Multi-Context Voice Communication In A SIP/SIMPLE-Based Shared Virtual Sound Room With Early Reflections

Yasusi Kanada
Central Research Laboratory, Hitachi, Ltd.
Higashi-Koigakubo 1-280, Kokubunji, Tokyo 185-8601, Japan
kanada@crl.hitachi.co.jp

Abstract

An improved prototype of the "voiscape" voice communication medium has been developed and subjectively evaluated. Voiscape enables natural and seamless voice communication by using sound to create a virtual "sound room" in which people, who are represented by different sounds, can move freely. It features low-delay motion-tracking spatial audio with simulated early reflections that produce out-of-head sound localization and sound distance expression. It also features virtual-location-based selective communication: a user can walk freely in the sound room using a map- and cursor-key-based user-interface and can select whom to talk to or which sound sources to listen to. A third feature is SIP-presence-event-notification (SIMPLE)-based sound room management: when users move, their locations and directions are distributed using SIP SUBSCRIBE/NOTIFY messages. The combination of these features creates a natural voice-communication space in which two or more parallel conversation contexts can coexist. Limited, subjective testing by around 200 people showed that this medium can be used for cocktail-party-like conversation; i.e., users could distinguish parallel conversations by paying attention to or by moving toward one of them.

Categories and Subject Descriptors

C.2.4 [**Computer-Communication Networks**]: Distributed Systems – *Distributed applications.*

General Terms

Design, Experimentation.

Keywords

Voice communication, Spatial audio, Early reflection, Event notification, Presence, Session Initiation Protocol (SIP), SIMPLE, Auditory virtual reality.

1. Introduction

Although the telephone has long been widely used as a general-purpose voice communication medium, it severely restricts communication patterns among people compared to face-to-face communications, making natural and seamless communication difficult. For example, a telephone conversation must be initiated by "ringing", which is highly intrusive. Moreover, the conversation is basically limited to two people — adding a third person is troublesome. These restrictions result from using an interface developed 130 years ago; i.e., to talk by telephone, you must first call the person to whom you want to talk and establish a connection; you then talk to that person one-to-one using one microphone and *one* speaker and disconnect the line when you finish. The main reason this unnatural interface has not been changed is that telephone networks are hard-wired and constrained by old-fashioned standards.

There are other voice communication media such as transceivers, amateur radio, and teleconference systems. Teleconference systems do a good job of overcoming the drawbacks of the telephone, enabling conversations among three or more people at distant locations. However, they require specialized conditions and equipment, such as speaker phones, audio conference systems, and multi-point or desktop video conference systems, so they can be used in only limited situations. Moreover, they suffer two problems in particular: speaker identification and multiple talkers.

Speaker identification is often difficult in audio-only environments. Not only can it be difficult to identify who is speaking, but it can also be difficult to remember who said what. One way to solve this problem is to use spatial audio technology. If the voice of each participant appears to come from a different direction, a listener can more easily separate the voices and identify the speakers. Experiments by Baldis [Bal 01] showed that listeners can more easily identify who is speaking and remember who said what if the voice is localized. The effectiveness of using spatial audio in teleconference applications was proved by Begault [Beg 99].

The multiple talker problem occurs in most conventional systems; it is difficult to distinguish the words when two or more people speak at once. While only one person speaks at a time in formal conferences or meetings, people often talk locally, i.e., to people sitting nearby. This extraneous talking could make it difficult to hear what the main speaker is saying. In less formal situations, such as cocktail parties, parallel conversations are the norm, with various communication patterns between people, including crossover of multiple contexts. Some conference systems use sidebars [Mar 04] or side conversations [Ber 95] to solve this problem; a sidebar is a small conference within a conference. However, creating sidebars is not an intuitive method for local conversations, and it does not allow crossovers. Mark [Mar 04] experimentally showed that it takes much time to create sidebars and that people seldom used them.

A more natural and powerful method, one that incorporates features of face-to-face meetings and that solves both problems, combines virtual reality and spatial audio technologies. People enter a space, which is shared among the people, to communicate with each other. If a person moves close to another person, a local conversation can be naturally initiated. This new voice communication medium is called *voiscape* [Kan 04]. A virtual *sound room*, in which each user is represented by a spatially located sound, is created (See **Figure 1**), and the people in the room can move freely. Initial testing using a prototype showed that the sound quality was poor and that the virtual-reality interface was not refined. In addition, the prototype was not based on a general framework or standards.

A new prototype called VPII (Voiscape Prototype II) has been developed. The implementation has been improved and is standards based. Section 2 describes VPII, its architecture, interface, and features. Sections 3 to 5 describe the features in detail. Section 6 discusses the evaluation, and section 7 concludes the paper.

