Easy Accessible Voice Gateway between Mbone and ISDN/PSTN Networks

Linqing Liu, Torsten Braun

Institute of Computer Science and Applied Mathematics, University of Bern

Neubrückstr. 10 CH-3012 Bern, Switzerland

{liu|braun}@iam.unibe.ch

Abstract—This paper presents a solution for setting up an easy accessible voice gateway between Mbone and ISDN/PSTN networks. It implements gateway control and administration mechanisms and utilizes some available public domain software to achieve necessary functionalities. With this solution, users can easily set up voice conferences with mobile WAP phones or web browsers, and join in multicast sessions using (mobile) telephones.

Index Terms—Gateway, Mbone, ISDN, PSTN, multicast.

I. INTRODUCTION

IP networks begin to play an active role in voice communication. IP telephony exploits open IETF and ITU standards to carry multimedia traffic over IP networks, offering users more flexibility. IP based telephony has the potential to significantly reduce the costs of long-distance voice communication. In addition to cost effective impacts with IP telephony, it would also be convenient to set up audio conferences over the Internet without the need of an expensive Multipoint Control Unit (MCU). IP multicast provides efficient many-to-many data distribution in an internet environment. The multicast backbone (Mbone) is a very suitable media for serving the purpose of multiparty conferences.

In this paper, we describe a solution to implement an easy accessible voice gateway between Mbone and ISDN networks based on public domain software. When users wish to join a multicast session, they can get a list of available session information from a HTML/WML web site. After selecting desired sessions and registering with the telephone number and email address etc. via a WAP capable mobile phone or a regular web browser, they get contact information such as gateway telephone numbers and session dial-in numbers. With the gateway contact information they can dial into the gateway to join the sessions. To initiate a conference, a user can create a new session via a WAP mobile phone or a web browser, and send the session information to other participants via email or SMS messages. It is also possible that the gateway calls other registered users connected with the gateway for joining the sessions.

The rest of the paper is organised as the follows: In section II we give a brief review of related work. Section III provides an overview of the proposed easy accessible solution. In section IV we elaborate the audio forwarding and support unit of the gateway, section V presents the control unit of the gateway, and finally section VI concludes the paper.

II. RELATED WORK

Many approaches related to ISDN and Mbone applications have already been developed. Some of them have been used and extended in our gateway architecture.

ISDN4Linux [12] is a set of Linux kernel modules, which consists of the main module ISDN and the actual hardware driver that controls some specific card. ISDN4linux can control ISDN cards that are connected to the PC's ISA or PCI bus. Basically, ISDN4Linux can receive and transmit data via ISDN in several ways.

An architecture and prototype implementation to allow PSTN users to participate in Mbone conferences has been introduced in [1]. The basic operation allows users to dial up the gateway with a conventional phone to join multicast sessions. The speech generation and recognition component is integrated into the solution to enable users to get session information and select session by voice.

The Audio Mbone-Telephony Gateway "AudioGate" developed in the MECCANO project [13,14] provides a dial-in interface that allows users on an arbitrary telephone network (PSTN, ISDN and GSM) to call a phone number and automatically be transferred into a pre-selected Mbone session. AudioGate uses an ISDN BRI to connect to the phone network. Upon connection setup, functions such as dynamic conference selection will be provided. As soon as a "connection" to an Mbone session is established, additional services such as user identification, muting etc. could be provided.

III. OVERVIEW

The target scenario of the easy accessible gateway is shown in Fig.1. The gateway provides a user-friendly HTML/WML

interface via the World Wide Web in addition to basic gateway functions. The gateway users can get all the necessary information regarding to multicast session information, gateway contact information etc. from web pages which can easily be accessed by a regular web browser or a WAP mobile phone before dialing up to the gateway.

The gateway users can also create new multicast sessions via HTML/WML pages by filling the form with necessary information such as session name, session description, user contact information, session duration, regional scale etc. The gateway will announce the new session over the Mbone and display the new session address and port in HTML/WML page as well as send them by email or SMS message to the users.

Here we give two examples of applications for gateway users.

First, we assume a user wishing to retrieve an existing live audio session with a mobile phone. This is a simple and straightforward process. What a user needs to do is to use a WAP capable mobile phone or any web browser to access the web page provided by the gateway, and select the session s/he wants to join. Then, s/he registers necessary user information such as telephone number and email address etc. with the gateway. The gateway will display access information including dialing number and session entry number via HTML/WML page or SMS/Email message. The user can then dial up the number and join the session by telephone. The client-server mechanism implemented in the gateway is shown in Fig.2.
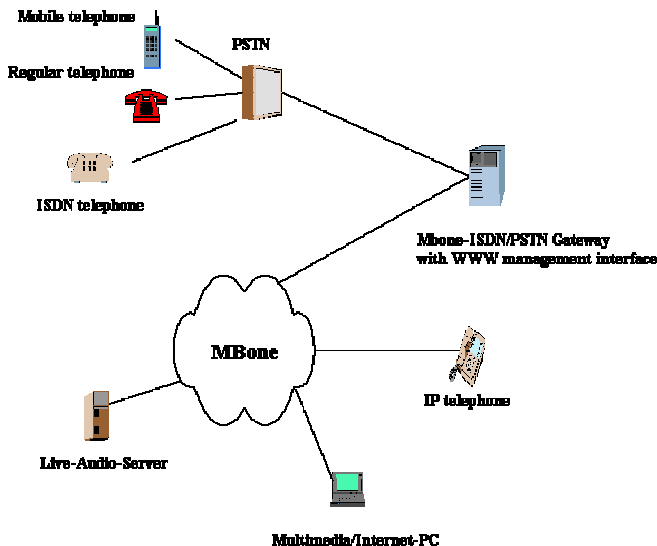

