Object-based Audio Reproduction and the Audio Scene Description Format

MATTHIAS GEIER*, JENS AHRENS** and SASCHA SPORS†

Quality and Usability Lab, Deutsche Telekom Laboratories, Technische Universität Berlin, Ernst-Reuter-Platz 7, 10587 Berlin, Germany
E-mail: *Matthias.Geier@telekom.de; **Jens.Ahrens@telekom.de; †Sascha.Spors@telekom.de
URL: http://qu.tu-berlin.de

The introduction of new techniques for audio reproduction such as *HRTF-based technology*, *wave field synthesis* and *higher-order Ambisonics* is accompanied by a paradigm shift from *channel-based* to *object-based* transmission and storage of spatial audio. Not only is the separate coding of source signal and source location more efficient considering the number of channels used for reproduction by large loudspeaker arrays, it also opens up new options for a user-controlled interactive sound field design. This article describes the need for a common exchange format for *object-based* audio scenes, reviews some existing formats with potential to meet some of the requirements and finally introduces a new format called *Audio Scene Description Format* (ASDF) and presents the *SoundScape Renderer*, an audio reproduction software which implements a draft version of the ASDF.

1. INTRODUCTION

Audio recording, transmission and reproduction have been very active fields of research and development in the past decades. Stereophony is still the most widespread audio reproduction technique. However, the spatial cues of an auditory scene, which allow the listener to localise sound sources and to identify features of the acoustical environment, are only preserved to a limited degree. This has led to a variety of new techniques for audio reproduction such as HRTF-based technology, wave field synthesis (WFS) and higher-order Ambisonics (HOA). The introduction of these techniques is accompanied by a paradigm shift from *channel-based* to *object-based* transmission and storage of spatial audio features. The separate coding of source signal and source location is not only mandatory with respect to the high number of sometimes several hundred reproduction channels used for large loudspeaker arrays for WFS or HOA, it can also be the basis for interactive installations in which the user has access to the spatial properties of the reproduced sound field and is able to adapt it to his or her individual requirements or aesthetic preferences.

In the following sections, a brief overview is given about traditional and more recent audio reproduction methods; the terms *data-based* and *model-based* rendering are explained as well as the orthogonal pair of terms *channel-based* and *object-based*. After that, it will be shown how *object-based* audio reproduction systems are normally implemented, how the exchange of audio material is quite cumbersome and that a common interchange format is desirable. In section 5, desired properties of such a format are listed, then some already existing formats are reviewed and their weaknesses are exposed. In section 7 it is shown how an existing format can be extended to hopefully fulfil all desired properties. Finally, in section 8 the current status of the format and its applications is presented.

2. AUDIO REPRODUCTION METHODS

A great variety of audio storage and reproduction methods have evolved since the invention of the phonograph in the second half of the nineteenth century. Obviously, in the beginnings only single audio channels were recorded. Two and more channels were used with the development of *stereophony*, based on the work of Alan Blumlein in the 1930s (Alexander 1999). Up until now, stereophony is still the most widespread sound reproduction method. The term stereophony not only includes two-channel setups but also 5.1 systems and larger systems (e.g. 22.2) used mainly in cinemas. Stereophonic recordings can also be replayed via headphones (e.g. on mobile devices), although they are typically produced in a way that gives best results on dedicated loudspeaker arrangements.

From the 1970s on, *quadrophony* and *Ambisonics* (Gerzon 1973) were developed in order to provide a domestic surround experience which stereo was not capable of delivering at that time. Quadrophony employs four loudspeakers placed in the corners of a square and Ambisonics typically a few more, placed on a circle or sphere. Although all of the above-mentioned methods were initially physically motivated, their success can be largely attributed to psychoacoustical properties of the human auditory system (Theile 1980; Gerzon 1992).

The highest spatial resolution in loudspeaker-based reproduction so far is achieved by methods such as higher-order Ambisonics (HOA) (Daniel 2001) and

Organised Sound **15**(3): 219–227 © Cambridge University Press, 2010. doi:10.1017/S1355771810000324

wave field synthesis (WFS) (Berkhout, de Vries and Vogel 1993; Spors, Rabenstein and Ahrens 2008), which aim at an explicit physical synthesis of a given sound field and utilise up to several hundred loudspeaker channels or even more. Due to this high number of loudspeakers, the latter approaches are also referred to as *massive multichannel* methods. Headphone-based methods, on the other hand, typically employ *head-related transfer functions* (HRTFs), which represent the acoustical properties of the human body. These methods are also known as *binaural* methods.

