Grant Agreement No.: 687645
Research and Innovation action
Call Topic: H2020 ICT-19-2015

Object-Based Audio Experience

Object-based broadcasting – for European leadership in next generation audio experiences

D3.2: Implementation and documentation of reverberation for object-based audio broadcasting

Version: v1.0

Deliverable type	R (Document, report)
Dissemination level	PU (Public)
Due date	30/11/2016
Submission date	09/12/2016
Lead editor	Markus Noisternig (IRCAM)
Authors	Markus Noisternig (IRCAM), Thibaut Carpentier (IRCAM), Matthias Geier (IRCAM), Olivier Warusfel (IRCAM)
Reviewers	Tilman Herberger (MAGIX), Chris Pike (BBC), Andreas Silzle (IIS)
Work package, Task	WP3, T3.1, T3.2
Keywords	Reverberation, audio production, object-based broadcast

Abstract

This document discusses methods for the representation of reverberation in object-based broadcast and also partly describes an example implementation.

Document revision history

Version	Date	Description of change	List of contributor(s)
v0.1	11/11/16	First draft	Markus Noisternig
v0.2	20/11/16	Added more information on reverberation processing, changes merged to one document	Markus Noisternig, Thibaut Carpentier
v0.3	22/11/16	Revised sub-section on ADM representation of reverberation	Markus Noisternig, Thibaut Carpentier, Olivier Warusfel
v0.4	23/11/16	Moved parts of the main document to the appendix	Markus Noisternig, Thibaut Carpentier, Olivier Warusfel
v0.5	25/11/16	Added examples for ADM metadata schemes, changes merged to document	Markus Noisternig, Thibaut Carpentier, Matthias Geier, Olivier Warusfel
v0.6	26/11/16	Minor corrections	Andreas Silzle, Markus Noisternig
v0.7	28/11/16	Minor corrections	Nicolas Epain, Markus Noisternig
v0.71	29/11/16	Minor corrections	Markus Noisternig, Thibaut Carpentier, Olivier Warusfel
v0.8	30/11/16	Minor corrections	Chris Pike, Markus Noisternig
v0.9	30/11/16	Changes merged to one document for PMC approval.	Markus Noisternig
v1.0	08/12/16	Document approved by the PMC	PMC

[1] http://creativecommons.org/licenses/by-nc-nd/3.0/deed.en_US

Executive Summary

Room effects and reverberation are essential in audio post-production. Reverberation can be used to simulate a specific room (e.g., a church for choral music) and can also be used to achieve a desired spatial sound effect (e.g., using the signal-to-reverb ratio for controlling the perceived sound source distance in a spatial audio scene). Existing object-based audio metadata schemes only provide some basic reverberation parameters (e.g., ADM, MPEG-H) or do not support reverberation objects at all.

This document presents methods, workflows and an example implementation of reverberation processing and storage in object-based broadcast.

Table of Contents

List of Figures

Abbreviations

AABIFS	MPEG-4 Advanced Audio Binary Format for Scenes
ADM	Audio Definition Model, ITU-R BS.2076-0
AR	Augmented Reality
BIFS	MPEG-4 Binary Format for Scenes
DOA	Direction of Arrival
DRIR	Directional Room Impulse Response (sometimes also referred to as Spatial Room Impulse Response, SRIR)
EDC	Energy Decay Curve
EDR	Energy Decay Relief
FBE	Fourier Bessel Expansion
FDN	Feedback Delay Network
FIR	Finite Impulse Response
HOA	Higher Order Ambisonics
IIR	Infinite Impulse Response
MPEG-H	MPEG-H 3D Audio Standard, ISO/IEC 23008-3 (MPEG-H Part 3)
MPEG-4	MPEG-4 Standard, ISO/IEC 14496 – Coding of audio-visual objects
RIR	Room Impulse Response
RSAO	Reverberant Spatial Audio Object
SLA	Spherical Loudspeaker Array
SMA	Spherical Microphone Array
SNR	Signal-to-Noise Ratio
SRR	Signal-to-Reverb Ratio
SpatDIF	Spatial Sound Description Interchange Format
S-T-F	Space Time Frequency (Representation)
T-F	Time Frequency (Representation)
VR	Virtual Reality

1 Introduction

Room effects and reverberation are essential in audio post-production. They can be used to simulate a specific room (e.g., a church for choral music) and to achieve a desired spatial sound effect (e.g., using the signal-to-reverb ratio for controlling the perceived sound source distance in a spatial audio scene). Existing object-based audio metadata formats (see e.g., ADM, MPEG-H) do not provide or only provide some very basic reverberation parameters. This document presents methods, workflows and an example implementation of reverberation processing and storage in object-based broadcast.

1.1 Structure

Section 2 starts with a short overview on methods for representing and processing reverberation, and discusses the key low-level and high-level parameters for controlling the reverberation process. Then, in Section 3, we outline the use of these methods in object-based broadcast.

1.2 Contributors

- IRCAM

1.3 Implementation

The proposed methods have been partly implemented and will be made available to the Orpheus project partners to validate future directions for the representation of reverberation in the workflow for object-based audio broadcast.

2 Reverberation processing

Room reverberation is responsible for temporal, spectral smearing of an audio signal, which results in distortion in both the envelope and the fine structure of the acoustic signal. However, room reverberation helps us to better perceive the acoustic environment and is regarded as one of the primary cues in human auditory perception that affect sound source localisation, distance perception, room dimension estimation, and orientation in a given space by a listener. For this reason, reverberation processing is considered as a critical part of audio production and plays a major role in sound engineering practice.

The following sections introduce different models for simulating and representing room acoustic reverberation, and briefly discuss their relation to object-based metadata schemes for broadcasting. A more detailed discussion on the processing of room impulse responses (denoising and FDN parameter extraction) is given in Appendix A.1; Appendix A.2, which is associated with Section 3, presents an example XML implementation of reverberation in the ITU Audio Definition Model (ITU-R BS.2076) [15] metadata scheme.

2.1 Reverberation model

The acoustical path between an emitter and a receiver in a room is usually modelled as a linear time-invariant system, which is fully characterised by its room impulse response (RIR). Although acoustic environments are inherently time-varying (e.g., due to moving sound sources and changes in atmospheric conditions), they generally change slowly compared to the length of their impulse responses. Hence most of the signal processing methods used for reverberation rendering assume the RIR being stationary and model it with an FIR filter. If needed, the non-stationarity can be addressed by so-called block-based methods that divide time into periods in which the acoustic channel is assumed to be stationary. The processing time periods can differ for the direct sound and early reflections (which change with the source position) and the late reverberation (which in most cases can be assumed as being static).