Fig. 1 Easy accessible Mbone-ISDN/PSTN gateway scenario

In the second case, the user is supposed to launch an audio conference over the Mbone with telephone access. There are two steps required for setting up the conference. The first step will be to create a new session for the conference and send the session information to other participants via email/SMS to invite them to join the conference with whatever facilities they prefer. By accessing HTML/WML pages in the gateway's web server, the user can easily create a new session over the Mbone with a WAP capable mobile phone or any web browser. The user will be asked to input session name, session description, user information, session scope, and session duration etc. The new session address/port/TTL will immediately be available upon the completion of the session creation procedure. The second step is to join the new created session which is identified by session address/port, and it will be the same procedure for this step as in the first case. The conference scenario is shown in Fig. 3.

The configuration of the gateway is required for allowing users to access the gateway with their telephone numbers to be authenticated in the gateway audio forwarding software. The available sessions' addresses and ports and TTLs as well as those session selecting access numbers should also be added to the gateway initialization files.

The main functionality of the gateway can be accessed via HTML/WML pages including user registration, gateway configuration, creating new multicast sessions, and gateway management. Gateway users and system operators can access the gateway very easily.

The gateway management and initialization can be done via a HTML interface for the gateway operator. The gateway system administrator can configure the gateway from the web.

The internal gateway architecture is depicted in Fig.4. It consists of two units: the gateway control unit and the audio forwarding and support unit. The audio forwarding and support unit is responsible for converting audio data between Mbone and ISDN/PSTN networks, listening to existing Mbone sessions and announcing new sessions over the Mbone. The main functionalities for this unit are realised by integrating public domain software such as AudioGate[15], mSD[10], and mAnnouncer[11]. The gateway control unit is implemented to achieve easy accessibility of the gateway configuration and initialization. Also accounting can be supported. The detailed information of this unit is described in section E.

IV.   AUDIO FORWARDING AND SUPPORT UNIT

The audio forwarding and support unit (AFSU) carries out the basic functionalities of the gateway including converting audio
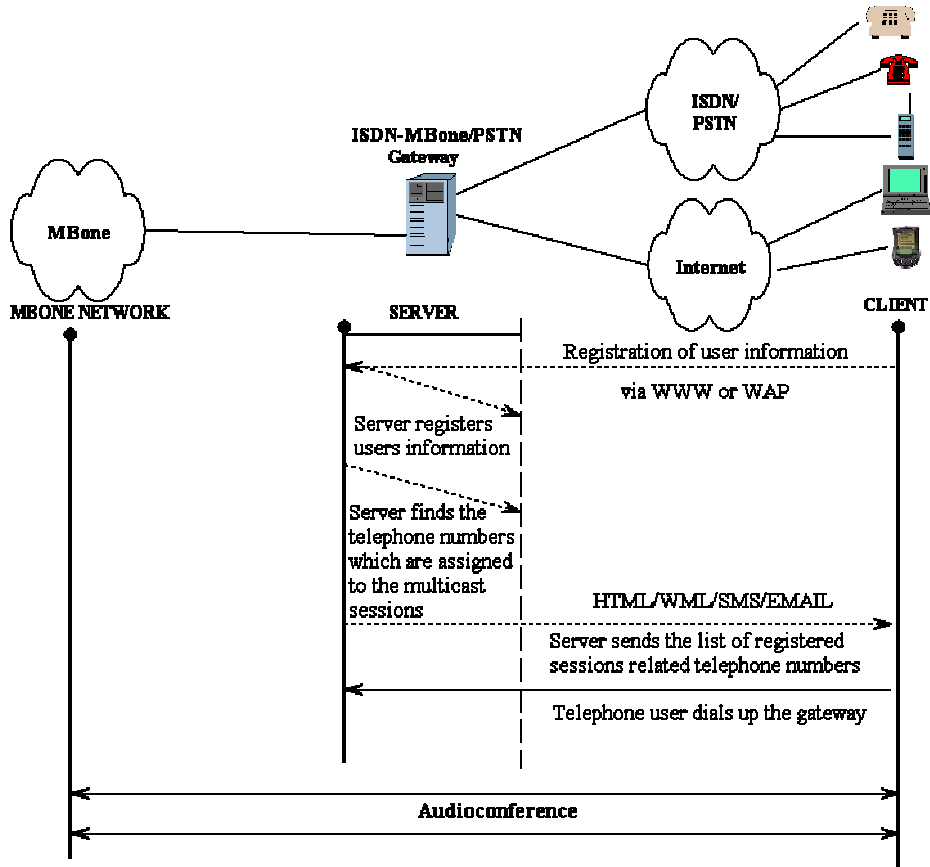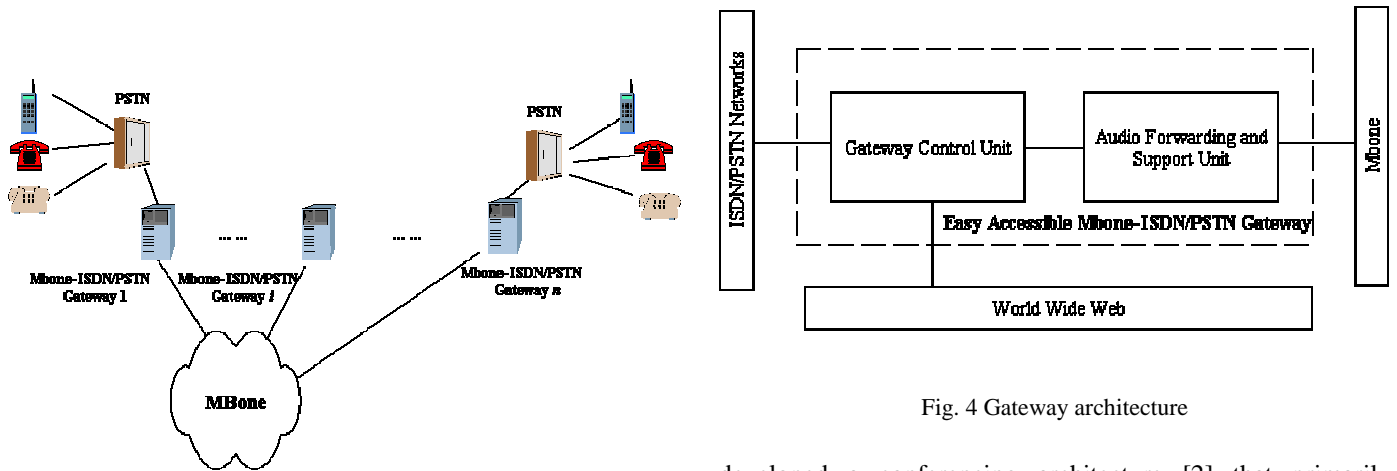
Fig. 2 Client-server mechanism

