Audio Engineering Society
Convention Paper 5764
Presented at the 114th Convention
2003 March 22–25 Amsterdam, The Netherlands

User Interaction and Authoring of 3D Sound Scenes in the Carrouso EU project

Riitta Väänänen[1]

[1]*IRCAM, France*

Correspondence should be addressed to Riitta Väänänen (Riitta.Vaananen@ircam.fr)

ABSTRACT

The Carrouso project combines technologies for recording, transmission and rendering of 3D sound scenes. The rendered virtual scene includes both the sound content, and the spatial and room acoustic description of the performance space. MPEG-4 tools are utilized for encoding of this data, using the general audio coding for compression of sound streams, and the scene description tools for creating virtual 3D audio scenes. We describe the creation of the virtual acoustic space in Carrouso, carried out with the help of a room acoustics analysis software and an authoring tool. A visual representation of the virtual sound scene is also transmitted to the renderer, and it acts as a user interface allowing renderer-side scene modification via the interaction mechanisms provided in MPEG-4.

INTRODUCTION

Carrouso is an EU funded Inofrmation Society Technology (IST) project whose aim is to recreate sound performances in a remotely located place by means of virtual acoustics, using the MPEG-4 as a data transmission format, and the Wave Field Synthesis for reproducing of spatial sound [1, 2]. The Carrouso system consists of three main components: the recording, transmission, and

rendering of a 3D sound scene. In the first stage, sound sources participating the performance are recorded, and related acoustic information about the surrounding space is obtained, e.g., by measuring room impulse responses of the recording space. The recorded sounds are encoded (compressed), and the room acoustic data is converted into a format that can be used to build a virtual correspondent of the recording room at the rendering side. Sound source positions are tracked in order to give an impression about a similar spatial positioning of the sources as in the recording situation. The different components forming the sound performance (the sound, the room acoustic data, and the sound source positions) are transmitted to the rendering side separately, to maintain the possibility of renderer-side modification of the sound scene [3, 4, 1].

Outline

The aim of this article is to explain how in the Carrouso framework the 3D sound scenes and their user interfaces are produced using the MPEG-4 format. First a brief introduction is made on the interactive scene concepts in MPEG-4. Then the flow of different types of data through the entire Carrouso system is described. The authoring of Carrouso content is explained, as it is needed for adding different data components together, as well as for monitoring and modifying the scene to be transmitted. Furthermore it will be explained how this authoring tool is used for including the user interface as a part of the transmitted content. This interface also provides a visual representation of the 3D sound scene, at the same time with detailed control mechanisms for the rendering-side user to modify the reproduced, virtual sound scene. An important aspect of the user interface definition is explained, namely, that with the authoring tool it is also possible to define the user rights to scene modification at the rendering side. In other words, the author may define which properties of the scene are available for the user to view and to modify.

INTERACTIVE MPEG-4 SCENES

MPEG-4 is a standard that contains several coding methods for audio and visual data, and *scene description* tools for compositing this data into a single multimedia presentation. Thus the audio or visual data are encoded separately from the information that defines their presentation in space and time. The MPEG-4 binary format scene description language (*BInary Format for Scenes*, or BIFS) enables, among generic data compositing, the creation of audiovisual virtual worlds. This is possible via a set of graphical and sound objects (*nodes*) defined manner as those in the Virtual Reality Modelling Language (VRML) [5]. Like VRML, also BIFS contains tools for creating interaction mechanisms, which allow a user to interact with the scene objects. This means that among the transmitted data and the scene, it is also possible to send components that enable this interaction (containing, e.g., modifications of sound-related parameters or positions of sound sources in the virtual space). These interaction functionalities, together with graphical BIFS objects allow making of graphical user interfaces that provide versatile modification on the transmitted scene [6, 7, 8].