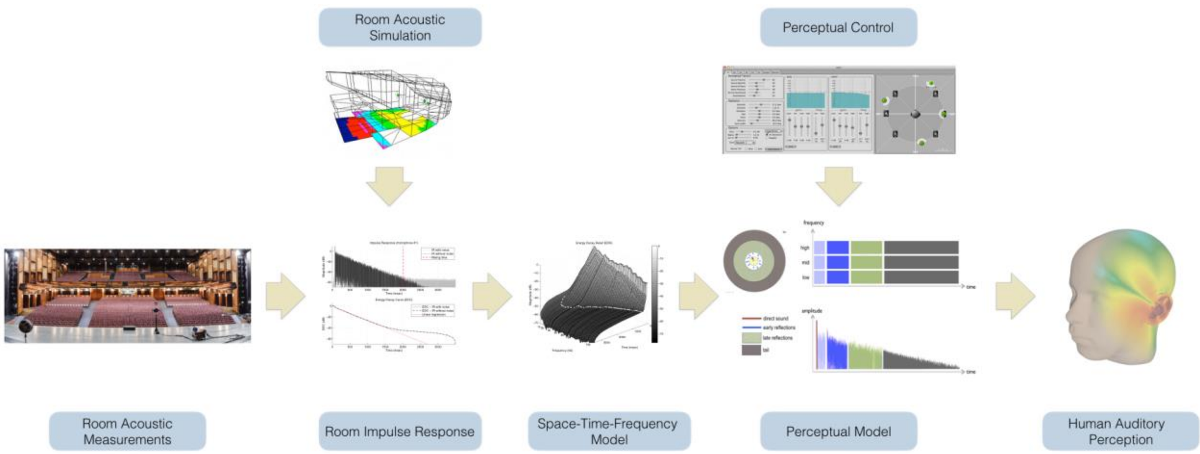

Figure 1: Room reverberation models

Figure 1 depicts different methods to create and represent room impulse responses. The impulse response for a given emitter and receiver position in a room can be obtained either from room acoustic measurements or from acoustic simulations given the architectural model of the room (see e.g., [26][54]). The time-frequency (T-F) and space-time-frequency (S-T-F) representations of the RIR ease the estimation of perceptual features (e.g., sound source presence and listener envelopment) and hence provide a perceptual signature of the room. With this high-level perceptual control the room acoustic behaviour can be modified along the various perceptual dimensions, preserving the microstructure of the original RIR, before being re-synthesized for the use with reverberation processors [3].

2.1.1 Room acoustic measurement

A common method for measuring room impulse responses is to apply a known input signal and to measure the room response. Then the room response is deconvolved by the excitation signal to obtain the RIR. In general, measured RIRs are corrupted by measurement noise and environmental noise that mask the late reverberation tail. In order to achieve the best quality for creating a room effect, the measured RIRs must reach a very good signal-to-noise ratio (SNR). The SNR depends on the choice of the excitation signal, the deconvolution technique, and the number of averages of consecutive measurements (see, e.g., [11][48]). A typical room impulse response and its corresponding energy decay curve (EDC) are depicted in Figure 2.

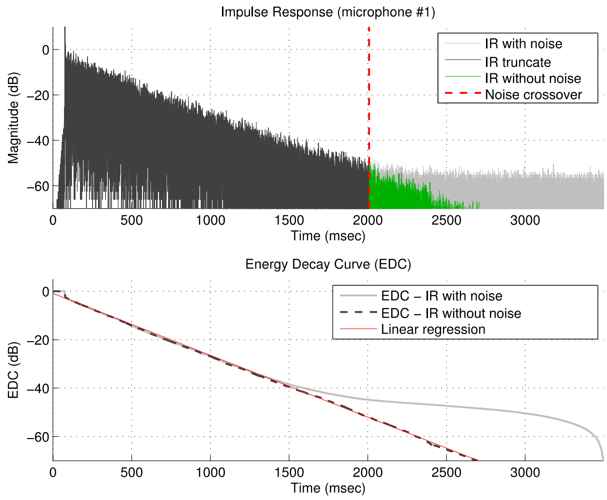

Figure 2: Room impulse response and energy decay curve

Room impulse responses are usually measured with omnidirectional loudspeakers and microphones. Microphone arrays have been proposed for the measurement of so-called directional room impulse responses (DRIRs)[2] [1][9][24]. DRIRs enable the S-T-F analysis of the incident sound field. Similarly, loudspeaker arrays and directional loudspeakers have been used for room acoustic analysis in order to radiate acoustic energy to some selected directions in space ([37][39][49]).

Combining a spherical loudspeaker array (SLA) with a spherical microphone array (SMA) results in a so-called acoustic MIMO[3] system ([30][31]) for room acoustic analysis, which in general adds higher spatial resolution (i.e. spatial diversity) to the three-dimensional analysis of the incident sound field [32]. Figure 3 depicts a typical MIMO RIR measurement system.

In room acoustics auralization, the RIR, DRIRs, and MIMO-RIRs are directly applied to convolution-based reverberation processors (see Section 2.2.2). The S-T-F representation of DRIRs allows for a perceptual control of the reverberation process (see the Spat-model in Section 2.2.4); MIMO-RIRs also allow for controlling the sound source directivity.

[2] DRIRs are often also referred to as spatial room impulse responses (SRIRs) in literature.

[3] Multiple-input multiple-output (MIMO) system.

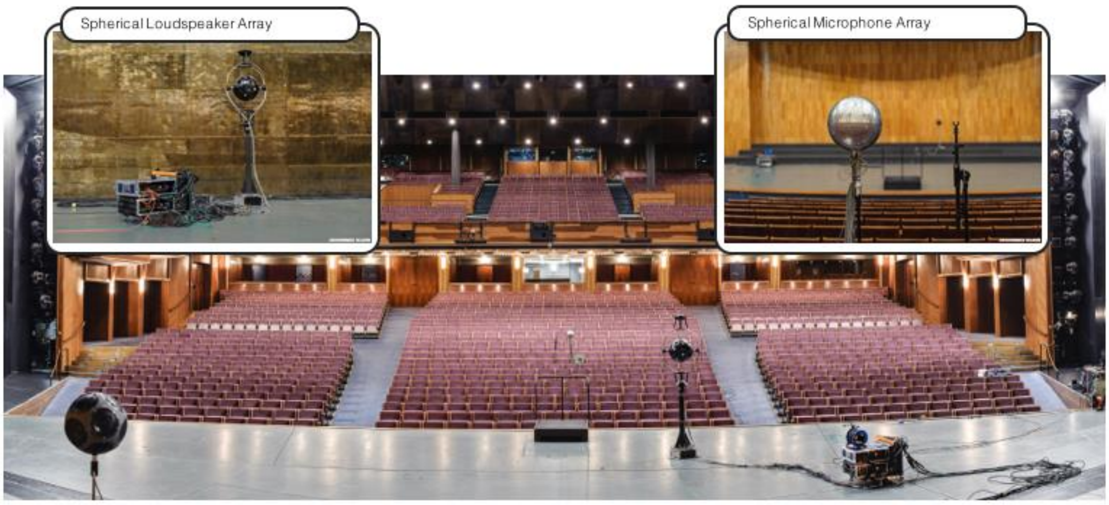

Figure 3: MIMO room impulse response measurements in a concert hall.

2.1.2 Room acoustic simulation model

Any sound field in an enclosure can be regarded as the common decaying of free vibrational modes, which depend on the shape of the room and the physical properties of its boundary (e.g., the surface material structure and absorption coefficients of the walls). These vibrational modes lead to sound coloration effects at the listener's position. Computing the vibrational modes of enclosures is computationally demanding and difficult to apply to realistic rooms (see e.g. [43]).