Fig. 3 A scenario for audio conference over Mbone using Mbone-ISDN/PSTN gateway

Fig. 4 Gateway architecture

data between Mbone and ISDN/PSTN networks, listening to multicast session information and announcing new sessions over the Mbone.

The teleconferencing applications in Internet and ISDN/PSTN networks are currently using different standards. The IETF has developed a conferencing architecture [2] that primarily supports loosely coupled teleconferences in the Internet while the ITU-T published series of Recommendations for more tightly coupled multimedia communications in various networks including ISDN (H.320) and the Internet (H.323). To enable the communication between ISDN/PSTN and Internet multicast backbone (Mbone), a gateway with audio forwarding mechanisms to achieve interoperability for different systems is fundamental.

Announcements are delivered to the audio forwarding and support unit to be multicasted. The Mbone announcements
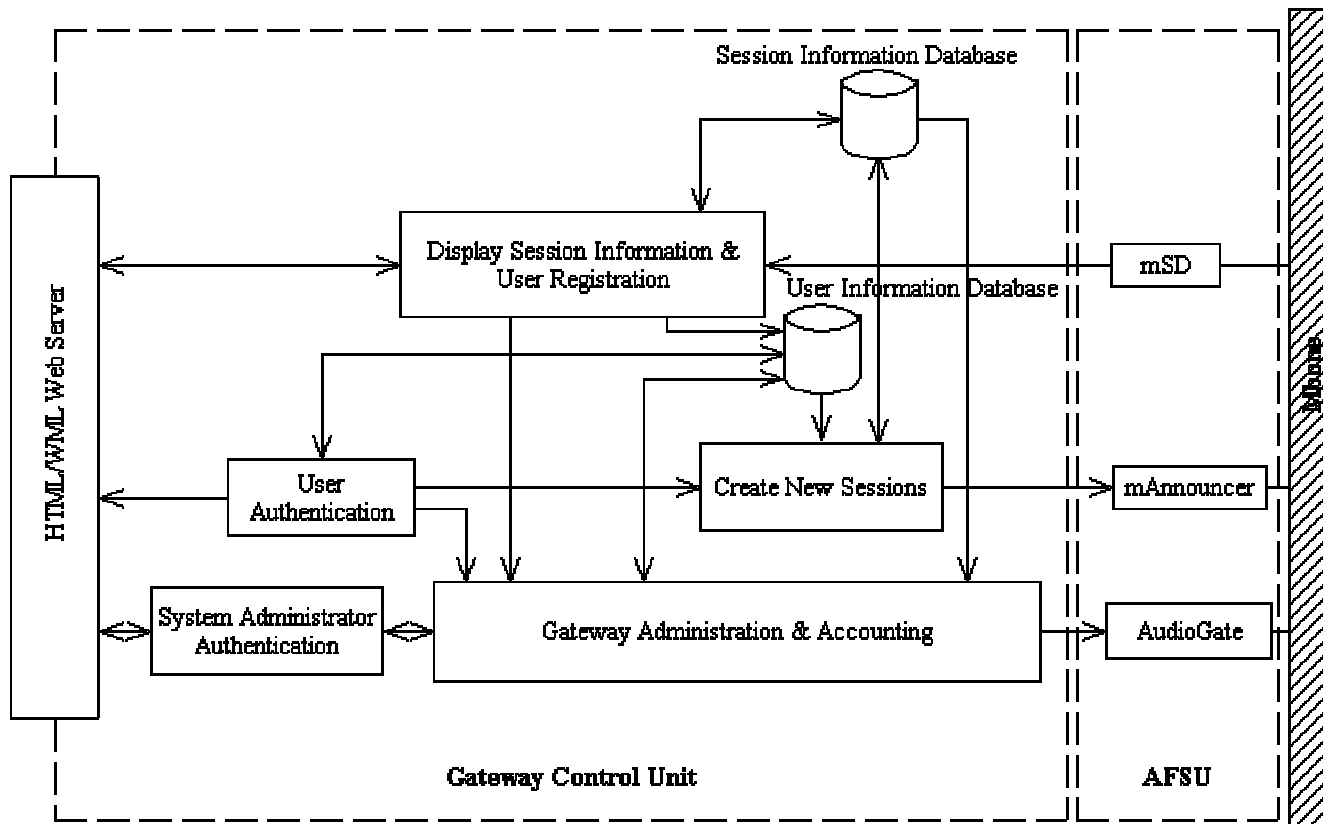
Fig. 5 Gateway control unit

are sent using the Session Announcement Protocol (SAP) with contents specified using the Session Description Protocol (SDP) [5,6]. The multicast group address assigned to Mbone session announcements is 224.2.127.254 that is known as sap.mcast.net via the Domain Name Service (DNS), the well-known UDP port 9875 is used.