NOSSDAV'05, June 13–14, 2005, Stevenson, Washington, USA.

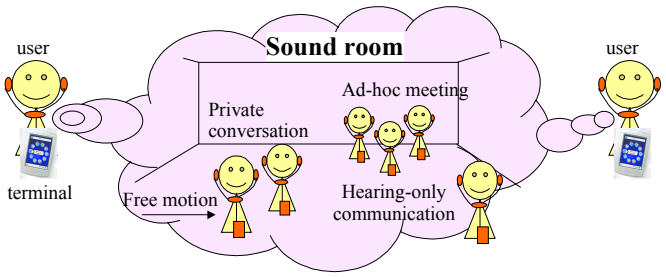

Figure 1. Sound room concept

2. VPII

2.1 Architecture

The previous version of voiscape prototype [Kan 04] had a mixed architecture: partly distributed and partly centralized. Voice streams were exchanged between terminals while SIP (session initiation protocol) [Ros 02] messages were mediated by SIP proxies. The new version, VPII, uses a typical centralized architecture to enable support for computationally weak terminal devices such as PDAs.

There are three major components in this architecture. **Figure 2** shows the protocols used between them and the message flows.

- **User Agent (UA)**: Each user terminal contains a UA, usually implemented in software. The terminal must have voice capture and playback functions, a pointing device such as cursor keys or a touch panel, a display, and an IP communication function. A wireless or wired LAN is used. A UA sends a voice stream to the 3D voice server and receives one back. It exchanges session control and presence-related messages with the room manager server. Currently, the sampling rate is 8 kHz, and the codec is ITU-T G.711, i.e., the sound has telephone quality.
- **Management Servers**: There are three management servers: a room management server (RMS), a room list server (RLS), and a SIP registrar. SIP [Ros 02] and the presence event notification mechanism [Roa 02] [Ros 04] called SIMPLE (SIP for Instant Messaging and Presence Leveraging Extensions) are used by these servers. A user selects a room from the room list distributed by the RLS. When the user enters a room, the UA sends an INVITE message to the RMS. The RMS collects users' presence information, including their location and direction of movement, manages it, and distributes it to the users. The RMS also manages the creation and destruction of sound rooms.
- **3D Voice Server (3VS)**: All the voice streams are mediated by the 3VS. It spatializes the voices and mixes the results. It receives control information from the RMS through a CLI (command-line interface) and communicates and processes voices according to the control information. The spatialization is a time-consuming computation. However, no DSP (digital signal processor) is used in VPII.

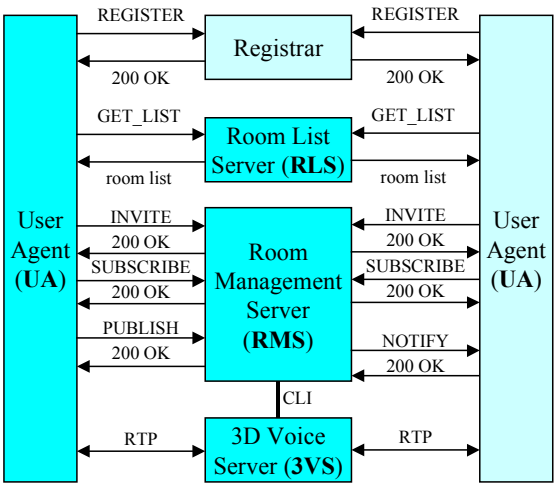

Figure 2. Architecture of VPII

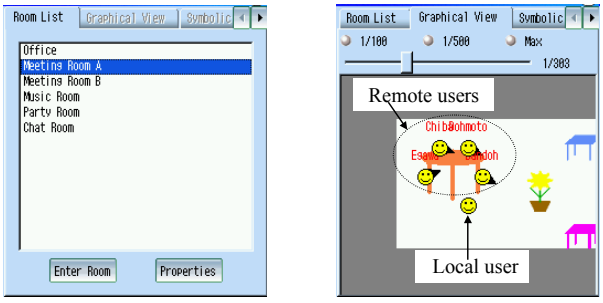

Figure 3. User interface of VPII

2.2 User interface

The user interface of a UA is illustrated in **Figure 3**. The user first selects a room to enter from the room list, shown on the left. The UA then displays a map of the sound room, shown on the right. The walls are displayed in gray. The scale of the map can be changed using a radio buttons (or slider). A unique icon can be used for each user. The orientation of the other users are displayed by arrows.