The concept of geometric room acoustics models the propagation of acoustic energy as sound rays, which is only valid for frequencies for which diffraction and interferences can be neglected. Room reverberation can thus be described as the sum of all sound reflections arriving at a certain point in the room after the room was excited by an impulsive sound signal (see e.g. [26][54]). With the concept of geometric room acoustics not only the spatial distribution of stationary sound energy can be modelled but also the temporal distribution of sound reflections arriving at a given point in the room ([33][45]). This can be applied to low-level descriptors for reverberation in object-based audio (see Section 3.1), which interpret the estimated direction-of-arrival of early reflections as plane waves.

Figure 4 compares the broadband energy time curve (ETC) for different reflection orders (up to order 3) simulated with a geometrical acoustics method and a very high order simulation [33]. It is clear that the simulation error decreases with increasing reflection order. A statistical model was used to approximate the late reverberation; the time envelope was estimated by applying linear regression on the results obtained from a simulation with higher order early reflections.

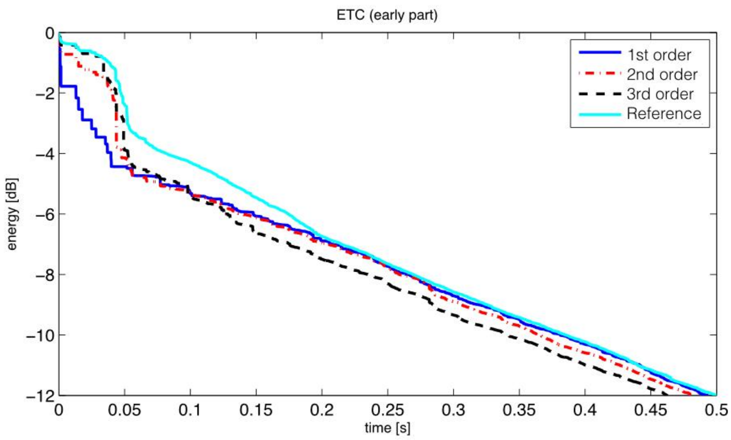

Figure 4: Geometric room acoustics simulation and the change of the early part of the corresponding energy time curve (ETC) with the reflection order.

The geometric room acoustic simulation model has been shown to be well suited for virtual reality (VR) and augmented reality (AR) applications, in which the acoustic environment can be well defined by geometrical models. MPEG-4 supports a geometrical room description metadata scheme for acoustic room modelling. It uses the so-called "Advanced AudioBIFS" nodes for creating scenes with enhanced spatial properties, and for transmitting a parametric representation of the 3-D environment (see e.g. [52]). However, geometrical room acoustic models are rarely implemented in audio post-production tools for broadcast.

2.1.3 Perceptually-motivated simulation model

Figure 5 depicts a generic simplified space-time-frequency model of an RIR. A survey on the temporal structure (lower part of the figure) shows the direct sound (R0), the discrete early reflections (R1), the late early reflections (R2), which are often also referred to as "cluster" in room acoustics literature, and the diffuse late reverberation (R3). The spectral analysis is limited to three frequency bands, and only the very early discrete reflections are modelled as sound rays coming from different directions (upper part of the figure). The late reflections (cluster) and the late reverberation are modelled as a diffuse (i.e. de-correlated isotropic) sound field that conserves the statistical properties of the diffuse sound field.

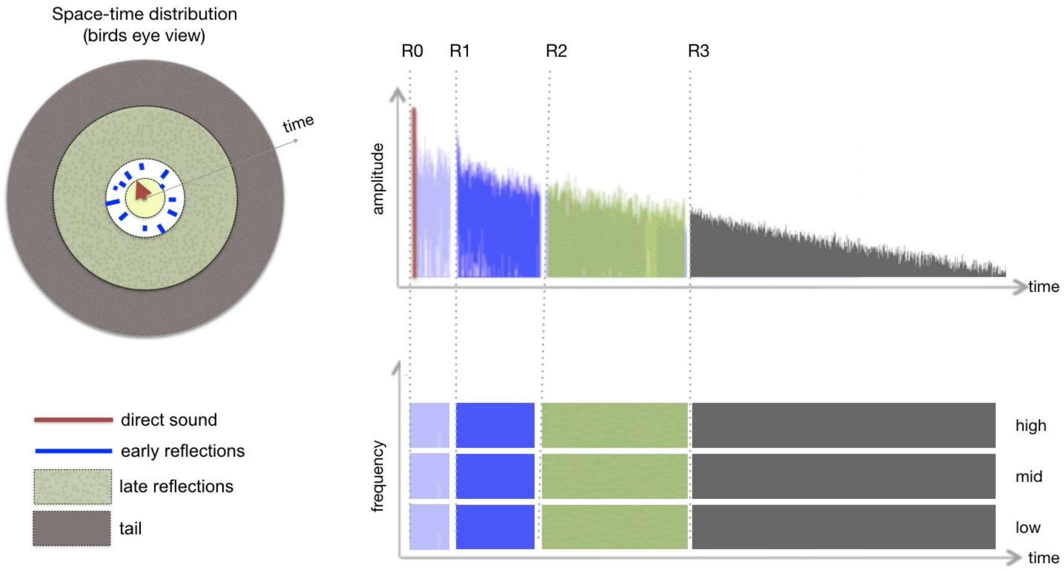

Figure 5: Generic simplified space-time-frequency RIR model showing the direct sound (R0), the discrete early reflections (R1), diffuse early reflections / cluster (R2), and late diffuse reverberation (R3). The left sub-figure shows a symbolic birds-eye view on the space-time representation of the spatial pattern, where the radius corresponds to time; solid lines depict the direction-of-arrival of the direct sound and the discrete reflections, the filled slices symbolise a diffuse sound field.

2.2 Simulation of reverberation

For several decades, digital reverberation processors used computationally efficient feedback delay networks (FDN) to simulate early reflections and the statistical properties of late room reverberation [16]. FDNs are scalable and allow for a flexible control of different levels of descriptors, from low-level signal processing parameters (e.g., initial delay, echo density, reverberation decay, etc.) to high-level perceptual descriptors (e.g., source presence, envelopment, reverberance, etc.). A commonly reported drawback of FDN rendering is the lack of authenticity in the early part of the room response. This is typically linked to transient coloration effects or from insufficient echo and/or modal densities, as it takes some time to build up dense reflection patterns with feedback loop structures [3].

Convolution-based reverberation processors filter the sound signal with the RIR for a given source and receiver position. They usually also provide some basic control over the perceptually most relevant parameters of an RIR. Convolution-based reverberation processors use large collections of RIRs measured in prestigious concert halls (e.g., the Concertgebouw and the Musikvereinssaal) and other typical venues (e.g., cathedrals and antique theatres). The convolution guarantees for a natural listening experience, but comes with a high computational cost [53].

2.2.1 Parametric reverberators

Feedback delay networks (FDN) processing structures have been proposed for digital reverberation processors ([16][44]) to simulate the statistical properties of room reverberation with arbitrary density and low tonal coloration in the late decay. FDNs can be efficiently implemented and allow for a continuous tuning of the time and frequency behaviour of the room response.