In the context of electroacoustic music, various loudspeaker arrangements are used. A common setup is a circle of eight identical loudspeakers, but there are more heterogeneous and complex ones like the Birmingham ElectroAcoustic Sound Theatre (BEAST) and the Acousmonium or other individual setups for acousmatic music performances. Such systems are normally connected to a mixing console which is operated by a sound artist or the composer of the piece. This form of audio reproduction is sometimes termed *sound diffusion*.

Different ways of categorising the abovementioned approaches are possible considering, for example, the number of listeners addressed, the size of the listening area, whether the method itself employs HRTFs or addresses the listeners' HRTFs, or whether a physical reconstruction of a sound field or rather the creation of a specific perception is targeted. Recordings can also be distinguished regarding *data-based* rendering and *model-based* rendering.

In *data-based* rendering (Rabenstein and Spors 2008), the audio scenes to be reproduced have been captured by certain microphone techniques. What microphone technique is appropriate depends on the situation and the targeted reproduction system. For stereophonic reproduction (including surround), classical main microphone setups ranging from simple spaced microphone and coincident setups to more sophisticated layouts like the Fukada Tree or Hamasaki Square (Rumsey 2001) can be used. Ambisonic recordings are typically done with the *Soundfield* microphone; high-resolution recordings for HOA (Moreau, Daniel and Bertet 2006) and WFS (Hulsebos 2004) use arrays of several dozen microphones or even more.

In *model-based* rendering, an audio scene consists of a number of virtual sound sources which are described by analytical models and which are driven with a specific input signal. Analytical source models can be point sources and plane waves as well as spatially extended sources and sources with complex radiation characteristics. Such analytical source models are mainly used in the context of WFS. The creation of phantom sources by panning a sound between stereo loudspeakers can also be seen as model-based rendering. This principle is also used in Vector Base Amplitude Panning (VBAP) (Pulkki 1997).

Of course, the audio reproduction process can also be performed as a combination of data-based and model-based rendering. For example, the reverberation in a scene consisting of a number of virtual sound sources can be accomplished by reproducing a set of plane waves which approximate a room transfer function measured by a microphone array (Hulsebos 2004).

Regardless of whether a reproduction system uses data-based or model-based rendering or a combination thereof, there are two ways of transmission and storage of such scenes: *channel-based* and *object-based*. The predominant approach has always been a *channel-based* representation; in other words, loudspeaker-driving signals are somehow generated based on the real or virtual scene to be captured and then stored on a medium. This is the way stereophonic recordings are typically stored after mixdown. The major drawback of this channel-based representation is the fact that the reproduction requires a loudspeaker setup which is similar to the one for which the representation was generated. Despite this drawback, channel-based representations were the *de facto* standard for a very long time and they still have an enormous market share nowadays. It is likely that this success can be attributed to the circumstance that the quality of stereophonic reproduction for which the signals are typically generated degrades gracefully if the loudspeakers are not positioned exactly as they were in the production process.

Object-based representations of audio scenes are more flexible in terms of the employed reproduction method and setup. The objects from which loudspeaker (or headphone) signals are generated are mainly the virtual sound sources of which a scene is composed. Those objects hold the source's signal as well as its position and other parameters which are relevant for its reproduction. Object-based representation can be seen as an earlier step in the production process than channel-based storage. Besides higher flexibility, object-based representations are also more efficient than channel-based representations for modern high-resolution reproduction methods such as HOA and WFS, because they typically exhibit significantly more loudspeaker channels than simultaneously active sound sources in a given scene. In large systems of several hundred channels, object-based reproduction may be the only feasible way to go.

It is important to note that an object-based representation is not limited to model-based rendering. It can also contain data-based objects such as Ambisonics B-format recordings. An object-based scene can even contain channel-based recordings as objects. Stereophonic signals can be incorporated into a scene

by adding source objects which are playing the signals like virtual loudspeakers. This technique is referred to as *Virtual Panning Spots* (Theile, Wittek and Reisinger 2003).

3. PROBLEM STATEMENT

In the pursuit of more spatial accuracy and overall fidelity many different audio reproduction systems have been developed and installed in different institutions and venues and for different target applications. Most current high-resolution systems need a large number of loudspeakers, but also vast amounts of associated hardware such as amplifiers, digital-to-analogue converters, and cabling. They use computers of some sort, running software which is in many cases custom-made. In most cases, the whole system is specifically engineered for one reproduction algorithm and for the specific setup at hand.