A user moves one-foot forward by pushing the forward (arrow) key and one-foot backward by pushing the backward key. The user turns left 18-degrees by pushing the left (arrow) key and right 18-degrees by pushing the right key. The user's icon is always displayed immediately below the screen center, and its orientation remains fixed. This means that the displayed wall moves downward as the user moves upward and that the room display and other users rotate right as the user turns left.

In the prototype implementation, a Sharp Zaurus (SL-B500 or SL-5600) and a Linux-based PDA were used as the terminals.[1] Qt middleware developed by TrollTech was used in the Zaurus to provide a light-weight window system and some additional functions such as XML parsing. Since Qt works on Microsoft Windows and Apple Mac OS X, software run on Qt can be easily ported between these environments.

2.3 Features

VPII has three key features.

- **Low-delay motion-tracking spatial audio**: For each user, the sounds from the other users are spatialized based on their relative locations and directions. The delay caused by spatialization is minimized (less than 1 ms) because it is used for bidirectional conversation. The sounds are attenuated based on the relative distance and filtered by an HRTF (head-related transfer function). Reflections caused by the room walls are added because they improve distance perception and prevent in-head localization. User motions are reflected in the sound in real time. Because motions are discrete, several interpolation algorithms are used to avoid click noises and to make the motions smooth.
- **Virtual-location-based selective communication**: A user turns and moves around the room to select which 3-D sound sources, which represent persons or objects, to talk to or to listen to. The virtual-reality user-interface that enables these motions is based on Hall's model of personal distance [Hal 66] and Benedikt's cyber space principles [Ben 91]. User privacy is protected by policies. Stationary objects, or "landmarks", such as tables, are added to help the users distinguish locations in the room.

[1] A microphone jack was added to the SL-B500 to enable it to be used with a headset.

- **SIMPLE-based sound room management**: Each user chooses his or her location and direction. This information, plus other user attributes and information objects in the room, must be managed and propagated. SIMPLE is used for both room and room list management; i.e., the UA sends a request for the room attributes, including those of the users and objects, to the RMS and sends a request for the room list to the RLS.

These features are explained in detail in the following sections.

3. Low-delay motion-tracking spatial audio

Spatial audio technologies [Beg 00] are used for creating virtual sound environments such as DIVA [Lok 02]. The Robust Audio Tools (RAT) [Har 96] support audio conferencing by enabling a participant to virtually position the other participants around his or her head by using a simple spatialization technique. Savior [Sav 99] described several sound interpolation techniques that enable smooth motion and prevent noise.

In VPII, these techniques are refined and integrated into a low-delay and motion-tracking spatial audio technology explained here.

3.1 8-kHz sampling rate

A sampling rate of 8 kHz is used in VPII for three reasons.

- **Reasonable communication bandwidth and delay**: Although networks are becoming wide-band, most voice communication paths are still narrow-band such as 64 kbps or less. If a codec with compression such as MP3 is used, it is possible to reduce the bandwidth even if a 22.05-kHz or higher sampling rate is used. However, such a codec increases the delay and makes bidirectional communication difficult. MPEG4 AAC LD, which is a low-delay, low-bandwidth, high-sampling-rate (48-kHz) codec, may be a solution, but it is still difficult to use in terminals, especially PDAs because it is computationally expensive.
- **Real-time signal processing**: Because an HRTF is used, if the sampling rate is higher, the signal processing for spatialization requires more computing power. The CPU time can be reduced if FFT (Fast Fourier Transform) is used. However, this increases the delay, making it difficult to incorporate interpolation algorithms. If the sampling rate is 8 kHz, time-domain signal processing does not require much CPU time, so the net delay is less than 1 ms. The signal processing in a 3VS can be reasonably processed by a general-purpose CPU, and it can also be accelerated and scaled up if DSPs are introduced.
- **Narrow bandwidth of voice**: Listeners localize sounds from above or below based on 6–12-kHz cues and localize sounds from front or back based on 8–16-kHz cues [Lan 02]. Therefore, the spatialized sound should be wide-band if the original sound contains such high frequency components. However, the human voices does not have large high frequency components. Therefore, if the purpose of the system is to transmit human voices, a high sampling rate probably does not have sufficient effect on vertical sound localization.

A down-sampled version of Gardner's HRTF sampled at 44.1 kHz [Gar 94a] is used in VPII. A Chebyshev filter in Matlab [Mat 00] was used for this down-sampling.

3.2 Reflections

Room reverberations consist of two components [Gar 94b] (See **Figure 4**).