IRCAM's software Spat~ [4] is an example of a parametric reverberation processor that is based on FDN. It is controlled via a set of perceptual descriptors relying on a simplified model of the time-frequency energy distribution of the RIR over four temporal segments and three frequency bands [18]. The spatial distribution can be taken into account by applying panning models that are specific to each time segment. Figure 5 shows the applied space-time model where the direct sound (R0) arrives from a given direction, the individual early reflections (R1) are synthetized as waves coming from deterministic directions, and the late reflections (R2) and reverberation decay (R3) are spatially diffuse fields. Figure 6 shows the FDN processing structure described in ([16][18]). It consists of a "lossless prototype" with infinite reverberation time that is based on lossless unitary feedback matrix structures combined with absorptive IIR filters. With this processing structure one can achieve arbitrary time and modal densities, low tonal coloration, and independent control of the frequency envelope and decay characteristics [16]. The cluster is synthesized with multi-tap delay lines feeding a decorrelation unit; the late reverberation is generated by a delay-network (that is fed by the output of the cluster section) with typically 4 to 16 feedback channels.

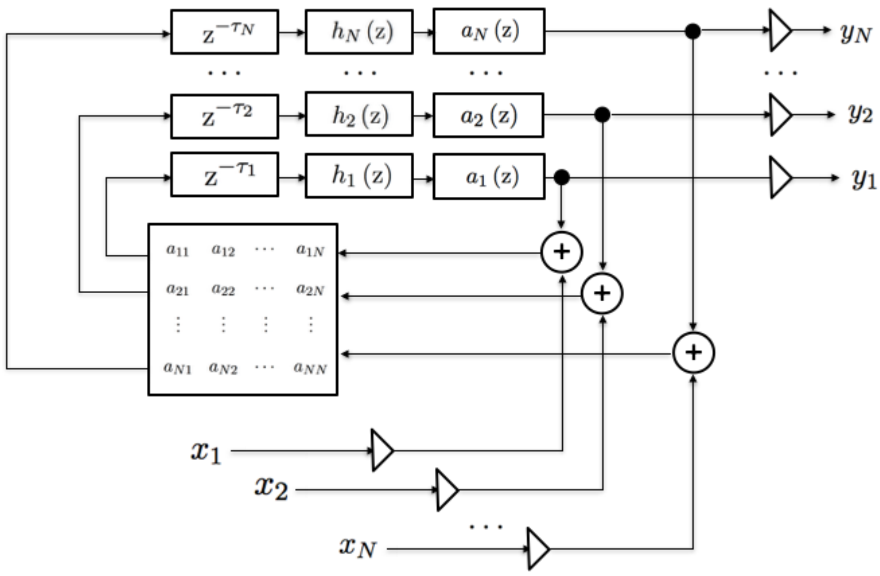

Figure 6: FDN processing structure

2.2.2 Convolution based reverberators

Convolving an anechoic input signal with an RIR reproduces the room reverberation and guarantees for an authentic and natural listening experience. This approach has been widely used in the last decades thanks to the increase of available processing power and the development of efficient frequency-domain filtering algorithms, such as the block-partitioned FFT convolution [7] and frequency delay lines [8]. The computational cost of this method depends on the length of the RIR, which in large concert halls and opera houses is on the order of a few seconds.

The convolution approach is usually restricted to static acoustic environments, as any change in the receiver and/or source positions results in a different RIR. In time-varying environments the RIRs have to be dynamically loaded. This requires a dense spatial sampling of the room and rapidly leads to an unmanageable amount of measurements. Therefore, convolution-based processors must provide controls allowing to modify the original RIR. A survey on available convolution-based reverberation processors shows that the control over the reverberation effect is, in general, limited to only a few low-level parameters, such as the early-to-reverb ratio and the decay time (with limited range). The range of possible transformations provided to sound engineers is therefore limited and does not include any control over the spatial behaviour.

IRCAM recently developed a 3-D convolution-based reverberation processor exploiting DRIRs that have been measured with higher-order spherical microphone arrays [2][34][35]. This method first extracts various acoustical and perceptual descriptors from the S-T-F representation of the DRIR. Then, in the re-synthesis, the S-T-F descriptors are applied to tune the DRIR without altering the microstructure of the original DRIR. This provides enhanced control parameters over the spatial parameters, such as the sound source *presence* and running *reverberance* [2].

2.2.3 Hybrid reverberation

Hybrid reverberation processors combine both convolution processing for the early part of the RIR and FDN for the late reverberation (see [3] for an overview on different algorithms). The hybridisation approach shows several advantages over full convolution processing: (a) Early reflections typically arrive within less than 50−200ms. Applying convolution filtering to this part of the RIR comes with a low computational cost, while it preserves the naturalness and spectral signature of the room response. (b) The late reverberation decay, which may be several seconds for large rooms, can be accurately modelled with computationally efficient FDNs. The feedback loop structure offers flexible control over the rendering parameters and can be adapted to perceptually-motivated control methods (see Section 2.2.4).

The two main challenges for the design of hybrid reverberation processors are: (a) to estimate model parameters from the original RIR for tuning the FDN, and (b) to guarantee the smooth transition between the two processing stages (i.e., the transition between early reflections and reverberation tail) without any perceptible artefacts.

Hybridisation requires adjusting the FDN model parameters (i.e. the reverberation profile and the initial frequency spectrum) to the time-frequency envelope of the original RIR. This is achieved by analysing the energy decay relief ([17][19]) of the original RIR (see Section A.1.2 for the estimation of FDN parameters and transition filters).

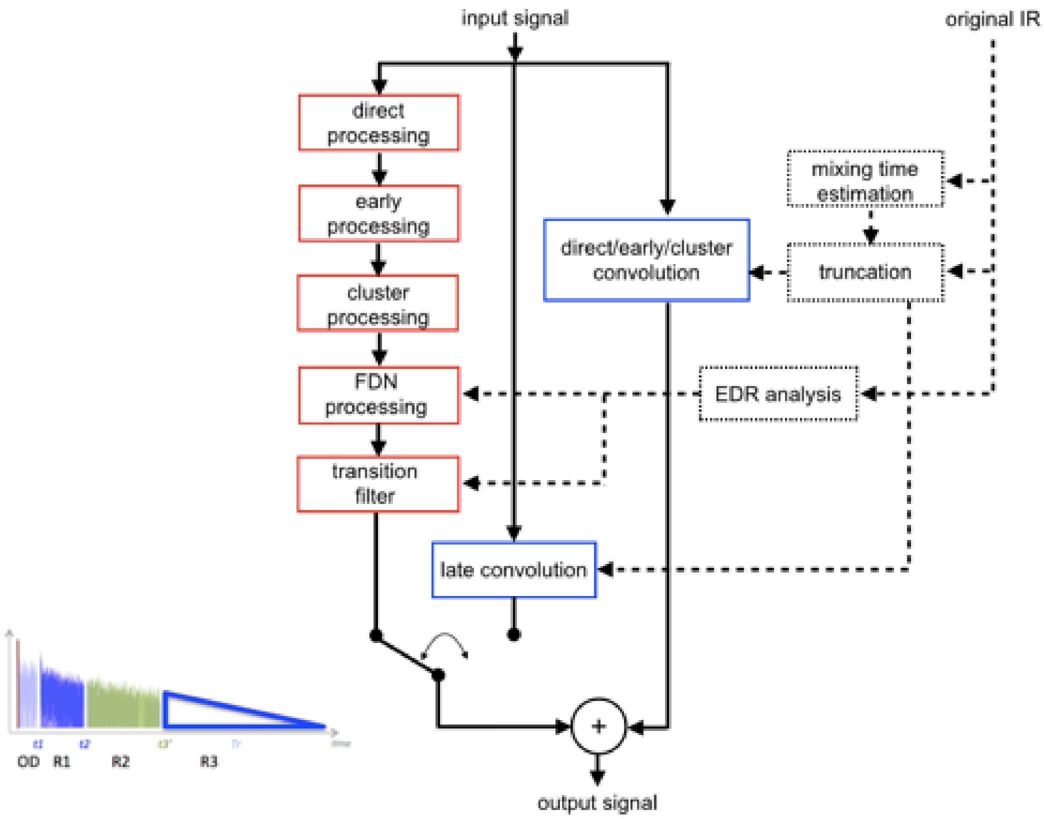

Figure 7: Hybrid reverberation processing structure that combines convolution processing (blue blocks) with parametric reverberation (red blocks). Blocks with dashed-lines indicate offline processing.