The MECCANO project [13] has developed various gateways for PSTN, ISDN and GSM users to realize audio forwarding mechanisms between Mbone and ISDN/PSTN, and the software is available in the public domain. In our solution, AudioGate from MECCANO has been selected to serve for establishing connections between Mbone and ISDN/PSTN. It can be integrated into the solution for the implementation of an easy accessible gateway, which provides users access to multicast audio sessions via arbitrary telephone networks.

Basically, AudioGate provides a dial-in point for users on any telephone network and enables them to participate in Mbone conferences. The main features include [15]:

- Calling users can be authenticated based upon their Calling Line Identification (CLI), i.e. the telephone numbers they are using when making the call. AudioGate can selectively allow or disallow access to

certain users, as well as to users that do not provide the calling party number information.

- For users that do provide a calling party number, AudioGate allows to assign a full name to each of these numbers for presentation in the Mbone session.
- AudioGate provides a simple (easily configurable) voice menu for prompting callers for a conference selection.
- Conferences can either be explicitly selected through the voice menu or callers can automatically be put into a particular conference. Optionally, this may depend on the phone number they dial. The mode of operation is controlled through the configuration file.
- AudioGate imposes no artificial limits on the number of simultaneous calls it can handle. Besides processing power for audio coding, the only practical limit is the number of B channels that are available. AudioGate supports one or more ISDN BRI boards.
- AudioGate supports up to nine simultaneous conferences. The calls from the telephone network may all lead into the same conference or into different ones.
- AudioGate can be configured to use any number of ISDN Multiple Subscriber Numbering (MSN). Some of

them can be directly mapped into a conference while others may first lead into a voice menu.

- AudioGate supports muting/unmuting callers using Dual-Tone Multi-Frequency (DTMF) for control. The initial status for the callers can be configured in the AudioGate initialization file.
- All codecs and transmission modes (particularly including redundancy coding) supported by Robust Audio Tool (RAT) [16] are supported by AudioGate.

The configuration of AudioGate is initialized based on simple format ASCII files which will be managed through the gateway control unit described in the next section.

The Mbone session information are fetched with mSD[10], a public domain software which can be integrated in the audio forwarding and support unit as a daemon program. mSD generates a series of session information files in its cache directory that can be transformed into HTML/WML pages with Perl programs at the gateway's web server.

For session announcement, we exploit mAnnouncer [11] to send out new session information over the Mbone. mAnnouncer is a Java application available in the public domain for sending out session announcements periodically using the Session Announcement Protocol [6] and the Session Description Protocol (SDP) [5]. Comparing with other available tools for session announcement such as SDR [14,15], mAnnnouncer interacts more convenniently with Perl programs which we use for building the gateway control unit.

V.   GATEWAY CONTROL UNIT

To achieve the easy accessibility of the gateway and other high-level gateway management functionalities, we implemented the gateway control unit on top of the audio forwarding and support unit. The gateway control unit plays a key role for providing the following functionalities:

- It maintains a database and displays the current available Mbone sessions via HTML and WML pages. The database is automatically updated upon the request from the gateway users via HTML/WML interface.
- It provides a registration mechanism via HTML and WML pages and maintains a database for registered users.
- It displays session specific contact information for users via HTML/WML/SMS/Email.
- It provides the mechanisms for creating new multicast sessions via HTML and WML pages.
- The AudioGate configuration is controlled via HTML pages and the AudioGate settings are updated automatically according to the session database and user information database.
- It provides gateway administration functionalities including gateway configuration and accounting.

The main functionalities of the gateway control unit and its internal structure of the implementation are shown in Fig.5. This unit has three main blocks in addition to two databases for realizing user registration, creating new sessions, and gateway administration.

Basically, the existing Mbone session information is saved in the Session Information Database, which serves for displaying session information, registering user information with feedback of available session contact information, and checking existing session addresses/ports when generating new session addresses/ports.   The user information database will be used to further processing of user-selected sessions, user authentication, and billing purposes.

A. Mbone session selection and user registration

When gateway clients wish to join multicast sessions, they have first to contact the gateway web server via a conventional web browser or a WAP mobile phone, to get a list of current available multicast sessions. Then, they register to the gateway with their telephone numbers and email addresses etc. for enabling the gateway to allow them to call in with their Calling Line Identifications (CLIs).

In order to display current available multicast sessions, a set of Perl programs have been implemented to fetch cache files generated by the daemon program mSD running on the server and to transform them into appropriate HTML and WML pages together with user registration forms. The program is capable to generate multicast session information which will be selected according to the keywords entered by user via HTML/WML pages.

The output is sent out to the users' web browser or WAP capable mobile phone for further registration. The session information will be saved in the session information database for providing session addresses/ports/TTLs to generate the AudioGate initialization file. Since the session information is dynamically changed and updated all the time, it is necessary to update the session information database frequently.