Audio scenes which are prepared to be played back are normally stored in non-standardised file formats developed for a specific reproduction system. Storage formats are often tailored to account for the strengths and weaknesses of one system and contain implementation-specific data to take full advantage of the given system. However, if content is transferred to another venue which uses another reproduction system, all those customisations will be in vain and have to be painstakingly recreated using the possibilities of the target system.

This leads to a problem which implementers and operators of modern high-resolution audio reproduction systems have: they had to invest considerable effort in the creation and installation of a very sophisticated system but they still need to put much effort into each single production which is created using it. Initially, there is no material available which can be played back directly on the system; everything has to be adapted to it, often causing the same amount of work as originally creating the content. Easy exchange of audio material between different venues is virtually impossible. Therefore, an interchange format is desirable that allows the exchange of system-independent object-based audio scenes. Such a format could also facilitate the performance of 'historic' electroacoustic music on modern headphone and loudspeaker systems. Another big advantage would be the possibility of doing the scene authoring at a different place from the venue of the final performance. The preparations could be done, for example, in a small studio with an eight-channel or 5.1-channel system, or even just with headphones. Once the scene description is ready, it can be easily transferred to the venue with a massive multichannel system where only some fine tuning is left to be done.

Sometimes the unaltered playback of a spatial performance is not enough; many composers want their pieces to be interpreted and interactively adapted to the situation in the current performance space. A spatial audio interchange format should therefore allow interactive events which can trigger and manipulate certain aspects of the performance. There should also be a mechanism to synchronise the recorded spatial performance with live performance, generated sound and video material.

4. EXAMPLE SETUP

Most object-based audio reproduction systems use a combination of already available pieces of software which are connected and extended to realise the desired overall functionality. This is not necessarily a bad strategy, but it complicates the exchange with other systems. For each sub-system, specific data has to be stored in different places in a variety of file formats. The performance can only be reconstructed on another system if all the components are exactly the same.

A very common way to realise an object-based spatial audio reproduction system is to use a digital audio workstation (DAW) as the central part of the system. There are several software solutions for DAWs available, most of them proprietary and not altogether cheap. DAWs are normally used for the production of channel-based audio content. Input tracks are recorded, aligned, edited and mixed to a desired output format, for example two-channel stereo or a 5.1 surround mix. To use a DAW for object-based audio production, the individual audio tracks can be regarded as source input signals. Plugins to the DAW software can be used to assign positions, trajectories and other parameters to the sources. As this functionality is normally not included in DAW software, the actual rendering process (i.e. the generation of the loudspeaker signals for a given reproduction system) has to be done with another piece of software which is in the most cases written for one specific hardware system. The DAW provides the audio data of the separate input tracks plus the data containing positions and other parameters, and the rendering software computes the loudspeaker signals based on this data. In many cases, the DAW plugin sends real-time control data over a network socket to the rendering engine. Open Sound Control (OSC) is a popular protocol for this purpose because of its simplicity and its widespread use in audio software.

The described example setup, although often used, has several disadvantages. Apart from most DAWs being expensive, they are normally not platform-independent. To enable data exchange with another system, it normally needs to have the same DAW software and the same operating system installed. Often custom plugins are used for recording source movement and animation of other parameters, whereby the values are stored as track envelope data in the

DAW software. The values are then included in the native (often proprietary) storage format of the DAW.

5. DESIRED PROPERTIES

There are already a number of file formats available which partly satisfy the requirements for an exchange format for spatial audio scenes. A selection of them is briefly presented in the next section. This section lists all the requirements and the motivation which led to the development of the Audio Scene Description Format (ASDF). Most essentially, such a format should be able to represent what is heard in an audio scene in order that the scene under consideration can be recreated by electroacoustic means. The audio scene should be described in a way that headphone or loudspeaker driving signals can be generated for an arbitrary reproduction system such that the reproduced scene sounds as close to the initial audio scene as the chosen reproduction system allows. The reproduction system itself should not be specified in the scene description and it may range from headphone-based reproduction in a small closet to several-hundred-channel loudspeaker systems in concert halls.

The ASDF is intended for pure audio scenes. If desired, videos and other multimedia content can be synchronised to the audio scene. But, in order to keep complexity low, no graphical elements are provided in the 3D scene description. Target applications are spatial audio presentations and performances; the ASDF is not meant for 3D computer graphics and virtual reality applications, or for computer games. Since many spatial sound reproduction systems are limited to reproduction in the horizontal plane, it is desirable to have an additional, simplified 2D input mode where the third dimension can be omitted. The audio scenes to be described can be either static or they may contain dynamic features like source movement or other animated properties. Real-time user interaction should also be possible. On one hand, this gives an individual listener the possibility to explore the audio scene and manipulate it according to personal preference; on the other hand, it allows artistic interpretation of a piece in a performance situation. It should also be able to incorporate live music and generated sound and synchronise it to the spatial performance.

