

C. Call routing in multi-conferencing.

A call loop is a complete call circle which shows all the points which the call passes through from its originating point till the end point. Figure 5: Representation of a call routing in multi-conferencing; shows the call loops and call routing in multi-conferencing.

The client 1000 shown in the figure is an Ekiga client to access one stream. This client represents a SIP session and belongs to the user '100'. The blocks conference-1, conference-2, conference-3, up to conference-N represent the conference rooms which are assigned a unique four digit number. Each conference can be accessed by the user by dialing the particular four digit conference room number.

When the user makes a call to join conference-1 by dialing the assigned conference room number, he will be assigned all the user rights first and then will get connected to conference-1. From this conference another call is generated automatically which connect conference-1 and all the other accessed conferences of user 100 to a second conference called 'multi-conference'. At multi-conference, the audio from all the conferences is mixed and given back to the user through his default SIP stream, in this case 1000. Call 1 is the standard call from the user to a conference and call 2 is the call towards the user from the multi-conference, which carry the audio from all accessed conferences of the user.

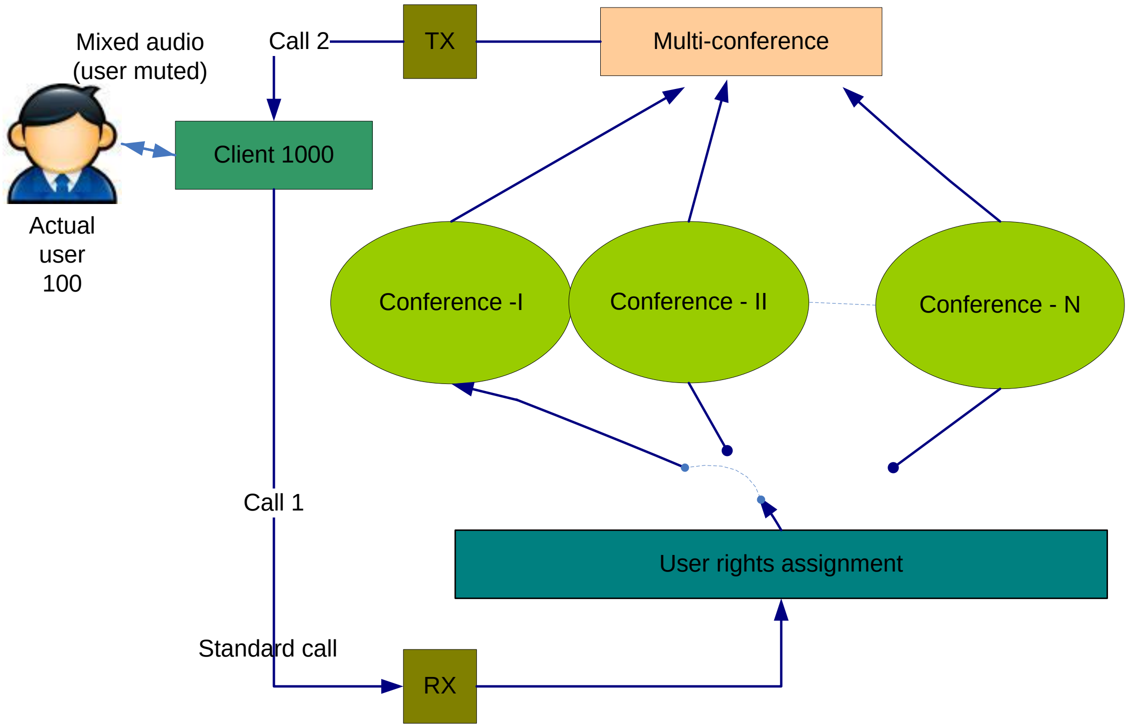

Figure 5: Representation of a call routing in multi-conferencing.

D. Permission Control

The members of the conference are given certain permissions to access the conference states. The conference states are the listen and the talk state. Members are given access permissions based on their assigned permission sets. If the permission is set to monitoring the incoming call will be muted automatically. If the permission is set to talk, the call is connected like any other call. Permission sets are controlled by dial-plan [12,13] configurations of FreeSWITCH.

E. Signaling and bandwidth consideration

Internal signaling in multi-conferencing is done with calls. A call connects the user to a conference, different conferences to multi-conference and multi-conference back to the user. During a call, SIP sets up the session between the end points and provides signaling. Audio exchange is enabled over RTP when the user connects a microphone to the loop. The SIP signaling stream is the one that enables the user to select or leave a conference. The SIP session is always established to the conference in order to keep the user connected. Although the SIP session is always active, RTP streams sends or receives data only if there is actual audio exchange required. If a user can access N parallel conferences at a time, he uses N signaling streams, a single audio stream for talking to a conference as well as a single receiving stream.