Among other functionalities, BIFS enables including 3D sound objects to virtual worlds, with associated parametric description of room acoustics. This description can be either physical or perceptual. The former means a description of a room geometry (and associated sound reflectivity data of each surface) where the sound captured by the receiver (listener) contains the direct sound, and early reflections tracked with geometrical room acoustics computation methods (such as the image-source method [9]). The perceptual description, on the other hand, relies on generating the room effect from parameters that describe the perceived impression of the acoustic space [10]. In the work described in this article, the latter approach is considered as an efficient way of transmitting, rendering, and modifying the spatial properties of Carrouso sound scenes [11]. The sounds from each live sound source are encoded, and associated with the corresponding virtual (BIFS) sound object, and at the rendering stage they are modified with a room effect that reproduces the transmitted acoustical (perceptual) parameters.

Besides the MPEG-4 room acoustics parametrization, also another way of defining room response is used in Carrouso. This method is called the Wave Field Decomposition (WFD), which is particularly meant for recreating wave fields with the Wave Field Synthesis (WFS) and loudspeaker arrays [12]. However, as the reproduction method is not defined in MPEG-4, this way of transmitting the room acoustic data is not compatible with an arbitrary MPEG-4 rendering system.

CARROUSO DATA FLOW

Figure 1 shows a generic block diagram of the flow of data in the Carrouso system, when all the components have been integrated together. On the recording side, three networked subsystems are responsible for produc-

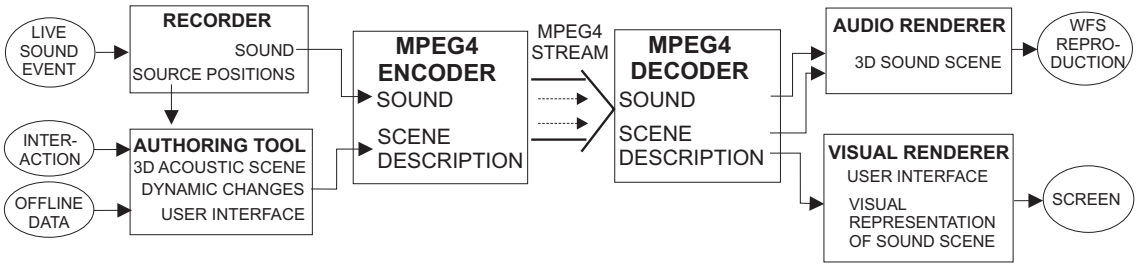

Fig. 1: Flow of data in the Carrouso system from the recoding to the rendering side.

ing a compressed, standard-format representation of the 3D sound performance, which are the recorder, the authoring tool, and the MPEG-4 encoder. On the renderer side, the MPEG-4 decoder decodes the sound and the previously mentioned scene description data, and produces audio and visual scenes, with the help of the audio and visual renderers.

Recorder, Authoring Tool, and Encoder

In the basic setup of the Carrouso system, an encoded 3D sound scene is composited of live-recorded sound, realtime-tracked source positions, and offline-computed room acoustics data. The spatial information included in this scene is gathered together in the authoring tool, which the recording-side user (the author) builds from the available room acoustic and source position data, and modifies if necessary. The sounds emitted by the sources in the performance are sent to the encoder and compressed as separate audio streams using the MPEG-4 General Audio (GA) coding methods [13]. Each sound stream is associated with a corresponding source position in the 3D sound scene, so that they can be heard coming from positions that are given by the recording system. During the live transmission of the sound performance, scene updates (MPEG-4 *BIFS commands*) can be sent from the recording side to the renderer to modify the desired parameters of the rendered scene. These updates are also a part of the MPEG-4 scene description tools, and the scene modifications may include, for example, tracked source position movements, changes in the room acoustic parameters of the virtual sound scene, and adding and removing sound sources in the performance. Along with the 3D sound scene, a 2D visual representation of the scene is created that enables the users at the rendering side to see the original source positions. This visual scene also acts as a user interface that enables local modification of the sound scene. To summarize, the encoder receives the recorded sounds from the sound recording system, and from the authoring tool it receives the spatial sound scene representation, its real-time modifications, and the associated interaction mechanisms.