In order to be able to follow the latest developments in audio reproduction techniques the format should be easily extensible. In particular, concepts such as sound source directivity, spatial extent and the Doppler Effect should be taken into account. On the other hand, it should also allow a kind of lowest common denominator description of audio scenes. Every audio scene should be able to be rendered with any conceivable audio reproduction technique. The ASDF is meant to describe the spatial audio scene itself, not a specific rendition on a certain reproduction setup. In the event that a specific feature cannot be rendered, fallback mechanisms should be integrated so that the reproduction system under consideration automatically reacts in a way that minimal perceptual impairment occurs. This also holds for the case when a reproduction system only supports fewer spatial dimensions than the scene description or a smaller panorama. However, for most situations the most perceptually favourable workarounds are yet to be determined.

As described in the previous section, object-based reproduction systems typically contain a control unit which continuously sends a stream of control data to the rendering unit(s), often via network sockets. One straightforward approach to storing and recreating a spatial audio performance would be to tag those control messages with time stamps and save the stream of messages to a file. To replay the performance, the messages can be loaded from the file and then re-sent to the rendering unit at the specified times. This method is straightforward and easy to implement; however, it has some severe drawbacks. Any continuous event is split up into an unstructured series of messages. For example, the movement of a source along a circle is not stored using a symbolic representation, but as a series of position changes, sampled at a certain rate. If the chosen sampling rate is too low, the movement will be choppy and incomplete; if it is too high, a huge amount of redundant data has to be stored and processed. Additionally, if several events are happening in parallel, the messages are intermingled, making it harder to follow the events. For these reasons it is difficult to edit single events after the initial recording of the messages. It is complicated to move a source's trajectory in time or space or to assign the same movement to another source. A similar strategy is pursued with the Spatial Sound Description Interchange Format (SpatDIF) (Peters 2008), a real-time control format based on Open Sound Control (OSC). It is – very much like the ASDF – still under heavy development. By now there are discussions about extensions to overcome this limitation, but in the initial proposal of SpatDIF, a flow of successive OSC messages is simply stored in a binary file, leading to the abovementioned problems.

The ASDF is more structured and aims at a higher level representation of events. Movements, for example, can be described by means of trajectories consisting of splines. These trajectories can be moved and scaled in both time and space, they can be edited, chained, looped and assigned to other sources or groups of sources.

A goal of the ASDF is to be simple. This simplicity is desired at different levels. It means that an ASDF file should be easy to read if opened in a text editor, it should be easy to create and change and it should be

easy to write a software application which uses the format on any operating system. This implies that it should be a text-based format and not a binary format. The syntax should be easily readable both by humans and by computers. Several markup languages are available which allow the structured storage of data in text files. For the ASDF the eXtensible Markup Language (XML) was chosen because of its widespread use, its being very flexible and extensible (as the name suggests), and because there are a lot of software tools and libraries available for handling XML data.

As mentioned above, it will be possible to create and edit an ASDF file in a text editor. There is no intermediate authoring format such as, for example, the XMT formats of MPEG-4 (see next section for a short description). This means that changes can be made very easily at any time and are effective immediately. The actual audio data can be stored in any traditional audio format and linked to the scene description. In this way some flexibility is gained as audio files and scene description can be edited separately with the appropriate tools and because several different versions of a scene can be created using the same audio source material. Audio streams can also be used as input signals to include, for example, a Voice over Internet Protocol (VoIP) stream into a spatial audio performance. However, streaming of the scene description itself is not intended. If an integrated streaming solution is needed, the MPEG-4 format should be considered (see the next section).

6. ALTERNATIVE FORMATS

There are several file formats available which could partly satisfy the previously stated requirements. In the following, a few promising formats are briefly presented and their strengths and weaknesses – with respect to the application at hand – are stated.

6.1. VRML/X3D

The Virtual Reality Modeling Language (VRML) is a format for three-dimensional computer graphics, mainly developed for displaying and sharing 3D models and virtual worlds on the internet. Its scene description is based on a single scene graph, which is a hierarchical tree-like representation of all scene components. Geometrical objects are placed in local coordinate systems which can be translated/scaled/rotated and also grouped and nested in other coordinate systems which can be manipulated in the same way, and so on. Light sources, camera views and also audio objects have to be added to the same scene graph. To add an audio object to the scene graph, a `Sound` node has to be used. This node contains an `AudioClip` node which holds the information about the audio file or network stream to be presented. The format of the actual audio data is not specified by the standard. All elements of the scene graph can be animated with the so-called `ROUTE` element. This, however, is quite cumbersome for complex animations, therefore in most cases the built-in ECMAScript/JavaScript interpreter is used. To enable user interaction, mouse-events can be defined and can be bound to any visual element in the scene graph.