In the user registration form appearing as HTML/WML pages, there will be a list of multicast sessions including the session names and session information. The users can tick the box corresponding to the desired sessions and input their names, telephone numbers and email addresses. All the selected session information including session addresses, ports, TTLs as well as the users' personal information will also be saved in the user information database.

The user information database will be automatically updated whenever a user registers a new session or changes user information. A snapshot of the user registration with a conventional web browser is shown in Fig.6, which shows a selected list of multicast sessions containing user-defined keywords together with user registration form in HTML pages. Fig. 8 shows a list of available sessions from a WAP mobile phone after user entering some keywords.
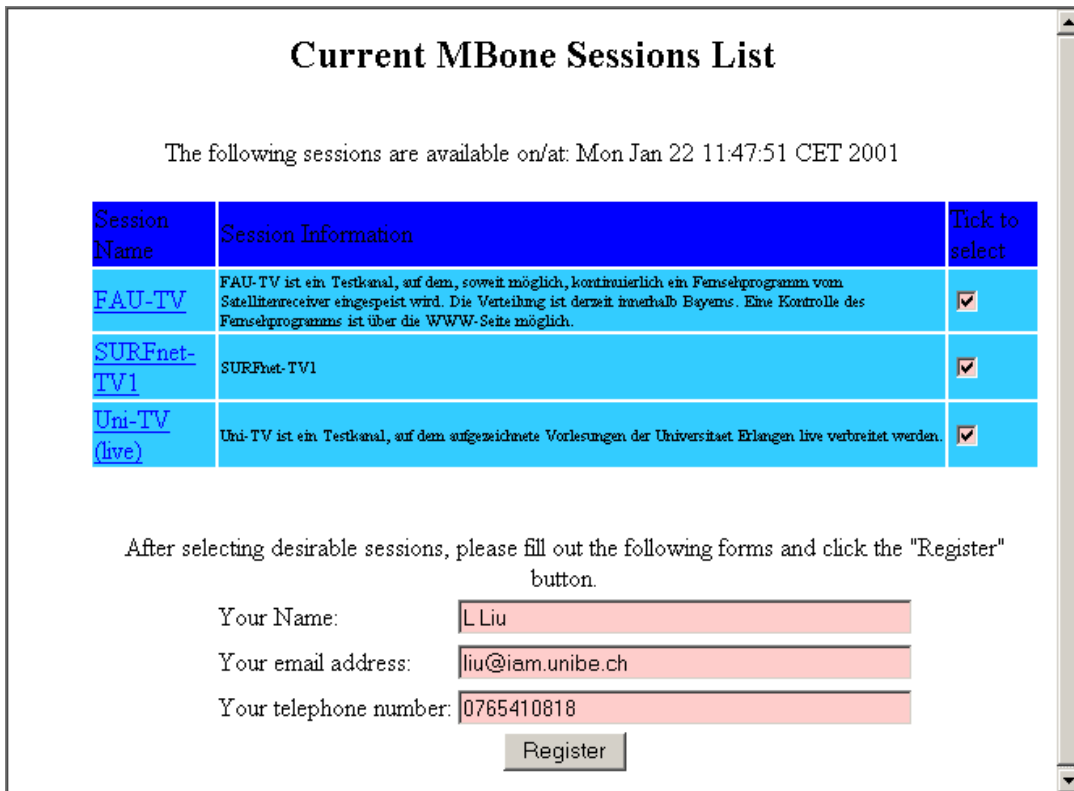
Current MBone Sessions List

The following sessions are available on/at: Mon Jan 22 11:47:51 CET 2001

Session Name	Session Information	Tick to select
FAU-TV	FAU-TV ist ein Testkanal, auf dem, soweit möglich, kontinuierlich ein Fernsehprogramm vom Satellitenreceiver eingespeist wird. Die Verteilung ist derzeit innerhalb Bayerns. Eine Kontrolle des Fernsehprogramms ist über die WWW-Seite möglich.	☑
SURFnet-TV1	SURFnet-TV1	☑
Uni-TV (live)	Uni-TV ist ein Testkanal, auf dem aufgezeichnete Vorlesungen der Universitaet Erlangen live verbreitet werden.	☑

After selecting desirable sessions, please fill out the following forms and click the "Register" button.

Your Name: L Liu

Your email address: liu@iam.unibe.ch

Your telephone number: 0765410818

[Register]

Fig. 6 Display available Mbone sessions and user registration in a conventional web browser

Thanks for using our service, please find the information below:

We have received the following sessions you would like to join:

for session of **FAU-TV**(239.192.139.43/3456/47), please dial gateway number at 0316318668 and press "1" after connected;
for session of **SURFnet-TV1**(233.4.79.2/4640/63), please dial gateway number at 0316318668 and press "2" after connected;
for session of **Uni-TV (live)**(224.2.156.100/3456/47), please dial gateway number at 0316318668 and press "3" after connected;

Thank you!