Decoder, Renderer, and User Interface

At the rendering side, the listeners should ideally have a possibility to listen to a (perceptually) similar scene as the one that was recorded. The different components of the MPEG-4 stream are demultiplexed and decoded, and audio and visual scenes are rendered with corresponding renderers.

In principle the only tool that is needed for the playback of the audio and visual scenes is an MEPG-4 decoder (and renderer) compliant with the transmitted data components. However, the quality of the scene largely depends on the rendering system, or the sound reproduction device (i.e., the loudspeaker setup). This is not a normative part of an MPEG-4 decoder, as it is not meaningful to restrict the rendering to a specific technology. In the Carrouso project, the Wave Field Synthesis (WFS) is mainly considered for reproducing the sound scene, although MPEG-4 compliant sound content can be rendered with any other reproduction system. WFS is a method where loudspeaker arrays are used to reproduce a sound field for a large listening area, and for many users simultaneously, unlike in most traditional loudspeaker and headphone setups [14, 12]. The visual scene including the user interface (also transmitted to the renderer as a part of the MPEG-4 scene description data), will be displayed on a computer screen, with the capabilities for the user to affect the contents of the scene. Thus the interface appearing at the renderer is customized to the transmitted 3D sound scene content, and no external interface at the renderer is needed. An advantage of encoding the user interface in a standard MPEG-4 format, any MPEG-4 de-

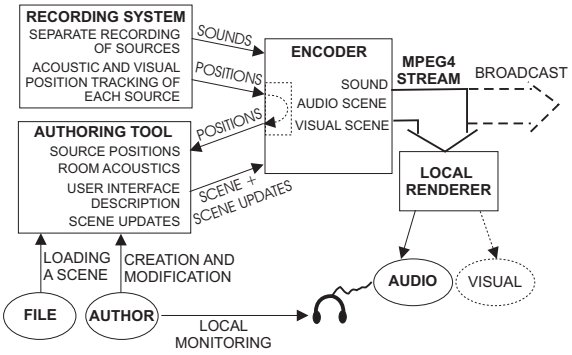

Fig. 2: Authoring of Carrouso scenes at the recording side.

coder (and not only the one created during the carrouso project) can be used to render the same 3D sound scene and the the 2D interface.

AUTHORING OF CARROUSO SCENES

Figure 2 illustrates in more detail the process of creating a 3D sound scene in Carrouso with the help of the recording system, the authoring tool, and the encoder. The authoring tool is controlled by a user through a graphical interface. It will communicate with the MPEG-4 encoder, receiving sound source related data through it, and sending a 3D sound scene data to the encoder, where it is converted to MPEG-4 binary format. In the following subsections it will be explained, how this authoring tool is used to produce the 3D sound and the 2D visual parts of the MPEG-4 scene.

Background

In another EU project called LISTEN, an authoring tool has been developed to allow defining virtual scenes in the context of audio augmented realities. This authoring tool, called ListenSpace, has been adopted also to the Carrouso framework because of the many similarities in creating virtual sound scenes in these two projects. With ListenSpace it is possible to create a parametric representation of a sound scene containing several sound sources and spatial regions with different acoustic properties [15]. Thus ListenSpace can be used to create and control (modify) such sound scenes in audio or audiovisual augmented realities, but also in pure virtual reality applications.

The composition of a Listen sound scene follows an object-based representation, which is close to the virtual sound scene representation in MPEG-4. With some modifications and additions of objects and functionalities (such as sound source objects and the associated perceptual room acoustic parameters), it can be used to produce interactive MPEG-4 sound scenes. Furthermore, it has been extended to enable storing of measured room acoustic data in an intuitive format. Like in LISTEN, also in Carrouso the authoring tool can be run on an independent computer, from which the created parametric sound scene is transmitted via a network communication to another system (in our case the MPEG-4 encoder), which takes care of merging the different (natural and virtual) data.