The use of a scene graph to represent a three-dimensional scene is very widespread in computer graphics applications. It is possible to combine very simple objects – mostly polygons – into more complex shapes and then combine those again and again to create arbitrarily complex high-level objects. When transforming such a high-level object, the transformation is automatically applied to all its components. In pure audio scenes, sounding objects normally consist of only one or a few parts, and an entire scene often contains only a handful of sources. Using a scene graph in such a case would make the scene description overly complicated. A far worse disadvantage, however, is the distribution of timing information. The timing of sound-file playback is contained in the respective `Sound` node; the timing information of animations is spread over `ROUTE`s, interpolators and scripts. This makes it essentially impossible to edit the timing of a scene directly in the scene file with a text editor.

The VRML became an ISO standard in 1997 with its version 2.0, also known as VRML97. It has been superseded by eXtensible 3D (X3D), which has been an ISO standard since 2004. X3D consists of three different representations: the classic VRML syntax, a new XML syntax and a compressed binary format for efficient storage and transmission.

6.2. MPEG-4 Systems/AudioBIFS

The ISO standard MPEG-4 contains the BInary Format for Scenes (BIFS), which incorporates the VRML97 standard in its entirety and extends it with the ability to stream scene metadata together with audio data. The audio codecs used are also defined in the MPEG standard. The spatial audio capabilities – referred to as (Advanced) AudioBIFS (Väänänen and Huopaniemi 2004) – were extended by many new nodes and parameters. Among the new features is the `AcousticMaterial` node, which defines acoustical properties such as reflectivity (`reffunc`) and transmission (`transfunc`) of surfaces, the `AudioFX` node to specify filter effects in the Structured Audio Orchestra Language (SAOL), and the ability to specify virtual acoustics in both a physical and a perceptual approach. For the latter, the `PerceptualParameters` node with parameters such as `sourcePresence` and `envelopment` can be used. Another new feature is the `DirectiveSound` node, used to specify source directivity.

AudioBIFS is a binary format which is designed to be streamed over a network. As a tool for easier creation and editing of scenes there is also a text-based representation, the eXtensible MPEG-4 Textual Format (XMT). It comes in two variants: XMT-A has a syntax very similar to X3D (see previous subsection); XMT-Ω is modelled after SMIL (see next subsection). However, the XMT is not a presentation language on its own; it must always be converted to the binary format before it can be transmitted or played back.

AudioBIFS as part of MPEG-4 Systems became an ISO standard in 1999, but has evolved since. In its most recent update – AudioBIFS v3 (Schmidt and Schröder 2004) – several features were added, among them the `WideSound` node for source models with definable shapes and the `SurroundingSound` node with the `AudioChannelConfig` attribute which allows to include Ambisonic signals and binaural signals into the scene.

AudioBIFS would definitely have all the features necessary to store spatial audio scenes. However, because of the huge size and complexity of the standard, it is very hard to implement an encoder and decoder. No complete library implementation of MPEG-4 Systems is available.

6.3. SMIL

In contrast to the aforementioned formats, the XML-based Synchronized Multimedia Integration Language (SMIL, pronounced like 'smile') is not able to represent three-dimensional content. Its purpose is the temporal control and synchronisation of audio, video, images and text elements and their arrangement on a 2D screen. Since 1998, the SMIL has been standardised as a Recommendation of the World Wide Web Consortium (W3C); the current version of the standard (SMIL 3.0) was released in 2008.