We have found you sent us the following session address before:

test paris-nancy(224.2.220.84/32502/63)
conference test from Bern(231.142.216.114/29786/6)
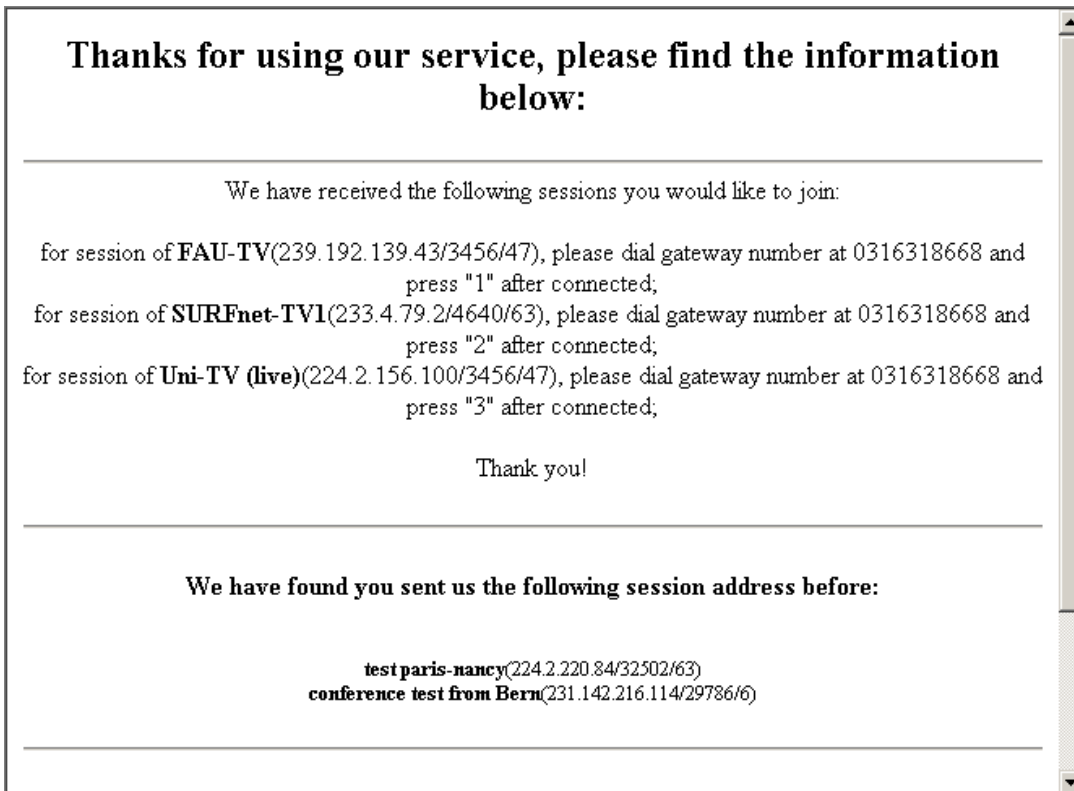
Fig. 7 Display available Mbone sessions contact information in a conventional web browser

(a)   (b)

Fig. 8 Search available Mbone sessions in WAP mobile phone

Once users complete the registration process, the following HTML/WML pages will display all the necessary information for contacting the gateway, which consists of gateway telephone numbers and extension numbers for selected sessions. The same information can also be sent out automatically to the users as email or SMS. The users' information stored in the user information database will be used for user authentication and other user specific service such as user account information etc.

Fig. 7 shows the selected sessions contact information following the session selecting and user registration shown in Fig. 6. It is clearly shown that the users could then follow the contact information provided here and dial up the gateway to join the sessions.

If the sessions that users have selected are not assigned to the gateway, a message will automatically be sent to the gateway administrator including the unassigned session information. When the gateway administrator completes the session assignment procedure, the users can receive the session contact information.

B. Create new multicast sessions via HTML/WML

Creating a new multicast session is crucial for user-initiated conferences. It is convenient for users to be able to create new multicast sessions via HTML/WML for their own conference. To announce a new session in Mbone, a SDP[5] file needs to be generated according to session information. SDP is a textual description, which describes the formats, protocols, contents, timing, etc, used in the session. A session description consists of a session-level description (details that apply to the whole session and all media streams) and optionally several media-level descriptions

(details that apply onto to a single media stream). An announcement consists of a session-level section followed by zero or more media-level sections.

When SDP is conveyed by SAP, only one session description is allowed per packet. When SDP is conveyed by other means, many SDP session descriptions may be concatenated together (the `v=' line indicating the start of a session description terminates the previous description). Some lines in each description are required and some are optional (marked with a `*' below) but all must appear in exactly the order given below (the fixed order greatly enhances error detection and allows for a simple parser) [5].

Session description:

    v= (protocol version)
    o= (owner/creator and session identifier).
    s= (session name)
    i=* (session information)
    u=* (URI of description)
    e=* (email address)
    p=* (phone number)
    c=* (connection information - not required if
        included in all media)
    b=* (bandwidth information)

One or more time descriptions (see below):

    z=* (time zone adjustments)
    k=* (encryption key)
    a=* (zero or more session attribute lines)

Zero or more media descriptions (see below):

Time description:

    t= (time the session is active)
    r=* (zero or more repeat times)

Media description:

    m= (media name and transport address)
    i=* (media title)
    c=* (connection information - optional if included
        at session-level)
    b=* (bandwidth information)
    k=* (encryption key)
    a=* (zero or more media attribute lines)

An example SDP description for a test session is shown as follows:

```
v=0
o=LLiu 3189143230 3190007230 IN IP4 130.92.66.147
s=A test session
i=Just a test.
u= http://coyote.unibe.ch/test.wml
e=L Liu <liu@iam.unibe.ch>
p=L Liu <0765410818>
t=3189143230 3190007230
a=type:broadcast
```

```
a=tool:mAnnouncer
m=audio 26210 RTP/AVP 0
c=IN IP4 224.20.146.105/63
a=ptime:40
```

We implemented Perl programs to create SDP files for user-defined new sessions. The generated SDP files are then used to trigger mAnnouncer[8] to send out the new sessions information via SAP[6] over the Mbone. In compliance with the requirements of mAnnouncer, a new line should be added on top of the SDP descriptions.