2.2.4 Perceptual control

In previous research, IRCAM has derived a set of nine mutually independent perceptual descriptors for the acoustic quality of concert halls (e.g., [20][21][23]). It has been shown that these descriptors correlate well with some objective room acoustic criteria [20]; e.g., the temporally extended direct sound energy (DirE) descriptor, which controls the perceived *presence* of a sound source in a reverberant environment, agrees well with Lochner and Burger's "energy ratio criterion" for speech intelligibility [27]. In order to control room effects along the relevant perceptual dimensions, most of the proposed descriptors require a temporal, spectral, and spatial weighting.

In its current implementation IRCAM's parametric FDN reverberation processor applies the simplified S-T-F model illustrated in Figure 5, and the perceptive control has been proven very useful in many music productions.

The notion of perceptual control can be extended to convolution-based and hybrid reverberation processors. The latter requires modifying the signal processing structure given in Figure 7 to the time segmented processing structure given in Figure 8. The convolution is split into three segments corresponding to the direct sound, early and cluster. In the case of a full-convolution reverb late reverberation is also considered as separate time segment. The outputs of the real-time convolution are first time-aligned using delays and then filtered with parametric three-band shelving filters that are controlled by the perceptual model parameters. This allows for a smooth modification of the RIR along the different perceptual dimensions.

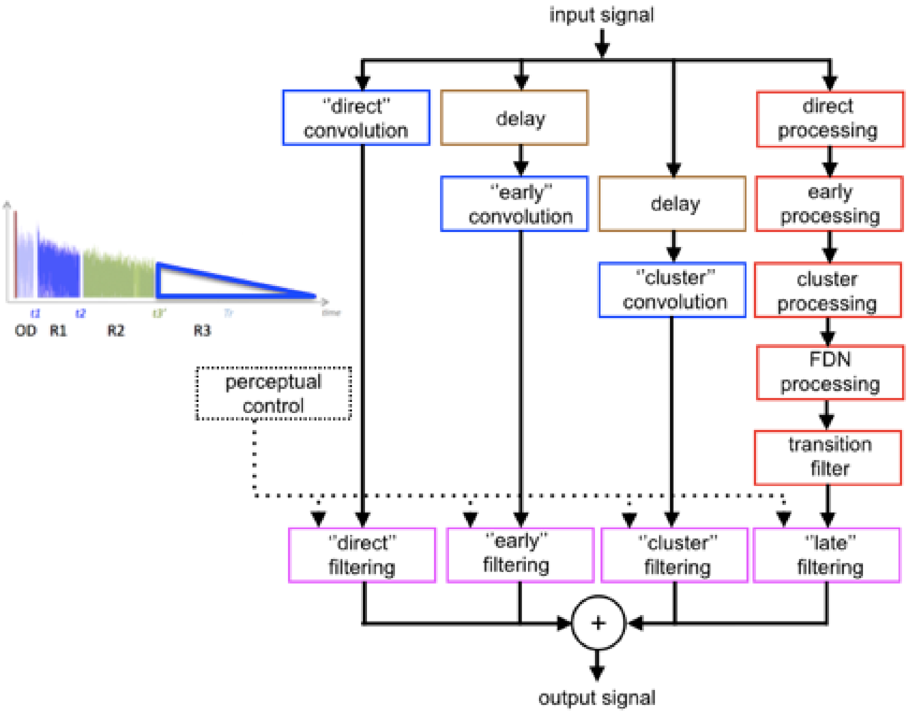

Figure 8: Hybrid reverberation with perceptual control processing structure that combines convolution processing (blue blocks) with parametric reverberation (red blocks). Delay lines (brown blocks) ensure time-alignment of the convolution segments and the IRCAM Spat~ perceptual model is implemented with three-band filters (magenta).

The MPEG-4 metadata scheme supports perceptual control parameters for room reverberation, which are based on the IRCAM Spat model.

3 Reverberation for object-based audio broadcasting

Spatial audio scenes can be represented by channel-based, scene-based or object-based approaches:

- *Channel-based* approaches mix the spatial audio scene for a target sound reproduction system and transmit a set of loudspeaker feed signals (e.g., Stereo, 5.1, etc.). This approach requires that the target loudspeaker positions are known during the mixing process. The spatial audio scene is fully encoded and cannot be modified at the decoder.
- *Scene-based* approaches decompose the spatial audio scene into a set of orthogonal basis functions. Then the encoded signals (e.g., the higher-order Ambisonics channels) are transmitted and decoded onto a target loudspeaker setup. This approach does not require that the target loudspeaker setup is known during the mixing process. The spatial audio scene is fully encoded; however, the decomposition gives some limited control over the sound scene in the decoder (e.g., scene rotation, zoom, scaling, and spatial windowing operators are available).
- *Object-based* approaches decompose the spatial audio scene into sound objects. These objects are then transmitted in separate channels each comprising the audio content and some metadata (e.g., the sound source position, aperture, and orientation). The decoder interprets the metadata and renders the audio scene to the target loudspeaker setup. This approach gives the greatest flexibility to modify the spatial audio scene at the decoder and allows the listener to interact with the audio scene.

The object-based approach is format agnostic and thus of particular interest for audio broadcasting. However, existing metadata schemes only provide very basic descriptors for the object properties, and in general do not support or only partly support reverberation. The following sections discuss different metadata schemes for the representation of a reverberation object, which are based on the discussion on reverberation processing in Section 2.

3.1 Metadata schemes for reverberation

The main advantage of object-based representations of audio scenes is that they are format agnostic and allow for creating interactive audio content. Therefore, object-based audio is widely used and well accepted in the VR and AR community, as scene graph modelling software usually provides a geometrical description of the virtual environment. This information is in general not available in post-production environments for digital broadcast (as it is typically not supported by software plug-ins and hardware devices). Also, sound engineers usually want to keep control over the audio quality of the final mix.

One possible solution is to separate the direct sound and the room reverberation signals, and send them to the end user over separated audio channels. The renderer at the end user device then guarantees for the highest available audio quality when mixing the audio signals to the playback device (e.g., binaural headphones, multichannel loudspeaker setup). Some of the reverberation models discussed in Section 2.1 can already be implemented in existing object-based metadata schemes.