- **Early reflections**: After hearing a direct sound, the listener hears tens of reflections from the walls, ceiling, and floor within 100 ms or so. They are called early reflections.
- **Late reverberations**: Because the sounds are repeatedly reflected and diffused, and the number of reflections is huge, a listener cannot hear each reflected sound separately. These sounds form late reverberations. They are usually modeled using exponentially attenuated sounds with randomized directions and phases.

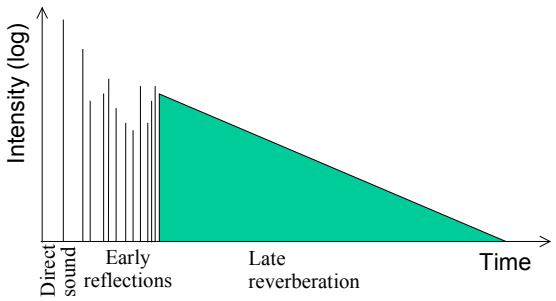

Figure 4. Structure of reverberation

In VPII, the reflections of the room walls are simulated using an image source method [All 79] while the late reverberations are not for the following reasons.

- **Out-of-head localization**: If no reverberations are added, the sounds tend to be localized in the listener's head. This unnatural situation can usually be resolved by adding reverberations, especially by adding early reflections [Beg 00].
- **Distance perception caused by early reflections**: Distance perception is believed to be based on the R/D ratio, i.e., the intensity ratio of indirect (reverberation) and direct sounds [Beg 00]. In a room, the R/D ratio increases when the distance from the sound source increases. Bronkhorst [Bro 99] showed that hearing only the first three to nine simulated early reflections makes the perceived distance sufficiently long.
- **Sound clarity**: Late reverberations reduce sound clarity. Early reflections change the color of the sound but have less effect on the clarity.
- **Amount of computation**: Late reverberations require much more computation time than early reflections.

The walls determines both the reflections and the range of user motions. A user motion is reflected in the sound in real time. Because the motion is not continuous, several interpolation algorithms are used to avoid click noises and to make the motion smooth.

The early reflections are computed using a 2-D image source method (see **Figure 5**). The ceiling and floor are assumed to be non-reflective, and 12 reflections off the four walls of the rectangular room are computed. The "real" sound room is at the center, and 12 mirror images surround the room. Each mirror image contains an image of the sound source, and the sound going straight from this source to the listener is computed. If the reflection ratio of the wall is α ($0 \le \alpha \le 1$), α^n is multiplied for the sample reflected by the walls n times. The reason the number of reflections is limited to 12 is that, if the shortest edge of a sound room is 10 m or larger, most of the reflections within 100 ms are included. However, if the sound room is smaller, more reflections reach the listener within

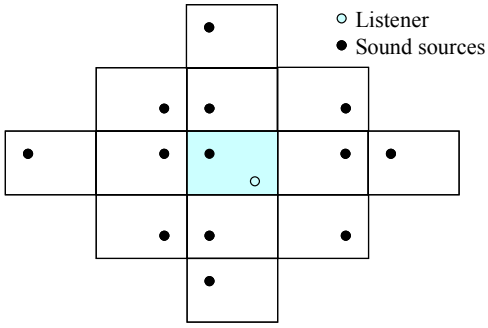

Figure 5. Early reflection computation by 2-D imaging method

this time range.

A reflection ratio of 0.7 is used for the following reasons.

- The resulting R/D ratio is sufficient for good distance perception (shown by experiment).
- Using a larger ratio would increase the indirect sounds, which could impair direction perception accuracy and reduce distance attenuation.

Because each reflected sound comes from a different direction, an HRTF different from that used for the direct sound should be used. Therefore, if convolutions of different HRTFs are applied to many reflected sounds, a huge amount of computation is required. A much simpler method is used in VPII, one that is similar to the HRTF computation used for the direct sound in RAT. Only one HRTF, which is the one for the front-direction sound, is applied to all the reflected sounds, and the difference in sounds reaching the left and right ears is expressed by applying a different ITD (interaural time difference) and IID (interaural intensity difference) to each reflected sound. This simplification reduces the amount of reflection computation to that needed for direct sound computation while still giving the direction information for the reflected sounds.

3.3 Motion tracking

There are two problems caused by user motion.

- **Click noises caused by a quick change**: If the volume and delay change quickly due to quick change in the distance to the sound source or in the direction, the user hears a click noise. In VPII, the location and direction are propagated only periodically. Therefore, setting the location and direction when they are received causes click noises.
- **Identity misses caused by a quick change**: If the sound source direction changes quickly, the user might lose the identity of the source after the change.

Three types of interpolation can be used to solve these problems: interpolation of user locations and directions, of the direct sounds, and of the reflections.