A screenshot of ListenSpace, with the added or modified components needed in Carrouso, is shown in Figure 3. The illustrated components are:

- Carrouso-compliant sound source object (object No. 1 in the figure), that can be used to store the room acoustic parameters as defined in MPEG-4.
- A reference listening position (object No. 2) that can be used as a sweet spot in the rendering terminal, and that corresponds to the ListeningPoint or ViewPoint node in MPEG-4 BIFS [11].
- Objects (No. 3 and 4) that represent the sound sources and the receiver in the process of measuring the room impulse responses of the performance space. In the Carrouso system an impulse response is measured for several source positions, and in the described work, MPEG-4 room parameters can be computed at each position, and applied to each virtual sound source depending on their current position with certain rules (explained further in this article).
- Conversion (button No. 5) of the ListenSpace sound scene format to a textual format which is close to the MPEG-4 BIFS (and can be used by the encoder to translate it to BIFS), and saving this data to a file.
- Communication with the Carrouso MPEG-4 encoder in order to receive the data needed for positioning of virtual sound sources, and sending the final virtual sound scene data to the MPEG-4 encoder (Button No. 6).

In addition to the above functionalities, another ListenSpace object is needed for setting correct scaling of the coordinate system. This is the reference axes object

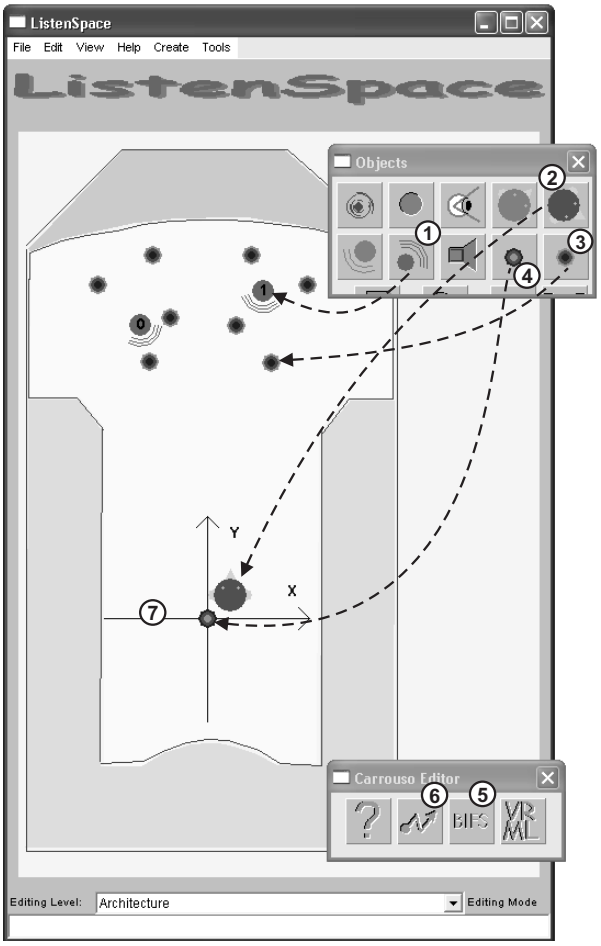

Fig. 3: Screenshot of the creation of a Carrouso scene in the ListenSpace authoring tool.

(No. 7 in Figure 3), with which the origin of the coordinate system is set to a wanted position, and scaling of the coordinates is set conveniently according to the size of the measured performance space so that it fits on the authoring tool window.

Producing 3D Sound Scene Data

The primary task of the authoring tool in the Carrouso project is to produce a 3D sound scene with positional (dynamic) sound source objects, and a room acoustic effect definition obtained from impulse response (IR) measurements of the performance space. The following will describe how a Carrouso scene is created with the authoring tool as a result of an offline process (of obtaining the room acoustics description), and a real time scene composition and modification.