Figure 6: Signaling and call flow diagram; represents the signaling and call flow in multi-conferencing. The blocks 1000, 1001 and 1002 are the Ekiga clients which are used by the user for multiple stream access. Bandwidth of the system is much reduced as there is only a single stream towards the user which carries the audio from all his accessible conferences.

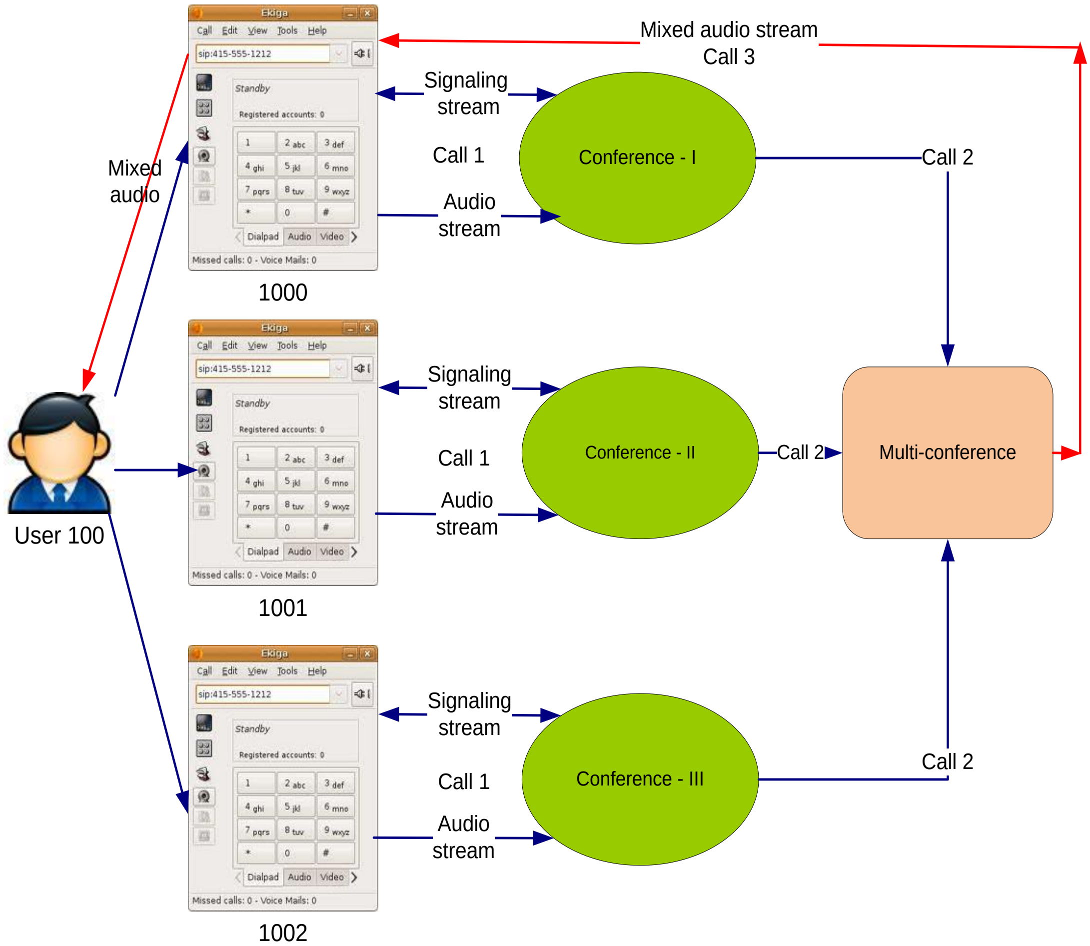

Figure 6: Signaling and call flow diagram

V. Conclusion

A. Assessment of the result

This work describes a prototype demonstration of the multi-conferencing capabilities required for space mission control room conferencing with user rights predefinition. The study proved that it is possible to implement the required conferencing system using standard open source PABX systems. The system is implemented satisfying all of the basic requirements. An alternate light weight system performing the functionalities of the existing complex proprietary system may be realized based on the output of this work.

One of the motivation for the work emphasize on simplification of the system. Reusing available basic functionalities within a standard PABX in such a way to meet the required functionality turned out to be successful. Basic call functions were reused in a way to accomplish the complex conferencing functionality.

Conferencing part of the overall system may be built without implementation, just by configuration. A simple dial plan configuration defined in FreeSWITCH is used to prototype the entire system, satisfying the multi-conferencing capabilities with all user interactions predefined in it. All the required functionalities are implemented with FreeSWITCH, without using any programming.