Interpolation of user locations and directions, i.e., the first type of interpolation, is illustrated in **Figure 6**. The locations of the local user, l, and the remote user, r, are adjusted immediately before the remote user's sound is spatialized by the 3VS. The location of user u ($u = l$ or r) before the adjustment is $x(u, t)$, and that after the adjustment is $x'(u, t)$. The direction before the adjustment is $\theta(u, t)$, and that after the adjustment is $\theta'(u, t)$. Time t takes a continuous value, but the spatialization starts when t is t_i ($i = 1, 2, \ldots$) and x' is defined only for this discrete time. The unit of spatialization is an RTP packet containing 20 ms of sound data. Therefore, the interval of spatialization is 20 ms, and $t_i - t_{i-1}$ is 20 ms on average, but there is fluctuation because the packet arrival time varies.

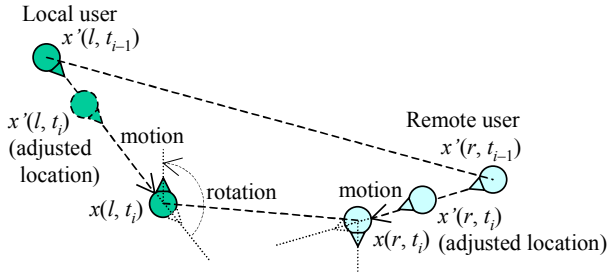

Figure 6. User location adjustment

The locations and directions are adjusted only when the motion, $x(l, t_i) - x'(l, t_{i-1})$ or $x(r, t_i) - x'(r, t_{i-1})$, is larger than a predefined value (0.1 m) or when the change in the local user's direction, $\theta(l, t_i) - \theta'(l, t_{i-1})$, is larger than a predefined value ($\pi / 72$). The location and direction are calculated by adding small values to the previous location and direction toward the current location and direction (See Figure 6). This causes motion delay. The number of adjustment steps for one motion is 20 to 30, so the delay is 400–600 ms. Because all the sound sources are non-directional in VPII, sound source directions are not taken into account.

Interpolation of direct sounds, i.e., the second type of interpolation, is done using a linear interpolation algorithm similar to Savioja's [Sav 99]. If the distance attenuation of the sound from user r at time t_i before the interpolation is $a(r, t_i)$ ($0 \le a \le 1$), the attenuation values for each sample after the interpolation are, by using $\delta = (a(l, t_i) - a(l, t_{i-1}))/N$ where N (= 160) is the number of samples, $(1+\delta)s_1$, $(1+2\delta)s_2$, …, and $(1+N\delta)s_N$. The convolution of the HRTF is computed using these values resulting in sounds without click noises. Sound source motion causes Doppler effect, but it is assumed to be negligible.

The interpolation of reflections, i.e., the third type of interpolation, is handled as follows. All the reflections are calculated using an HRTF. Two interpolations are required: the volume for each sample of the sound source is interpolated before the HRTF is convoluted, and the delay of each sample calculated by this convolution are interpolated when mixing the reflection with the direct sound.

If the first interpolation is performed before convolution of the HRTF, computation of each reflection must contain a convolution. This spoils the advantage of the reflection computation method described in the previous section, i.e., severely increases the computation time. Therefore, no interpolation is performed before the convolution in VPII. By doing so, if a user hears reflections without the direct sound, the user will likely hear click noises. However, since the reflections are attenuated, if the direct sound is added, the noise is not significant, and the sound quality is acceptable.

In the second interpolation, the volume and delay for each reflection are interpolated after the convolution using the same method used for the direct sounds. This suppresses the click noise. Although delay of a reflection should change dynamically as the user moves, and there should be Doppler effect, this effect is not simulated. That is, a moving user hears the same reflections as if the user were not moving. However, the direct sound does change, which causes a contradiction. The effect of this contradiction has not been determined, but no serious problem seems to come from it.

4. Virtual-location-based Selective Communication

OnLive Traveler [DiP 02] and its successor, Digital Space Traveler (http://www.digitalspace.com/traveler/index.html), enabled spatial-audio-based conversation in a virtual environment. In a virtual sound environment such as DIVA or Digital Space Traveler, users can move freely in the environment.

The auditory virtual-reality interface of VPII is simpler and has a more symbolical visual display than that of the above systems. It has the following features.

