AN ACOUSTIC EVENTS MODELING LANGUAGE (AEML)

PACS REFERENCE: 48.38.Md, 43.48.Vk

Yoshinori TAKAHASHI; Mitsuo MATSUMOTO; Mikio TOHYAMA
Kogakuin University, Bldg. #5 Room 603
2665-1 Nakano-machi Hachioji-shi Tokyo, 192-0015, Japan
Tel: +81-426-28-4658
Fax: +81-426-28-4658
E-mail: em01014@ccs.kogakuin.ac.jp

ABSTRACT

A computer-based acoustic events modeling language (AEML) will enable to render an acoustic virtual-event based on a model of virtual acoustic events related to a signal source, a signal transmission path, and signal perception and understanding. Our AEML system consists of a database that includes dictionaries related to hearing properties such as head-related transfer functions (HRTFs), and an audio-scene processor of scene-description texts. The functions consist of virtual acoustic-event modeling programs, where the modeling is mostly done from the point of view of hearing (rather than being based on a physical model of signal generation). Generating an acoustic event that represents a moving sound source is a typical example of an AEML application in virtual acoustics.

INTRODUCTION

Through our studies of acoustic generation and transmission, and of human perception and comprehension of acoustics, we hope to develop an immersive communication system [1-3] based on a virtual acoustic space. Tohyama and Yanagawa have been developing the Interactive Sound Field Network (ISFN) [4] shown in Fig. 1. To generate and communalize the acoustic space, we need to render the information normally provided by sound sources and transfer systems. We have therefore developed the acoustic events modeling language (AEML) for rendering acoustic events.

AEML, which can be used to represent acoustic events by text, is a computer language created specifically for rendering acoustic events. It embodies our belief that acoustic events can be expressed by algorithms based on linear systems. The AEML compile system that generates acoustic events from an AEML script comprises an AEML processor, acoustic event function programs, and an event parts database. When the AEML processor generates acoustic events by processing an acoustic events script, it requests event parts from the database – for example, the head-related transfer functions (HRTFs) and room impulse responses needed for the acoustic events. In this article, we explain the acoustic events that AEML can replicate, the operation of the AEML compiler, and the format used for the AEML script.

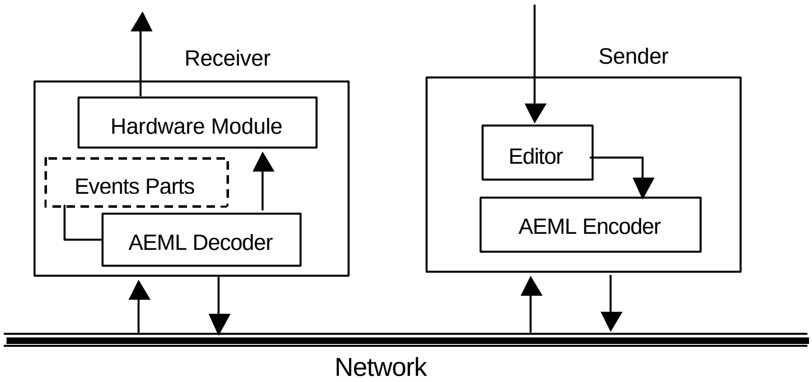

Figure 1. Interactive Sound Field Network

ACOUSTIC EVENTS

We often perceive a single sound image in front of us when we listen to sounds from two loudspeakers (Fig. 2). Although the physical phenomenon is that we hear two sounds from two separate loudspeakers on our right and left, the psychological phenomenon is that we perceive a single sound image in front of us. In this paper, we call this latter phenomenon an "acoustic event". For example, the acoustic event in Fig. 2 is that there is one sound image in front of the listener, not two sounds from two loudspeakers. Furthermore, a human listener can be conscious of and understand the sound phenomenon underlying an acoustic event. Other similar examples are when a sound seems to come from behind a screen or the sound reverberates as if in a large room. We define acoustical phenomena that we can listen to and perceive as a distinct sensation as acoustic events.

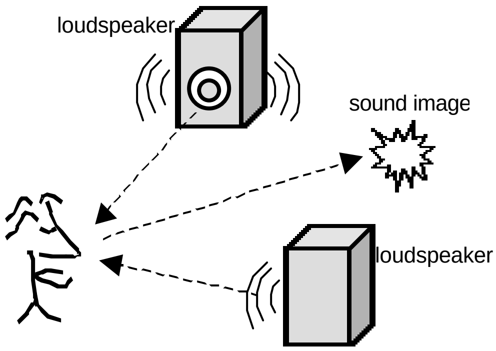

Figure 2. An acoustic event

AEML COMPILER

The AEML compiler – consisting of an AEML processor, acoustic event functions, and an event parts database -- creates acoustic events from the AEML script coding (Fig. 3).

<u>AEML Processor</u>

The AEML processor, the most important part of the AEML compiler, includes a token analyzer, parser, and signal-processing program.

Acoustic Event Functions

A set of the event function programs necessary for modeling acoustic events are stored and requested by the AEML processor as needed.