Setting the offline data

Like explained earlier, the MPEG-4 perceptual room acoustic parametrization is considered when the Carrouso scene is produced with the help of the ListenSpace authoring tool. Before storing the room acoustic data in a ListenSpace (XML) file format, the perceptual parameters are extracted from the measured room impulse responses. This process is carried out in a tool developed earlier for that purpose, and the first-stage result is a time and frequency distribution of the energy of the impulse response. This data is converted to the MPEG-4 perceptual parameter set, which can be stored in the authoring tool, in the fields of the IR measurement source objects (No. 3 in Figure 3) [16]. Figure 4 shows the editor panel (”MEASUREMENT POINT PARAMETERS”) for these parameters. Thus for each measured source position the data can be given manually, and saved for a later use of the scene (e.g., for when the Carrouso performance and transmission is finally carried out). For all the IR measurement source positions, there is one common receiver position representing the microphone used in the measurements, and at the same time the origin of the system. It is also used for enabling the computation of a *reference distance*, which is one of the fields associated with each perceptual sound object in MPEG-4. It informs the renderer about the distance at which the defined parameters are valid, and at other distances the parameters are modified through a distance-dependent rolloff factor causing automatic update to the response (and the distance effect) [17, 18]. Finally, this receiver object (being unique for each scene), contains the information about how the data held in the IR measurement source positions should be taken into account when the parameters are set for the the sound source objects. Figure 4 shows 2 simple options, first meaning that the initial perceptual parameter set is the same as that in the the closest measured position, or that it is a result of interpolation between closest points. The idea in this way of setting up the acoustic scene is to have a more precise description of the spatial distribution of the perceptual parameters, than in the conventional perceptual approach, where one set of parameters is enough for describing a single room. In the absence of measured parameters, each new source contains the default set that the author can modify.

Although the impulse responses, and the derived parameters originally are computed as an offline process, they