The generation of the SDP file for the new multicast session follows the requirements of Multicast Address Allocation (MALLOC) [3,4,8], SAP and SDP. The session information database and mSD cache files are used to avoid address collision since the new generated addresses/ports are selected randomly.

Fig.9 shows a set of screen snapshots for using a WAP mobile phone to create a new session. By selecting the scope of the new session (in Fig.9a) and entering necessary information such as session/user information and session duration etc. (in Fig. 9b-9f), the information of the created session including session address/port/TTL is displayed (in Fig. 9g). The new session information can then be sent out to all the participants who are invited to join the conference.

By utilizing the functionality provided by AudioGate, we enable gateway users to call other users connected to the gateway via WML/HTML page, i.e. provide "click-to-dial" for mobile and regular users to call other participants to join the sessions.

C. Gateway administration

The gateway administration includes the gateway initialization, assigning existing sessions to different entry positions from the voice menu of AudioGate, user account maintenance as well as authentication. Currently, AudioGate supports nine sessions for each access number, which is required to be assigned in its initialization file.

The solution has been implemented so that the main tasks including gateway configuration, initialization and accounting are achieved via HTML page access after system administrator authentication.

The configuration of the gateway depends on the available multicast session addresses, ports, TTLs, registered users telephone numbers. The gateway administrator can manually assign the gateway telephone numbers and session selection extension numbers to the AudioGate initialization file via an HTML page (an example snapshot is shown in Fig. 10).

It is also possible to generate the AudioGate initialization file automatically from the database of the session information together with the user information database whenever a user submits registration information. The initialization of the gateway or refreshing with a new initialization file can also

(a)
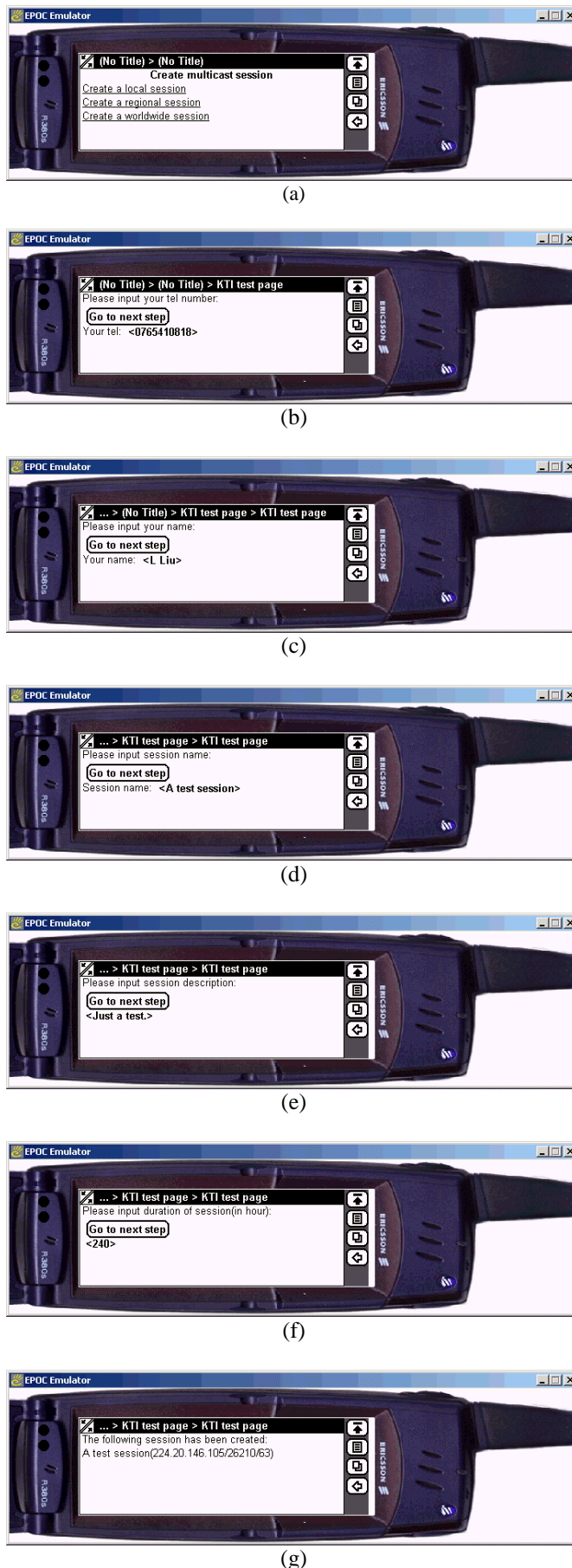
(b)

(c)

(d)

(e)

(f)

(g)

Fig. 9 Create a new session with WAP mobile phone

Gateway Administration: Assign Gateway Telephone Numbers and Session Entry Numbers

The following available sessions on Tue Jan 23 14:27:59 CET 2001 will be processed.

Note: When assigning the gateway telephone numbers and session entry numbers, please assign in the following form "gateway number/session entry number", e.g. 012345678/9 for gateway number 012345678, and session entry number 9. For user's telephone number, please use "," as separator. Ususally, you don't need to change user's telephone numbers, as they are automatically loaded from user information database.