All SMIL functionality is organised in modules, for example `MediaDescription`, `PrefetchControl` and `SplineAnimation`. Different sets of modules are combined to language profiles tailored for different applications and platforms. With the 3GPP SMIL Language Profile, SMIL is used for the Multimedia Messaging Service (MMS) on mobile phones. The central part of a SMIL document is a timeline where media objects can be placed either relative to other objects or by specifying absolute time values. The timing does not have to be static: interactive presentations can be created where the user dictates the course of events, for example by mouse clicks. Animations along 2D-paths are possible with the `animateMotion` element. The temporal structure is mainly defined by `<seq>`-containers ('sequential'), whose content elements are played consecutively one at a time, and by `<par>`-containers ('parallel'), whose content elements start all at the same time. Of course, these containers can be arbitrarily nested giving possibilities ranging from simple slide shows to very complex interactive presentations. Inside the time containers, media files are linked to the SMIL file with `<img>`, `<audio>`, `<text>` and similar elements. SMIL has very limited audio capabilities. Except for the temporal placement, the only controllable parameter of audio objects is the sound level, given as a percentage of the original volume. The SMIL format itself is particularly not able to represent 3D audio scenes, but it can either be used as an extension to another XML-based format or it can be extended itself. To extend another XML-based format with SMIL timing features, the W3C Recommendation SMIL Animation can be utilised. This was done, for example, in the widespread Scalable Vector Graphics (SVG) format. However, SMIL Animation is quite limited because a 'flat' timing model without more powerful time containers (such as `<par>` and `<seq>`) is used. A more promising approach would be to extend SMIL with 3D audio features. An example for such an extension is given by Pihkala and Lokki (2003), where SMIL was extended with the so-called Advanced Audio Markup Language (AAML).

7. EXTENDING SMIL

As mentioned before, SMIL is not able to represent three-dimensional audio scenes. It has, however, a very convenient timeline concept and the temporal alignment of audio objects is very flexible and powerful. To add 3D audio features, SMIL can be extended using a new XML namespace. Figure 1 shows an example scene. All scene elements which are part of the ASDF and not part of the SMIL, are prefixed by '`a:`'. The file can still be opened with a standard SMIL player, which just ignores the added elements. If opened in an ASDF-aware player, the additional elements are also taken into account and the whole scene with both its temporal and spatial properties is played back as intended.

The example scene is very simple; it comprises only three short audio files. The `<body>` element holds two `<par>` elements which are played consecutively, because the `<body>` element implies a `<seq>` element. In the first `<par>` container there is only one audio file, which is played once while its position is changed. When the file is finished, the second `<par>` element is entered. The second of the two contained audio files is played 7 seconds later; the first one is repeated for 1 minute and 15 seconds. After this time, the whole scene is finished.

It is important to note that the coordinate system of the audio scene is not the same one used in the original SMIL format. SMIL coordinates are specified in pixels and are used to place text, images and

```
<smil xmlns="http://www.w3.org/ns/SMIL"
      baseProfile="Language" version="3.0"
      xmlns:a="http://qu.tu-berlin.de/ASDF">
<head>
  <meta name="title" content="SMIL+ASDF example"/>
</head>
<body>
  <par>
    <audio xml:id="intro" src="media/intro.wav" a:pos="1 0.4 0"
        soundLevel="60%"/>
    <set begin="2s" targetElement="intro" attributeName="a:pos"
        to="1.2 2 0.5"/>
  </par>
  <par>
    <audio xml:id="background" src="media/background.wav"
      repeatDur="1:15" a:pos="0.7 0.7 1">
      <animate attributeName="soundLevel" begin="6s" dur="1.5s"
        from="100%" to="0%"/>
    </audio>
    <audio xml:id="voice" src="media/voice.wav" begin="7s"/>
    <set begin="2s" targetElement="background" attributeName="a:pos"
    to="-0.7 0.7 1"/>
  </par>
</body>
</smil>
```

Figure 1. Code listing for a SMIL document extended by the ASDF.

videos on a two-dimensional screen. The coordinates of the ASDF are given in metres and they are specified in a right-handed Cartesian coordinate system where the x- and y-axes lie in the horizontal plane and the z-axis points upwards. The listener's position is not the origin of the coordinate system. The listener can be placed anywhere in the coordinate system and can be animated along trajectories in the same way like sound sources.

Another advantage of using the SMIL as basis of ASDF is that all the visual features can still be displayed on a screen. Therefore, video material can be synchronised to the spatial audio scene, titles and descriptions of scenes or parts thereof can be displayed, and subtitles can be provided. It is even possible to define a custom user interface with means to play and pause the performance, but also to jump to certain 'chapters' or 'movements'.

8. STATUS OF THE ASDF AND ITS APPLICATIONS

The ASDF is being developed in parallel with the SoundScape Renderer (SSR) (Geier, Ahrens and Spors 2008). The SSR is a versatile software framework for spatial audio reproduction. It follows the concept of object-based reproduction, as described in section 2. This means that the reproduction method is not specified in the scene description. The SSR provides arbitrary rendering methods with one common scene management and a common graphical user