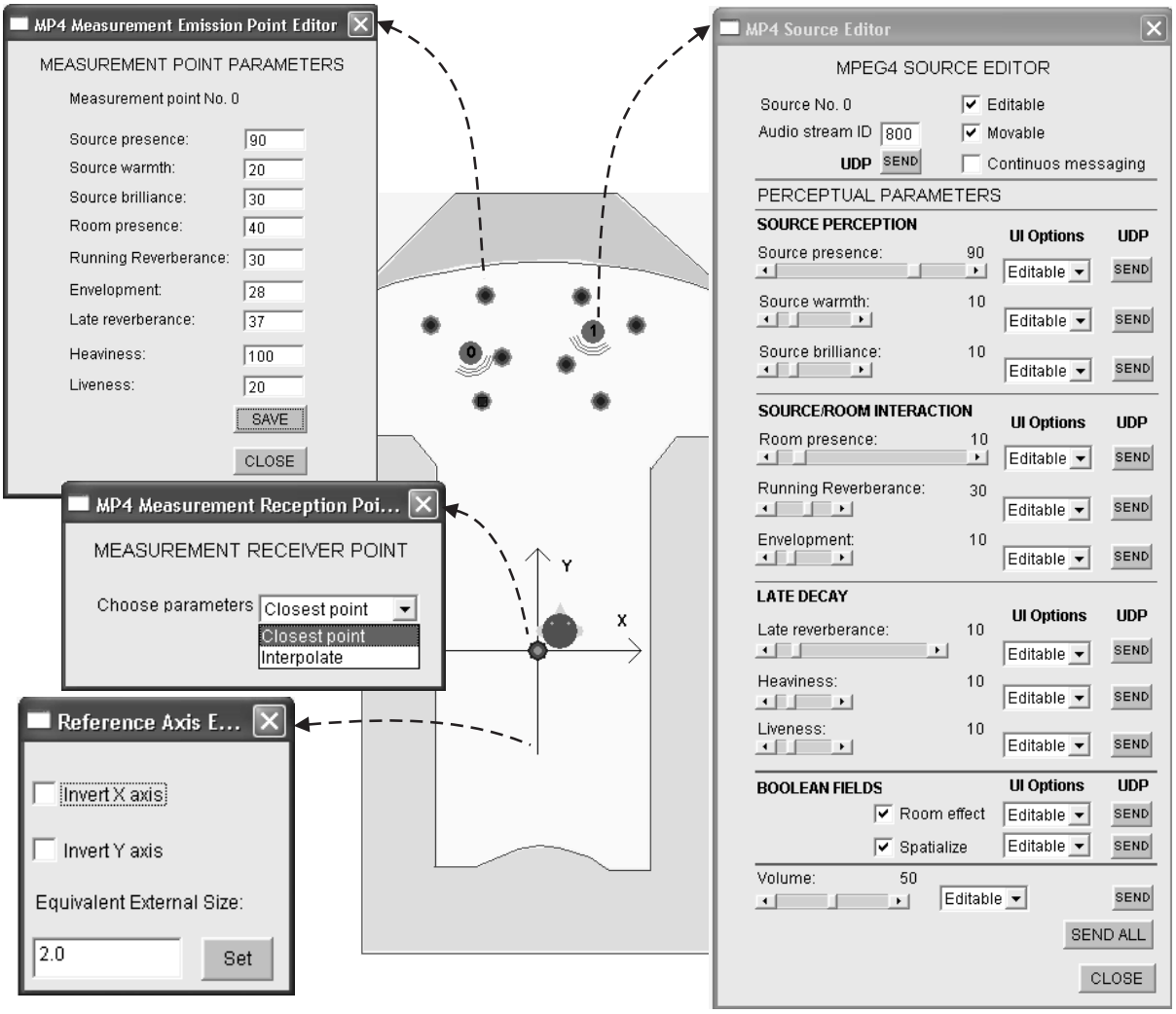

Fig. 4: Carrouso scene with editing panels for the objects.

can be dynamically changed during the rendering process, giving the possibility of changeable acoustics.

Dynamic scene creation and modification
When a Carrouso authoring process is started, an initial scene can be loaded from an XML file where the previously set perceptual parameters were stored. MPEG-4 format sound source objects are created on top of that scene according to the source data received from the recorder. Their perceptual parameters are taken from the stored data at the IR measurement positions (as defined in the IR receiver object explained above), or in the absence of those, given by default values that the author can modify.

As Figure 4 shows, each sound source object of the authoring tool scene is associated with a visual control panel that the author can open for viewing and modifying the acoustic parameters attached with the source. Each parameter has its own slider for continuous modification from the user.

When the author is ready to start encoding the scene, the listen space is converted to MPEG-4 compatible format and sent to the encoder. Local monitoring of the sound

scene is possible via an MPEG-4 decoder and renderer connected to the encoder output. This gives the possibility to listen to the scene, and the modifications made by the author, before it is finally broadcasted to the remote Carrouso renderer(s). The author modifications can be continuous (enabled by the ”Continuous messaging” checkbox in the interface), in which case a slider movement sends a series of parameter update commands to the encoder. If this option is unchecked, a single modified parameter is sent only when the author presses a corresponding UDP ”SEND” button of the interface. In each case, the encoder sends the modifications as BIFS commands explained earlier in this article.

When the author starts broadcasting the 3D sound performance to remote decoders, the transmitted scene description data is first used by the decoder to initialize the spatial sound rendering process. After this, the incoming sound streams are played through spatial processing controlled by the room parameters of the scene. Changes to the room acoustic parameters and source positions may be done by the user (like will be explained in the next section), or they may originate from the recording side via the BIFS commands sent by the authoring tool. This way for example source movements can be continuously sent from the recording side to the decoder.

Producing 2D visual interface

Above the creation and modification process of a 3D sound scene was described. However, like explained earlier, also a visual representation of the scene is sent to show the approximate positioning of sound sources in the recorded space. The author may also decide to give the renderer-side user (restricted) rights to modify the scene. For this, an interface is needed that shows the modifiable parameters. It can be assumed that in a situation where the renderer-side user has the same abilities to modify the scene as the author, he/she should ideally see a control panel containing the same sliders as in the authoring tool. The MPEG-4 source editor in Figure 4 shows how the renderer-side user rights can be chosen at the authoring stage. Common options (”movable” and ”editable” checkboxes) of the source define if the rendered source can be moved by the user, and if in general any parameter modification of that source is enabled or disabled. Next to each parameter slider is an ”UI Options” box for selecting if that parameter can be edited by the user (choice ”editable”), or if its value can be seen but not edited (choice ”visible”), or if it is completely hidden.