Session Name	Audio address/port/TTL	Gateway Tel/Session Numbers	User Tel. Numbers
A test session	224.20.146.105/26210/63		
ATHENS-Renater	224.2.220.53/45380/65		
Causeries du FMBone	224.2.252.183/23966/127	0316318668/8	0765410818,0319918880
FAU - Vorlesung OR V (Lagerhaltungsmodelle)	224.2.244.231/3456/47		
FAU-TV	239.192.139.43/3456/47	0316318668/1	0765410818
Frequence Banane	224.2.171.31/18010/127	0316318668/6	0319918125,0319913445
IMJ -- Channel 1	224.2.1.1/19960/127		

Fig. 10 Assign gateway telephone number and session entry numbers.

be done in either manual or automatic ways.

Accounting which includes the cost allocation and billing mechanism is supported by setting up an accounting database, with parsing the log file of AudioGate and allocating cost table. A clip of the AudioGate log file is shown below:

... ...

```
Jan 12 08:45:42 coyote AudioGate (0.3)[8871]:
  accepting call from 59
Jan 12 08:45:42 coyote AudioGate (0.3)[8871]:
  call established
Jan 12 08:45:43 coyote AudioGate (0.3)[8871]:
  switching to session 233.4.218.44/23718/127
Jan 12 08:45:48 coyote AudioGate (0.3)[8871]:
  hanging up
Jan 12 08:45:48 coyote AudioGate (0.3)[8871]:
  call duration:(00:00:06)
Jan 12 08:45:48 coyote AudioGate (0.3)[8871]:
  switching to session 224.224.2.224/2222/127
Jan 12 08:45:50 coyote AudioGate (0.3)[8871]:
  accepting call from 59
Jan 12 08:45:50 coyote AudioGate (0.3)[8871]:
  call established
Jan 12 08:45:52 coyote AudioGate (0.3)[8871]:
  switching to session 224.2.246.13/30554/127
Jan 12 08:46:08 coyote AudioGate (0.3)[8871]:
  hanging up
Jan 12 08:46:08 coyote AudioGate (0.3)[8871]:
  call duration:(00:00:18)
```

... ...

Here it is clearly indicated that the time and the duration of the call from user 59 have been recorded, and hence they could be used for the accounting purposes.

D. Limitations and future work

The current solution has been implemented on a Debian2.2/Linux 2.2.15 platform with ISDN BRI (BRI: basic rate interface) boards. The number of lines the gateway can support depends on the number of ISDN boards in the gateway. The current version of AudioGate can only support up to nine sessions simultaneously.

Due to the limitation of AudioGate's maximum allocated sessions and inability for dynamic reconfiguring, it is difficult to get a seamless service to accommodate large number of sessions and users. It is desirable to develop a mechanism in future to provide seamless service without any interruption for current users when refreshing AudioGate with newly generated configuration file. The AudioGate's Mbus interface also provides the possibility to simplify the integration with other component in future development.

VI. CONCLUSION

This paper presents an easy accessible solution for a voice gateway between Mbone and ISDN/PSTN networks. The solution focused on the establishment of an easy accessible mechanism to control the gateway which enables mobile users as well as regular users to join or create audio conferences over Mbone via ISDN/PSTN networks. The presented solution can exploit existing resources to easily set up audio conferences while it is time consuming and complex to set up such a conference over ISDN/PSTN networks. The current solution has been implemented on a Linux platform and

makes use of some public domain software to realize necessary functionalities. The limitation for the scalability is partially relied on the available number of ISDN boards and the functionality of integrated public domain software. The latter is expected to be improved soon.

REFERENCES

[1]   R. Ackermann, J. Pommnitz, L. Wolf, R. Steinmetz, MBone2Tel – Telephone Users Meeting the MBone, *Sixth International Workshop on Interactive Distributed Multimedia Systems and Telecommunication Services (IDMS'99)*, October 12-15, 1999, Toulouse, France.

[2]   M. Handley, J. Crowcroft, C. Bormann, J. Ott, The Internet Multimedia Conferencing Architecture, INTERNET-DRAFT, July, 2000.

[3]   M. Handley, "Multicast Session Directories and Address Allocation", Chapter 6 of PhD Thesis entitled "On Scalable Multimedia Conferencing Systems", University of London, 1997.

[4]   M. Handley, S. Hanna, Multicast Address Allocation Protocol (AAP), Work in Progress.

[5]   M. Handley, V. Jacobson, SDP: Session Description Protocol, RFC 2327, April 1998.

[6]   M. Handley, C. Perkins, E. Whelan, Session Announcement Protocol, RFC 2974, October 2000.

[7]   M. Handley, D. Thaler, D. Estrin, The Internet Multicast Address Allocation Architecture, INTERNET-DRAFT, Dec 1997.

[8]   S. Hanna, B. Patel, and M. Shah, Multicast Address Dynamic Client Allocation Protocol (MADCAP), RFC 2730, December 1999.

[9]   http://www.aciri.org/malloc/

[10] http://www.cdt.luth.se/~peppar/progs/mSD/

[11] http://www.cdt.luth.se/~peppar/progs/mAnnouncer/

[12] http://www.isdn4linux.de

[13] http://www-mice.cs.ucl.ac.uk/multimedia/projects/meccano/

[14] http://www-mice.cs.ucl.ac.uk/multimedia/projects/meccano/ deliverables/d67.2/mec-d67.2.pdf

[15] http://www-mice.cs.ucl.ac.uk/multimedia/projects/meccano/ deliverables/d4.3/mec-d4.3.html

[16] http://www-mice.cs.ucl.ac.uk/multimedia/software/rat/