The ITU Audio Definition Model (ITU-R BS.2076) [15] provides a diffuse parameter[4] and the audio object can have a size. The MPEG-H 3D Audio Standard, ISO/IEC 23008-3 (MPEG-H Part 3) [12]

[4] The parameter `audioBlockFormat.diffuse` (see ITU specifications [15], Section 5.4.3.3) can take values between 0.0 and 1.0 and describes the diffuseness of an audio object. No information is given on how this value should be interpreted and implemented. The diffuseness parameter is only available when the parent audio channel format uses the type definition “objects”.

supports a spread parameter. Non-standardised object-based metadata schemes, (such as SpatDIF [36] and the XML-based 3D audio scene metadata scheme [38]) provide some basic descriptors for room information. Recently, Coleman et al. proposed a reverberant spatial audio object (RSAO), which comprises metadata schemes for specular early reflections and late reverberation ([40][5]).

In general, object-based metadata schemes represent reverberation as

- a set of channel-based or scene-based audio signals, or
- a set of low-level and/or high-level descriptors.

The latter approach requires the reverberation to be synthesized in the audio renderer, e.g. by applying the rendering methods that have been discussed in Section 2.2. The main point of criticism on the parametric approach is that the audio quality strongly depends on the implementation of the reverberation synthesis in the decoder (which may significantly differ from device to device); consequently, the sound engineer loses control over a critical part of audio production.

With reference to Section 2, the following schemes can be used to represent reverberation in object-based broadcasting.

3.1.1 Signal-based approaches

The signal-based approaches (see [5]) are closely related to usual sound engineering practice. Reverberation is typically captured with room microphones and/or room microphone arrays during the recording. It can also be rendered with reverberation processors (see Section 2.2) during post-production (i.e. software plug-ins and external hardware devices). The reverberation signals are then monitored together with the recorded spot microphone and main microphone signals during the mixing and mastering process.

The channel-based and scene-based approaches transmit the mastered reverberation signals as separate audio channels, depending on the target output format (channel based or scene based). The renderer then mixes the “dry” audio objects with the corresponding reverberation channels to reproduce the spatial audio scene over the sound reproduction system. In the mastered audio scene, the spatial relations in between the audio objects and the reverberation are fixed; only the signal-to-reverb ratio (SRR) can be modified by the listener, which gives (very) limited control over the perceived sound source distance and source presence. The use of scene-based audio formats provides some additional control over the sound scene (e.g., rotations, scaling, etc.), with less restrictions on the loudspeaker setup (e.g., in HOA the number of loudspeakers depends on the reproduction order). However, user interaction is subject to the same restrictions as for channel-based audio formats.

An extended approach is to segment the mastered reverberation signals into different time sections that correspond to the early reflections, cluster, and late diffuse reverberation (see also Figure 5). The direct sounds are transmitted as audio objects, the “reverberation chunks” as separate audio channels depending on the chosen output format (channel-based or scene-based). To reproduce the spatial audio scene, the renderer then mixes the “dry” audio objects with the corresponding reverberation channels. Due to the segmentation into time sections the simplified S-T-F model (see Figure 5) can be applied to the reverberation synthesis, which allows the listener to modify the audio scene along the perceptual dimensions given in Section 2.2.4.

Figure 9 depicts a simple and generalised signal-based metadata scheme for reverberation in object-based audio using the time segmentation of the reverberation signals.

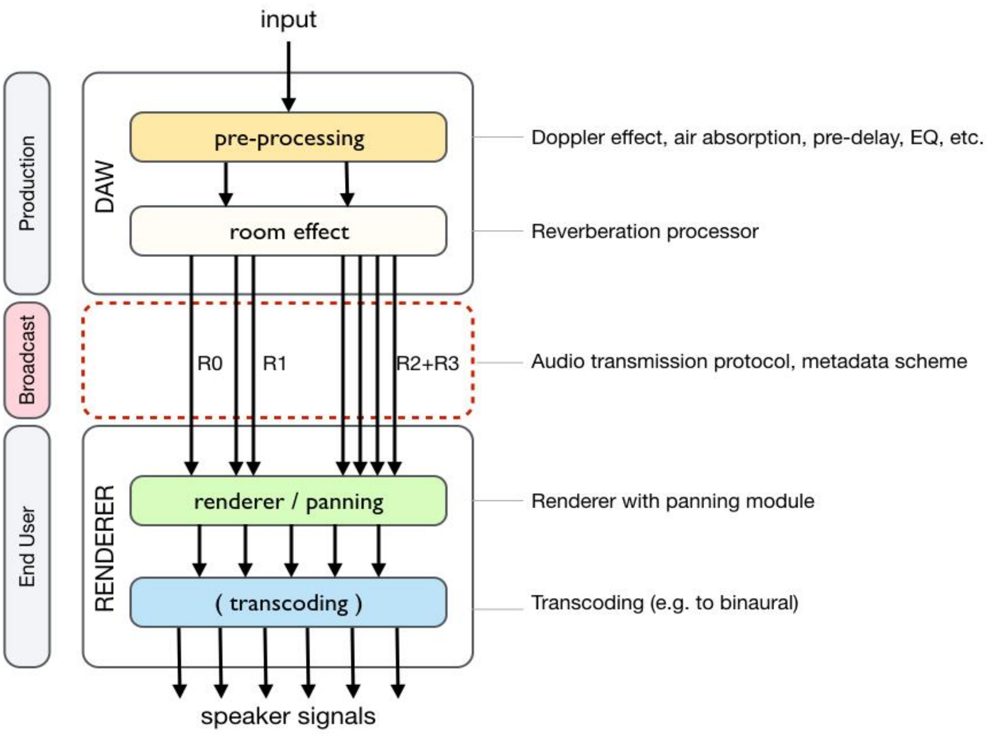

Figure 9: Generalised signal-based metadata scheme for reverberation rendering in 3D audio production. R0, R1, R2, and R3 refer to the direct sound, early reflections, late reflections, and late reverberation, respectively.

3.1.2 Parametric approaches

In Section 2, we discussed different methods for creating and representing reverberation signals using both low-level and high-level descriptors. Parametric approaches extract these descriptors during the post-production process and transmit them as metadata to control the reverberation synthesis in the renderer.

Low-level parameters

The simplest form of low-level reverberation parameters in object-based audio is to directly transmit the room impulse responses (RIRs, DRIRs), which are then interpreted in the renderer as FIR filters for convolution-based reverberation synthesis. Before transmission, the RIRs are often transcoded to the targeted channel-based or scene-based output format (e.g., DRIRs for 5.1 Surround).

Extensions of this approach try to increase the efficiency and to improve the user interactivity by parametrising the RIR ([28][29][50]) before it is transmitted to the renderer. These approaches have in common that they estimate the direction-of-arrival (DOA) of the strongest reflections in each S-T segment[5] of a DRIR that has been measured with microphone arrays. The renderer interprets them as plane waves (i.e. image sources), for which the DOA can be modified as a function of the position and orientation of the sound source. Assuming that the diffuse part of the reverberation is the same for all sources it remains unaltered.