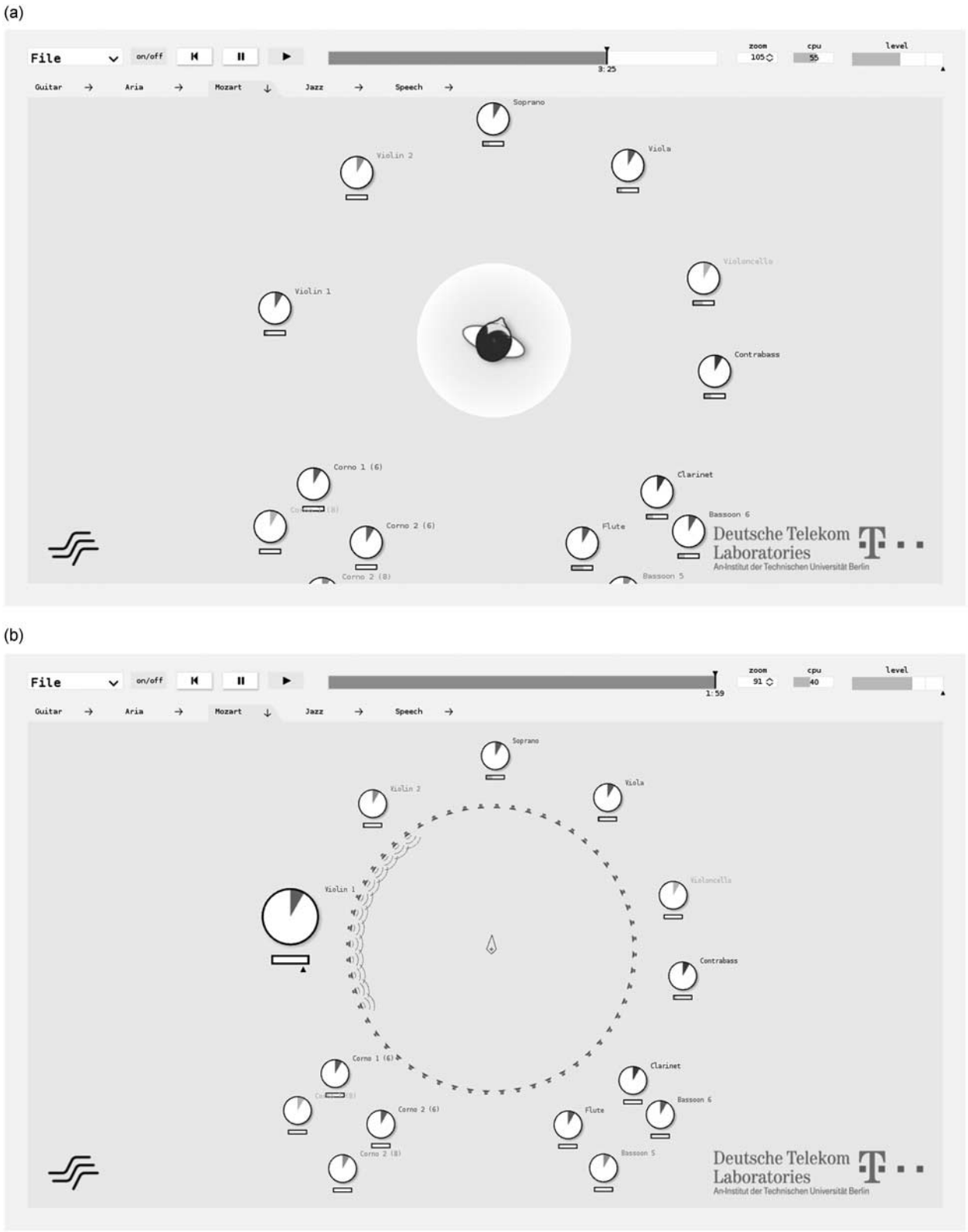

Figure 2. Two screenshots of the SoundScape Renderer showing the same scene using (a) headphone-based reproduction and (b) wave field synthesis.

interface. Several rendering modules are already implemented, among them WFS, HOA, VBAP, HRTF-based rendering and a binaural room scanning (BRS) renderer. The two screenshots in figure 2 show the graphical user interface of the SSR displaying the same audio scene but with different reproduction

modules; in one case with headphone reproduction and in the other case using the WFS renderer. The SoundScape Renderer is Free and Open Source Software, released under the GNU General Public License (GPL).[1]

Although the ASDF is developed in conjunction with the SSR, it is a separate project and can be used with any other reproduction software. Once it is finished, it will be freely available along with the Open Source reference implementation of a software library for reading and storing ASDF files. The ASDF is still in an early stage of development and it supports, for now, only static scenes. The description of source trajectories is one of the next goals in its further development. Another important aspect is an event system which allows interactive manipulation of a spatial audio scene and synchronisation with external events. At a later state, reverberation and a simple room model will be discussed.