- **2-D map view**: In voiscape, the auditory display is the main display, and it is immersive, i.e., each remote user virtually occupies a location around the local user, and the visual display is used only supplementary. A front-view 3-D graphics display was used in the first prototype [Kan 04], while a 2-D map view is used in VPII. Although a 3-D display is good for certain purposes, using a 2-D display probably makes it is easier to find the corresponding voice and icon, and it is easier to identify the direction and distance to a remote user.[1] The user can adjust the scale by using a radio button (or slider). The user should have a mapping between the auditory display and the 2-D display, and the user

[1] A 3-D view can be easily misunderstood to be the main display, i.e., the remote users are understood to be in the 3-D view instead of being in the auditory display. In addition, a 2-D view consumes significantly less CPU time, which is particularly important for mobile terminals.

should learn that.[1]

- **Icons and landmarks**: In addition to the room users, stationary objects, or "landmarks", such as tables or plants can be placed in sound rooms. The users and objects in the room are represented by icons and the user names can also be displayed on the map. The orientation of icons and names always remain fixed on the display (See Figure 3). Landmarks can be used for indicating places in the room. If there are no stationary objects in the room, it is difficult to identify the orientation of the room and difficult to specify a place where users can meet.
- **User-motion control**: A user can move in the room by using cursor keys or a touch pad. (See Section 2.2.) A short push of a forward or backward cursor key moves the user forward or backward by a foot (30 cm), and a long push continuously advances the user to the object or wall in front of the user. No warping (i.e., a direct motion to a specified location) is allowed based on Benedikt's cyber space principles [Ben 91].
- **Distance-based communication and awareness control**: Each user is surrounded by a circular area called an *aura*. The aura corresponds to the concepts of aura and nimbus in MASSIVE [Ben 93]. If a remote user comes into the aura, the local user is made aware of this by hearing the remote user's auditory icon, and the remote user is made aware of this by a warning sound (or by hearing the local user's auditory icon). In real-world situations, the distance between persons is very important in communication [Hal 66]. However, the distance might not be as important in a virtual sound world.
- **Privacy protection**: Protecting privacy is very important in a virtual environment that many people can enter. Therefore, distance-based policies, including connection and disconnection policies, are implemented in VPII [Kan 04].

5. SIMPLE-based Sound Room Management

The management servers manage the room list and each room including the presence management of users and objects in the room. The most important task among these management tasks is room management. Each room is managed by an RMS. Room management includes room membership management, so it is similar to the management of chat rooms or mailing lists.

A proprietary protocol based on a Java object stream was used in the first prototype of voiscape. However, standard-based protocols are used in all the VPII management servers; i.e., SIP, its event notification mechanism [Roa 02], and the presence event package [Ros 04], which is a part of SIMPLE, are used.

5.1 Three types of messaging

There are three types of messaging between the UA and the management servers.

- **Room entrance and exit**: When a user enters a room, the UA sends an INVITE message with an SDP (session description protocol) message that contains the IP address and port of the UA for VoIP communication to the RMS, and the RMS usually sends a "200 OK" reply with an SDP message that contains the IP address and port of the 3VS for VoIP communication. When the user exits the room, the UA sends a BYE message to the RMS, and VoIP communication is closed.
- **Room presence management**: When the presence of a user or object in the room changes, the UA sends a PUBLISH message to the RMS [Nie 04]. The RMS stores the presence and sends the updated presence of other users by using a NOTIFY message. The UA requests the room presence information, which includes the presence of users and objects in the room, to the RMS by sending a SUBSCRIBE message. The minimum interval for presence notification should be 5 seconds because RFC 3856 [Ros 04] required this. Creation, modification, and deletion of a room are also handled by the RMS.
- **Room list management**: The UA also sends a request for subscribing a room list to the RLS (room list server), and the RLS replies by sending a NOTIFY message. This protocol can be used as an event notification, but it is currently used as a request-reply protocol. The SIP event notification mechanism can be used in this way.

SIP and SIMPLE are used for three reasons.

- **Standard protocol**: SIP is a standard and promising protocol for real-time communication control, especially for bidirectional communication. Using SIP will enable voiscape functions to be merged with IP telephony and conventional conferencing functions.
- **Flexibility**: SIP and SIMPLE are flexible enough to support voiscape functions, including the three types of messaging described above.
- **Economy**: While other standard protocols could be used for some of the servers (HTTP could be used for room list management, for example), using SIP reduces the complexity of the concepts and the implementation.

5.2 Presence message examples

It is not possible to explain all types of messaging in detail here. However, two examples of presence documents are shown here.