High-level parameters

High-level parametric reverberation is synthesised in the renderer from a set of perceptual and/or room acoustical parameters. This approach has been applied in the widely-used MPEG-4 v2 [51]

[5] Strictly speaking, the approach in [28] estimates the highly correlated parts in each sub-band and time segment.

standard. The MPEG-4 perceptual parameters are based on the IRCAM Spat model ([20][21][23]), in which the renderer synthesises the reverberated audio scene by panning the direct sound according to the source direction (which is transmitted as metadata); the first early reflections are rendered as delayed versions of the direct sound and panned according to the source direction, but with an angular deviation of ±10 to ±30 degrees; the diffuse part is rendered by an FDN (see Section 2.2.1). As shown in Section A.1.2, the FDN parameters can be estimated from measured RIRs/DRIRs.

A hybrid solution (see Section 2.2.3) combines convolution-based reverberation processing for the early reflections part with FDN processing for the late reverberation part. This approach can be extended to high-level perceptual parameters (see Section 2.2.4), which give more control over the sound scene to the listener.

3.2 ADM metadata scheme as container format for reverberation

The ITU Audio Definition Model (ADM) [15] in its current implementation can be used as a container format for channel-based and scene-based approaches for representing reverberation in object-based broadcasting (see Section 3.1.1).

The ADM `audioPackFormat` groups together one or more `audioChannelFormats` that belong together for rendering purposes (e.g., stereo and 5.1 for the channel-based, and HOA for the scene-based formats). This is important when rendering the audio, as channels within the group may need to interact with each other. The `audioPackFormat` can also contain references to other packs to allow nesting. The `typeDefinition` is used to define the type of channels described within the audio pack. The `typeDefinition/typeLabel` must match those in the referred `audioChannelFormat`.

Currently, five different `typeDefinitions` are defined in ADM:

- `DirectSpeakers`: For channel-based audio, where each channel feeds a speaker directly.
- `Matrix`: For channel-based audio where channels are matrixed together.
- `Objects`: For object-based audio where channels represent audio objects (or parts of objects), so include positional information
- `HOA`: For scene-based audio where Ambisonics and HOA are used.
- `Binaural`: For binaural audio, where playback is over headphones.

Applying the generalised metadata scheme for reverberation rendering from Figure 9, channel-based and scene-based room reverberation can be represented in an ADM compatible “acoustic container” format as follows:

A. Channel-based / scene-based related approach

- [R0] is represented as a single-channel “direct sound” audio object with directional metadata that corresponds to the DOA of the direct sound.
- [R1, R2, R3] are grouped together and represented as a “room effect” audio pack that either contains M audio channels[6] of type `Objects` or N channels of type `HOA`.
- An ITU Audio Definition Model (ITU-R BS.2076) metadata scheme implementation of this approach is given in Appendix A.2.1.

[6] The IRCAM Spat model typically uses M=4 audio channels [18]. However, determining the number of transmitted audio channels is a tradeoff between the transmission channel bandwidth / available CPU power and the achievable diffuseness of the reproduced room reverberation effect.

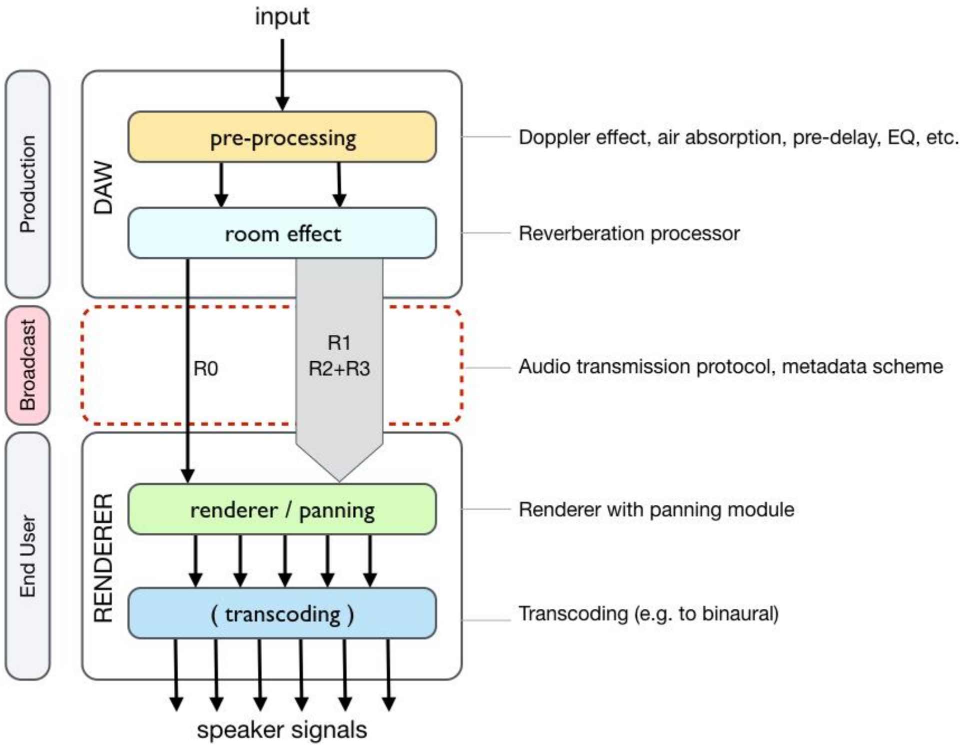

Figure 10: Acoustic container metadata scheme for the channel-based and scene-based mastering approach.

B. Intermediate approach

- [R0, R1] are grouped together and represented as a two-channel "extended direct sound" audio pack with directional metadata that corresponds to the DOA of the direct sound and the angular deviation of the DOA for the early reflections.
- [R2, R3] are grouped together and represented as a "late room effect" audio pack that either contains M audio channels of type `Objects` or N channels of type `HOA`.

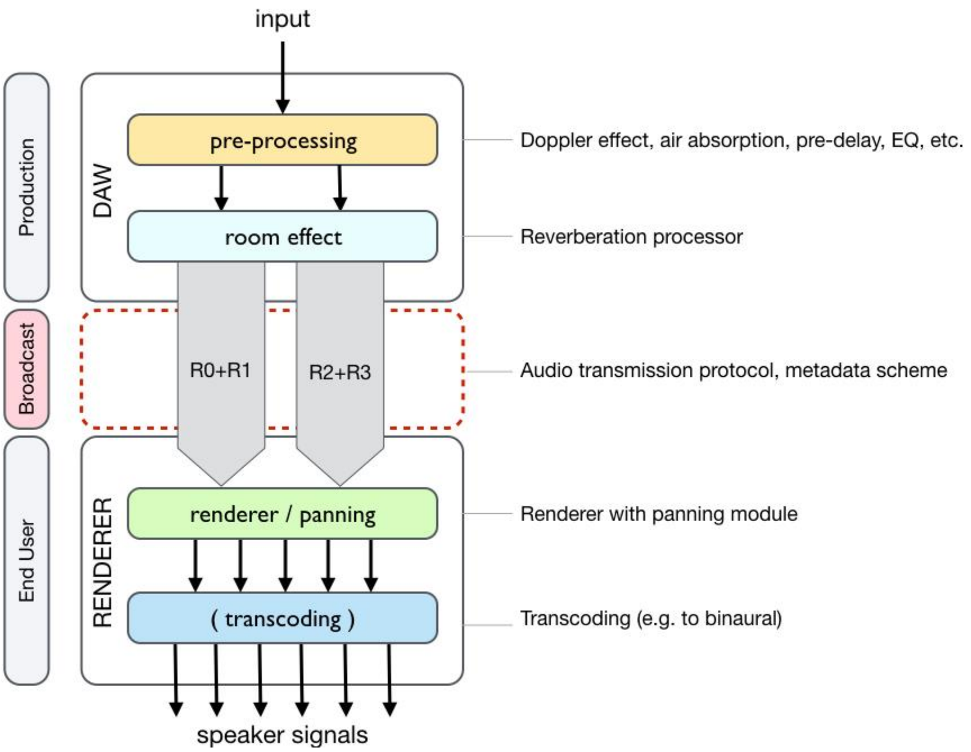

Figure 11: Acoustic container metadata scheme for the intermediate approach.