B. Benefits of the new approach

The simplification of the system by following a bottom-up approach will help in making a light weight system. When compared to the existing system which was developed on top-down approach, the new system would have some of the given advantages:

- More easy to operate, maintain and extend. The complexity will be much less compared to the in-use system. This reduces the skill requirement expectations.
- The scalability will be much faster and easier. The system can be connected to any number of other systems enabling maximum users to participate in conferences.
- System demands much less capital cost and implementation cost.
- For a beginner in the field, it is much easier to master the system and makes the work easier and doesn't demand intense expertise in the field.

Appendix A

Acronym List

GSOC	German Space Operations Centre
IP	Internet Protocol
IVR	Interactive Voice Response
LEOP	Launch and Early Orbit Phase
PABX	Private Automatic Branch Exchange
PBX	Private Branch Exchange
PTT	Push To Talk
RBAC	Role Based Access Control
RTP	Realtime Transport Protocol
SIP	Session Initiation Protocol
TCP	Transmission Control Protocol
VoCS	Voice Conferencing Subsystems

Appendix B

Glossary

Dial-plan	Dial-plan is a list of actions to perform depending upon what digits are dialed.
Gateway	In computer networking, a gateway is a node (a router) on a TCP/IP network that serves as an access point to another network.
IP	IP (Internet Protocol) is the primary network protocol used on the Internet. IP supports unique addressing for computers on a network.
Multi-mission	Multiple operations in the same environment
RTP	RTP (Real Time Protocol) is a protocol designed to handle real-time traffic over internet.
SIP	SIP (Session Initiation Protocol), is a signaling communications protocol. It is widely used for controlling multimedia communication sessions such as voice and video calls over Internet Protocol (IP) networks. SIP is an important protocol of consideration as this work uses Voice over IP.
Soft-switches	Soft-switches are software based platforms which can interconnect several call lines and manage traffic that contains a mixture of voice, fax, data and video. It is software installed on a computer.
Soft-phone	A piece of software installed on a computer to allow the computer to provide real time audio communication service is called a soft-phone. A softphone is a software program for making telephone calls over the Internet using a general purpose computer, rather than using dedicated hardware.

References

[1] Anthony Minessale, Darren Schreiber; Michael S Collins; Raymond Chandler*: FreeSWITCH Cookbook.* Release Date : February 2012

[2] Anthony Minessale; Darren Schreiber; Michael S Collins; Raymond Chandler*: FreeSWITCH1.2 – Build roburst, high performance telephony systems using FreeSWITCH.*

[3] CCSDS Secretariat, Space Communications and Navigation Office: *Voice and Audio Communications- White Book*. Issue 0, April 2012.

[4] Emily S. Patterson; Jennifer Watts-Perotti; David D. Woods: *Voice Loops as Coordination Aids in Space Shuttle Mission Control,* Cognitive Systems Engineering Laboratory, Institute for Ergonomics, The Ohio State University, December 2 1998.

[5] Col-CC Sustained Engineering Team: *Columbus –CC Voice Conferencing Subsystem Replacement.* Issue1, Revision 1, November 2009

[6] Markus Töpfer: *Prototyping an Event based Distribution Model for Multicast based Multi-conferencing Parameters with SIP*. August 3, 2009.

[7] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston. J. Peterson. R. Sparks, M. Handley, E. Schooler: *SIP: Session Initiation Protocol.* [RFC 3261] (Standard)

[8] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson: *RTP: A Transport Protocol for Real-Time Applications.* [RFC 1889] (Standard),Version: January 1996 (Requests for Comments).

[9] R. Even, N. Ismail: *Conferencing Scenarios*. (Informational), Version: July 2006

[10] VoIP Softswitch: Important Aspects of SIP Based Softswitch. http://www.articlesbase.com/voip-articles/voip-softswitch-important-aspects-of-sip-based-softswitch-6151059.html. [Cited: 21- May-2013].

[11] *Understanding SIP based VoIP:* http://www.packetizer.com/ipmc/sip/papers/understanding_sip_voip/. Version 1.0, [Cited: 12-June -2013].

[12] FreeSWITCH wiki Main Page: https://wiki.freeswitch.org/wiki/Main_Page , [Cited: 19- April-2013].

[13] Dialplan XML: http://wiki.freeswitch.org/wiki/Dialplan_XML , [Cited: 2- May-2013].

[14] Mod_conference : http://wiki.freeswitch.org/wiki/Mod_conference , [Cited: 15- May-2013].