User presence is expressed using an extended PIDF (Presence Information Data Format [Sug 04]) document. Presence is regarded as a status and is thus indicated by `status` tags (`<status>` and `</status>`). While a status can be changed easily, presence, in a general sense, contains properties that are not easily changed. Although it is not necessary to propagate unchanged properties every time a presence message is sent because of a status change, they must be propagated the first time the user or object appears. Such properties should be distinguished from the status. PIDF has tags that are parallel to the status tag, but there is no tag for properties. A new tag, `vs:property`, was thus introduced. This tag includes new tags such as `vs:type`, `vs:room-size`, and `vs:location`. The former indicates the type of entity, e.g., (sound) room, human, and monument, and the latter indicates the coordinates of the object.

Two examples of presence document fragments are shown below. The first example describes the presence of a sound room.

```
<tuple id="Office@serverdomain.hitachi.co.jp">
 <nickname>Office</nickname>
 <status><basic>open</basic></status>
 <contact>sip:Office@1.2.3.4:5060</contact>
 <vs:property>
  <vs:type>room</vs:type>
  <vs:room-size x=”50” y=”30” z=”5” />
 </vs:property>
 </tuple>
```

This tuple indicates that the entity is a room. Its identifier is `Office@serverdomain.hitachi.co.jp`, which is an SIP URI, and its short name is `Office`. The room is $50 \times 30 \times 5$ m. The contact address contains the IP address of the RMS. The second example describes the presence of a user.

```
<tuple id="George@userdomain.hitachi.co.jp">
 <nickname>George</nickname>
 <icon>http://hitachi.co.jp/icons/George.bmp</icon>
 <vs:auditory-icon>
  http://hitachi.co.jp/auditory-icons/George.wav
 </vs:auditory-icon>
```

[1] Because the reverberation depends on the room acoustics (in the real world), listeners should learn the relationships between the source distance and the reverberation. Experiments by Shinn-Cunningham [Shi 00] confirmed that listeners can learn these relationships.

```
 <status>
  <basic>open</basic>
  <vs:location x="10" y="5" z="0" />
  <vs:aura><vs:radius>3.0</vs:radius></vs:aura>
 </status>
 <vs:property><vs:type>human</vs:type></vs:property>
</tuple>
```

This tuple describes the user's properties and status. The identifier is `George@userdomain.hitachi.co.jp`, and his short name is `George`. The 2-D and auditory icons of George are specified by their URLs. The property shows that he is a person. The status includes George's location and direction (orientation). It also includes the shape and size of his aura: a circle with 3 m radius.

TCP, instead of UDP, is used for transmitting a presence message because the message size is usually larger than the MTU of Ethernet. SIMPLE and PIDF are computationally quite heavy, so if the status is updated frequently, presence propagation requires much resource. Although this is a problem because users can move very often, the purpose of voiscape is not to propagate motions but to support communications among people. Therefore, restricting status updates should not be a serious problem.

6. Evaluation

Around 200 people tried VPII, mostly for only 5 to 10 minutes. In general, they understood it can be used for cocktail-party-like conversations. In particular, they could distinguish parallel conversations by paying attention to or by moving toward one of them.

The features of VPII were evaluated as follows.

- **Low-delay motion-tracking spatial audio**: The spatial sound of VPII was judged to be mostly good. Several listeners wanted sounds with a wider bandwidth, but most were satisfied with the sounds produced by the 8-kHz sampling rate. Large percentage of people said that the sounds localized out-of-head and sounded like distant when their virtual location was distant. Sound localization on the vertical plane was judged ambiguously. Some felt the sound was located forward, but some others felt it was located upward or backward. One evaluator, the author, felt that the user motion and the sound change caused by the motion were sometimes unnatural, but no one else made that observation.
- **Virtual-location-based selective communication**: Because the evaluators used the VPII interface only briefly, the evaluation results are only preliminary ones.
- **SIMPLE-based sound room management**: Presence propagation was delayed several seconds by SIP messaging, and the 2-D display was delayed other several seconds by the terminal because of intensive GUI and XML processing. These delays should be shortened by, for example, partial publication and notification. However, the sound motion delay due to spatialization seemed to be tolerable.

7. Conclusion

The "voiscape" voice communication medium is being developed to overcome the problems inherent in conventional voice communication media. It combines virtual reality and spatial audio technologies. Preliminary, subjective testing of a second prototype, called VPII, has shown that it provides good spatial sound and that its SIMPLE-based management generally works well.

The auditory virtual-location-based interface requires much more evaluation and probably requires improvements. Because VPII was implemented mainly for testing and demonstration, not for actual use, it still lacks important management and security functions and operational stability. VPII needs to be improved and evaluated in actual communication among people.

References

[All 79] Allen, J. B. and Berkley, A., "Image Method for Efficiently Simulating Small-Room Acoustics", *J. Acoustical Society of America*, Vol. 65, No. 4, pp. 943–950, April 1979.