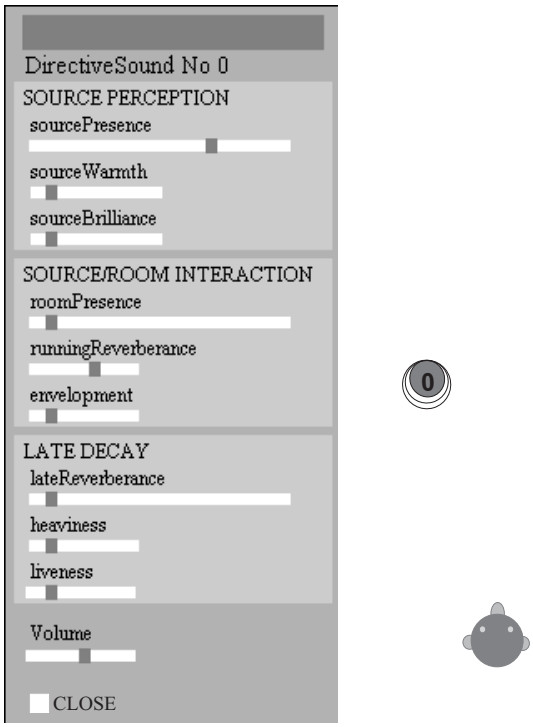

Fig. 5: The graphical part of the rendered MPEG-4 scene containing the source, the reference listening position, and the control panel for viewing and modifying the parameters of that source.

To enable the above interactions at the rendering side, a visual interface is automatically created for each sound source, and sent as a part of the MPEG-4 scene description. This visual interface (rendered on a computer screen) consists of parameter sliders defined as simple MPEG-4 2D graphical elements. When interaction is allowed (a parameter or source position is ”editable”), a slider or a graphical source object is associated with an MPEG-4 sensor node, so that they can be dragged with a mouse. The displacement of each slider is routed to a value of the corresponding parameter of the sound source that this panel is associated with (and a visual source movement to the position of the virtual sound source). To summarize, the interface consists of symbols of the sound sources, and associated control panels. When interaction is allowed, moving of a sound source (or slider corresponding to a single parameter) is routed to changes in the parameters in the 3D audio part of the scene.

IMPLEMENTATION OF THE USER INTERFACE

The previous section explained the creation process of a rendering-side user interface. As it is transmitted to the MPEG-4 decoder in an MPEG-4 format, its implementation is trivial as long as the decoder and visual renderer are compliant to the corresponding part of the standard. In other words, no external device or tool needs to be plugged to the MPEG-4 decoding and rendering system, that would realize the user interface. Figure 5 shows an example of a simple graphical representation of the transmitted MPEG-4 audiovisual scene.

SUMMARY

A framework for authoring and user interaction of MPEG-4 audiovisual scenes in the Carrouso EU project was presented. In the Carrouso system, the graphical representation of an MPEG-4 sound scene is automatically created by the authoring tool and sent as a part of the scene description. Thus each sound scene at the rendering side has a customized visual representation and interface. The author additionally has means for protecting the content by being able to restrict the rendering-side user rights to modify and/or view the scene.

The Carrouso scene authoring and real-time interaction with the scene are still at an early state of development, and not all the components in the described chain have yet been integrated together. Future improvements would include more precise study on the meaning and the proper way of defining perceptual parameters at several spatial positions, and how this can be taken into account in the MPEG-4 scene description. Also bringing more physical character to the virtual scenes (e.g., by taking into account geometrically computed reflections in a room) will probably increase the coherence between the virtual and the real space.