Events Parts Database

Ideally, all acoustic events described with AEML would be synthesized through signal processing, but this is not yet possible. As a result, we have to use a database to store and sort parts for modeling acoustic events such as dry sources and transfer functions (e.g., HRTFs), human speech and the sounds of musical instruments.

AEML FORMAT

One event is described on one line in AEML. The general format of an event is:

Subject : *Event type* (*Parameter1*, *Parameter2*, ~) ;

"*Subject*" is a variable that causes events. The "Event type" determines what event will occur. "Parameters" establish how an event will occur. A series of events written in a particular order provides an acoustic events script. By writing time information in a parameter, we can describe multiple events occurring at the same time through AEML. Figure 4 shows an example of an AEML acoustic events script where the wave file "demo.wav" is read as variable s1. At time 0, sound source s1 appears at a point where the angle between the s1 location and the median plane is 90º left. From there, s1 will slowly rotate 360º to the right.

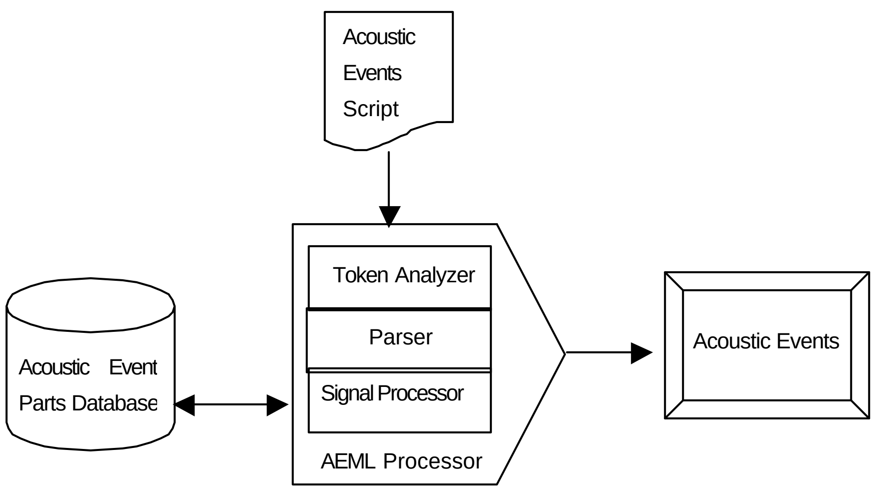

Figure 3. AEML Compiler

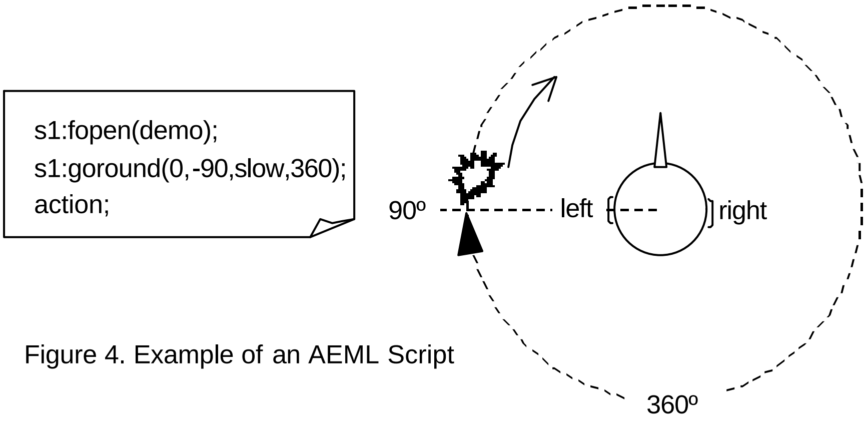

Figure 4. Example of an AEML Script

CONCLUSION

Acoustic events modeling language (AEML) can be used to render acoustic events for the ISFN. With improved signal-processing algorithms and processor computing power, we expect AEML to enable real-time processing. For use in the ISFN, the AEML will have to enable both the modeling of the acoustic events of a receiver as well as the coding of the acoustic events of a sender. To achieve this, the AEML must include functions to allow event separation and direction detection of a sound source in order to extract a sound source from acoustic phenomena. This research has been supported by the Telecommunications Advancement Organization (TAO) of Japan, and the International Communications Foundation.

REFERENCES

[1] L. Sacioja et al., Creating Interactive Virtual Acoustic E nvironments, J. Audio Eng. Soc., vol. 47 pp.675-705 (1999)

[2] C. Kyriakakis et al., Surrounded by Sound, IEEE Signal Processing Magazine, vol.1, pp.55-66 (1999)

[3] C. Kyriakakis, Fundamental and Technological Limitations of Immersive Audio Systems, Proc. IEEE 86 pp. 941-951 (1998)

[4] M. Tohyama et al., Acoustic Events Modeling for Rendering 3D Sound and Perception, Proceedings of IWSM'01 The Sound International Workshop on Spatial Media, pp.80-85 (2001)