[Bal 01] Baldis, J. J., "Effects of Spatial Audio on Memory, Comprehension, and Preference during Desktop Conferences", *ACM CHI 2001* (*Conference on Human Factors in Computing Systems*), pp. 166–173, March 2001.

[Beg 99] Begault, D. R., "Virtual Acoustic Displays for Teleconferencing: Intelligibility Advantage for 'Telephone-Grade' Audio", *J. Audio Engineering Society*, Vol. 47, No. 10, pp. 824–828, October 1999.

[Beg 00] Begault, D. R., "3-D Sound for Virtual Reality and Multimedia", NASA/TM-2000-XXXX, NASA Ames Research Center, April 2000, http://human-factors.arc.nasa.gov/ihh/-spatial/papers/pdfs_db/Begault_2000_3d_Sound_Multimedia.pdf

[Ben 91] Benedikt, M. (ed), "Cyberspace — first steps", MIT Press, 1991.

[Ben 93] Benford, S. D. and Fahlén, L. E., "A Spatial Model of Interaction in Large Virtual Environments", *3rd European Conference on CSCW* (*ECSCW'93*), Milano, Italy, Kluwer, 1993.

[Ber 95] Berc, L., Gajewska, H., and Manasse, M., "Pssst: Side Conversations in the Argo Telecollaboration System", *17th ACM Symposium on User Interface Software and Technology* (*UIST 95*), pp. 155–156, November 1995.

[Bro 99] Bronkhorst, A. W. and Houtgast, T., "Auditory Distance Perception in Rooms", *Nature*, 397, pp. 517–520, 1999.

[DiP 02] DiPaola, S. and Collins, D., "A 3D Virtual Environment for Social Telepresence", *Western Computer Graphics Symposium*, 2002.

[Gar 94a] Gardner, B. and Martin, K., "HRTF Measurements of a KEMAR Dummy-Head Microphone", MIT Media Lab Perceptual Computing – Technical Report #280, 1994.

[Gar 94b] Gardner, W. G., "The Virtual Acoustic Room", Masters Thesis, MIT, 1994.

[Hal 66] Hall, E. T., "The Hidden Dimension", Doubleday & Company, 1966.

[Har 96] Hardman, V. and Iken, M., "Enhanced Reality Audio in Interactive Networked Environments", *Framework for Interactive Virtual Environments (FIVE) Conference*, December 1996.

[Kan 04] Kanada, Y., "Multi-Context Voice Communication Controlled by using an Auditory Virtual Space", *2nd Int'l Conference on Communication and Computer Networks* (*CCN* 2004), pp. 467–472, 2004.

[Lan 02] Langendijk, E. H. A. and Bronkhorst, A. W., "Contribution of Spectral Cues to Human Sound Localization", *J. Acoustical Society of America*, Vol. 112, No. 4, pp. 1583–1596, 2002.

[Lok 02] Lokki, T., Savioja, L., Väänänen, R., Huopaniemi, J., and Takala, T., "Creating Interactive Virtual Auditory Environments", *IEEE Computer Graphics and Applications*, July/August 2002, pp. 49–57.

[Mar 04] Mark, G. and Abrams, S., "Sensemaking and Design Practices in Large-scale Group-to-Group Distance Collaboration", *ACM CHI 2004 Workshop on Designing for Reflective Practitioners*, 2004.

[Mat 00] The Math Works, Inc. Using MATLAB, Version 6, 2000.

[Nie 04] Niemi, A., Ed., "Session Initiation Protocol (SIP) Extension for Event State Publication", RFC 3903, IETF, October 2004.

[Roa 02] Roach, A. B., "Session Initiation Protocol (SIP)-Specific Event Notification", RFC 2543, IETF, June 2002.

[Ros 02] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and Schooler, E., "SIP: Session Initiation Protocol", RFC 3261, IETF, June 2002.

[Ros 04] Rosenberg, J., "A Presence Event Package for the Session Initiation Protocol (SIP)", RFC 3856, IETF, August 2004.

[Sav 99] Savioja, L., Modeling Techniques for Virtual Acoustics, Helsinki University, 1999.

[Shi 00] Shinn-Cunningham, B., "Learning Reverberation: Consideration for Spatial Auditory Displays", Int'l Conference on Auditory Display (ICAD), pp. 126–134, April 2000.

[Sug 04] Sugano, H., Fujimoto, S., Klyne, G., Bateman, A., Carr, W., and Peterson, J., "Presence Information Data Format (PIDF)", RFC 3863, IETF, August 2004.