REFERENCES

[1] CARROUSO. Creating, Assessing and Rendering in Real time Of high quality aUdio-viSual envirOnments in MPEG-4 context. Carrouso homepage: http://emt.iis.fhg.de/projects/carrouso/, 2002.

[2] CARROUSO. IST-1999-20993. IST website http://www.cordis.lu/ist/ka3/iaf/projects/carrouso.htm, 2002.

[3] S. Brix, T. Sporer, and J. Plogsties. CARROUSO – A European Approach to 3D Audio. In *Preprint No. 5314 of the 110th AES Convention*, Amsterdam, Netherlands, 2001.

[4] R. Väänänen, O. Warusfel, and M. Emerit. Encoding and Rendering of Perceptual Sound Scenes in the CARROUSO Porject. In *Proceedings of the AES 22nd International Conference (Virtual, Synthetic, and Entertainment Audio)*, pages 289–297, Espoo, Finland, June 2002.

[5] ISO/IEC 14772-1. International Standard (IS) 14772-1. The Virtual Reality Modeling Language (VRML97) (Information technology – Computer graphics and image processing – The Virtual Reality Modeling Language (VRML) – Part 1: Functional specification and UTF-8 encoding.). April 1997. *url*: http://www.vrml.org/Specifications/VRML97/.

[6] J. Signes, Y. Fisher, and A. Eleftheriadis. MPEG-4's Binary Format for Scene Description. *Signal Processing: Image Communication. Tutorial Issue on MPEG-4*, 15(4-5):321–345, 2000.

[7] R. Koenen. Mpeg-4: Multimedia for our time. *IEEE Spectrum*, 36(2):26–33, February 1999.

[8] G. Zoia, S. Battista, A. Simeonov, and Ruo hua Zhou. Mixing Structured and Natrual Audio Coding in Multimedia Frameworks. In *Preprint No. 5513 of the 112th AES Convention*, Munich, Germany, May 2002.

[9] J. Borish. An extension of the image model to arbitrary polyhedra. *Journal of the Acoustical Society of America*, 75:1827–1836, 1984.

[10] J-M. Jot and O. Warusfel. A real-time spatial sound processor for music and virtual reality applications. In *Proceedings of the International Computer Music Conference*, pages 294–295, Banff, Canada, September 1995.

[11] R. Väänänen and J. Huopaniemi. Advanced AudioBIFS: Virtual Acoustics Modeling in MEPG-4 Scene Description. *Accepted for publication in IEEE Transactions on Multimedia*, 2003.

[12] M. M. Boone. Acoustic Rendering with Wave Field Synthesis. In *Proceedings of the ACM SIGGRAPH Campfire*, Snowbird, Utah, May 2001.

[13] K. Brandenburg, O. Kunz, and A. Sugiyama. MPEG-4 natural audio coding. *Signal Processing: Image Communication. Tutorial Issue on the MPEG-4 Standard*, 15(4-5):423–444, 2000.

[14] A. J. Berkhout. A holographic approach to acoustic control. *Journal of the Audio Engineering Society*, 36(12):977–995, December 1988.

[15] O. Delerue and O. Warusfel. Authoring of virtual sound scenes in the context of the Listen project. In *Proceedings of the AES 22nd International Conference (Virtual, Synthetic, and Entertainment Audio)*, pages 39–47, Espoo, Finland, June 2002.

[16] J.-M. Jot, L. Cerveau, and O. Warusfel. Analysis and synthesis of room reverberation based on a statistical time-frequency model. In *Preprint No. 4629 of the 103th Audio Engineering Society Convention*, New York, USA, September 1997.

[17] J-M. Jot. Efficient models for reverberation and distance rendering in computer music and virtual audio reality. In *Proceedings of the International Computer Music Conference*, Thessaloniki, Greece, September 1997.

[18] ISO/IEC 14496. International Standard (IS) 14496:2000. Information Technology – Coding of audiovisual objects (MPEG-4). Second Edition. 2000.