9. CONCLUSION

We have presented the Audio Scene Description Format (ASDF), which constitutes an extension of the Synchronized Multimedia Integration Language (SMIL). SMIL's very convenient media timing features were extended by 3D audio capabilities. The ASDF describes an *object-based* representation of an audio scene in order to avoid the drawbacks of conventional *channel-based* representations, especially the restricted flexibility and the enormous amount of data which arises in conjunction with modern high-resolution reproduction systems. The reproduction setup is not specified in the scene description and is therefore arbitrary. This allows system independent mixing, which means, for example, that an audio scene may be prepared using headphones and then reproduced for a large audience with a given loudspeaker-based system. The fact that the individual audio objects are available at the consumer side brings high flexibility in terms of real-time interaction with the scene, since local modifications in the scene can be straightforwardly performed.

The present paper has outlined the fundamental properties and current status of the ASDF. For further development, the spatial audio community is asked to contribute by describing the specific requirements of single reproduction systems and by suggesting any improvements that can be added to the format.

[1]The software can be downloaded for free from the website http://tu-berlin.de/?id=ssr.

REFERENCES

Alexander, R.C. 1999. *The Inventor of Stereo: The Life and Works of Alan Dower Blumlein*. Oxford: Focal Press.

Berkhout, A.J., de Vries, D. and Vogel, P. 1993. Acoustic Control by Wave Field Synthesis. *Journal of the Acoustical Society of America* **93**(5): 2,764–778.

Daniel, J. 2001. *Représentation de champs acoustiques, application à la transmission et à la reproduction de scènes sonores complexes dans un contexte multimedia*. PhD thesis, Université Pierre et Marie Curie (Paris VI).

Geier, M., Ahrens, J. and Spors, S. 2008. The SoundScape Renderer: A Unified Spatial Audio Reproduction Framework for Arbitrary Rendering Methods. *Proceedings of the 124th Convention of the Audio Engineering Society*. Amsterdam: AES.

Gerzon, M.A. 1973. Periphony: With-Height Sound Reproduction. *Journal of the Audio Engineering Society* **21**(1): 2–10.

Gerzon, M.A. 1992. General Metatheory of Auditory Localisation. *Proceedings of the 92nd Convention of the Audio Engineering Society*. Vienna: AES.

Hulsebos, E.M. 2004. *Auralization using Wave Field Synthesis*. PhD thesis, Delft University of Technology.

Moreau, S., Daniel, J. and Bertet, S. 2006. 3D Sound Field Recording with Higher Order Ambisonics – Objective Measurements and Validation of a 4th Order Spherical Microphone. *Proceedings of the 120th Convention of the Audio Engineering Society*. Paris: AES.

Peters, N. 2008. Proposing SpatDIF – The Spatial Sound Description Interchange Format. *Proceedings of the 2008 International Computer Music Conference*. Belfast/San Francisco: ICMA.

Pihkala, K. and Lokki, T. 2003. Extending SMIL with 3D Audio. *Proceedings of the 2003 International Conference on Auditory Display*. Boston: ICAD.

Pulkki, V. 1997. Virtual Sound Source Positioning using Vector Base Amplitude Panning. *Journal of the Audio Engineering Society* **45**(6): 456–66.

Rabenstein, R. and Spors, S. 2008. Multichannel Sound Field Reproduction. In J. Benesty, M.M. Sondhi and Y. Huang (eds.) *Springer Handbook on Speech Processing*. Berlin: Springer.

Rumsey, F. 2001. *Spatial Audio*. Oxford: Focal Press.

Schmidt, J. and Schröder, E.F. 2004. New and Advanced Features for Audio Presentation in the MPEG-4 Standard. *Proceedings of the 116th Convention of the Audio Engineering Society*. Berlin: AES.

Spors, S., Rabenstein, R. and Ahrens, J. 2008. The Theory of Wave Field Synthesis Revisited. *Proceedings of the 124th Convention of the Audio Engineering Society*. Amsterdam: AES.

Theile, G. 1980. *On the Localisation in the Superimposed Soundfield*. PhD thesis, Technische Universität Berlin.

Theile, G., Wittek, H. and Reisinger, M. 2003. Potential Wavefield Synthesis Applications in the Multichannel Stereophonic World. *Proceedings of the 24th International Conference of the Audio Engineering Society*. Banff: AES.

Väänänen, R. and Huopaniemi, J. 2004. Advanced AudioBIFS: Virtual Acoustics Modeling in MPEG-4 Scene Description. *IEEE Transactions on Multimedia* **6**(5): 661–75.