C. Spat-metadata compatible approach

- [R0] is represented as a single-channel "direct sound" audio object with directional metadata that corresponds to the DOA of the direct sound.
- [R1] is represented as a two-channel "early reflections" audio object with directional metadata that corresponds to the angular deviation of the DOA of the early reflections to that of the direct sound.
- [R2, R3] are grouped together and represented as a "late room effect" audio pack that either contains M audio channels of type `Objects` or N channels of type `HOA`.
- An ITU Audio Definition Model (ITU-R BS.2076) metadata scheme implementation of this approach is given in Appendix A.2.2.

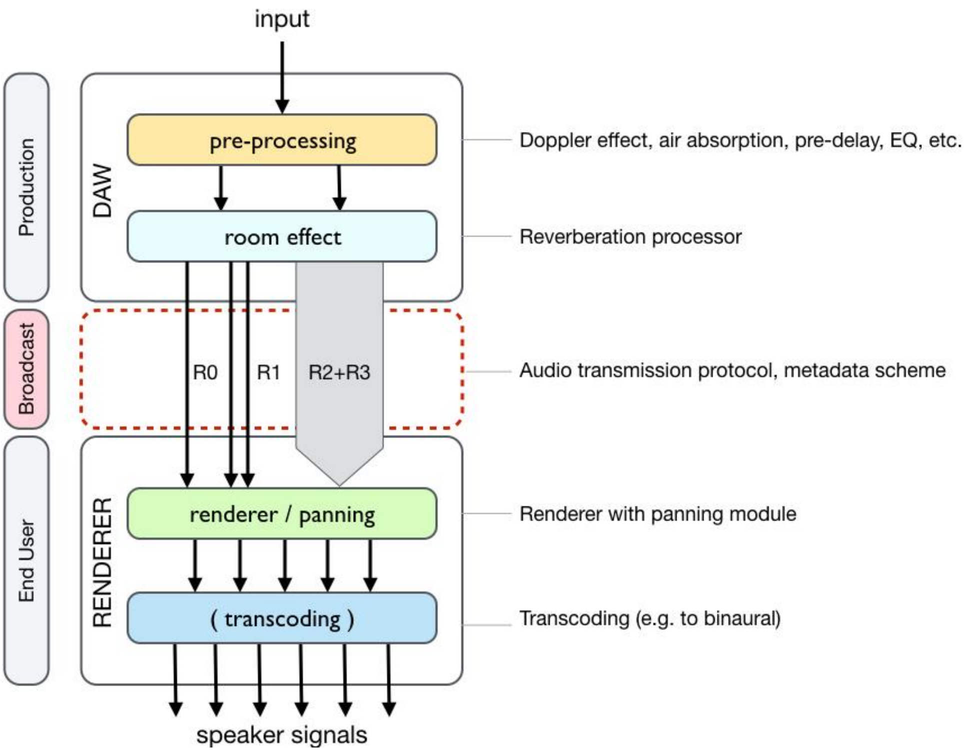

Figure 12: Acoustic container metadata scheme for the IRCAM Spat metadata model.

For audio scenes with multiple sound sources the audio pack for diffuse reverberation [R2+R3] can be shared among all sources, whereas the direct sound [R0] (case A), the direct sound plus early reflections [R0+R1] (case B), or direct sound [R0] and early reflections [R1] (case C) have to be transmitted separately for each sound source.

In order to limit the number of transmitted audio channels the audio pack for diffuse reverberation [R2+R3] could be reduced to one single audio channel, assuming that the renderer is able to de-correlate and matrix it to the playback channels for creating diffuse late reverberation (see also [41][42])

The "acoustic container" approach described in this section is format agnostic and compatible with the current version of the ITU Audio Definition Model. It delivers the same immersive listening experience over different audio reproduction systems. The result should be very close to what the sound engineer expects when mastering 3D audio in a studio environment.

Possible extensions include high-level control parameters that allow for an interactive control of the perceptually most relevant parameters of an audio scene (source distance, source rotation, etc.). However, this requires changes of the ADM metadata schemes in order to label the audio channels in the "acoustic container", such as:

- `typeDefinition = Reverberation`
- `typeLabel = DirectSound, EarlyReflections, LateReverberation`

The "acoustic container" approach is signal-based. The representation of parametric reverb metadata would be possible within the ADM structure, but also requires a new `typeDefinition`. Future work in the Orpheus project includes further extensions of the ADM metadata structure that will be presented to the ITU/EBU committee for future standardisation.

4 Conclusions

In this document, approaches for the processing and representation of room reverberation in object-based audio for broadcast were discussed. First prototype implementations of the proposed methods showed that the existing object-based audio metadata schemes (such as ADM and MPEG-H) need to be extended to support reverberation.

Future work includes the refinement of the parameter estimation for 3-D DRIRs measured with microphone arrays, the development of advanced metadata schemes for standardisation, and the perceptual evaluation of the overall quality of the synthesised reverberation in the renderer.

References

[1] Balmages, I., and Rafaely, B., “Room acoustics measurement by microphone arrays,” In 23rd IEEE Convention of the Electrical and Electronics Engineers in Israel, Beer-Sheva, Israel, (2004), 420–423. doi:10.1109/EEEI.2004.1361181

[2] Carpentier, T., Szpruch, T., Noisternig, M. and Warusfel, O., “Parametric control of convolution-based room simulators,” in Proc. of the Int. Symposium on Room Acoustics, Toronto, Canada, June 9-11 (2013)

[3] Carpentier, T., Noisternig, M., and Warusfel, O., “Hybrid Reverberation Processor with Perceptual Control,” Proceedings of International Conference on Digital Audio Effects (DAFx), Erlangen, Germany (2014), 93–100

[4] Carpentier, T., Noisternig, M., and Warusfel, O., “Twenty years of Ircam Spat: looking back, looking forward,” In International Computer Music Conference (ICMC), Denton, TX, USA (2015)

[5] Coleman, P., Franck, A., Jackson, P., Hughues, R., Remaggi, L., and Melchior, F., “On Object-Based Audio with Reverberation,” in Proc. 60th AES Int. Conf., Leuven, Belgium (2016)

[6] Defrance, G. and Polack, J.-D., “Estimating the mixing time of concert halls using the extensible Fourier Transform,” *Applied Acoustics*, 71: 9 (2010), 777–792

[7] Gardner, W. G., “Efficient convolution without input-output delay,” in Proc. of the 97th AES Convention, San Francisco, CA, Nov. 10-13 (1994)

[8] García, G., “Optimal filter partition for efficient convolution with short input/output delay,” in Proc. of the 113rd AES Convention, Los Angeles, CA, Oct. 5-8 (2002)

[9] Gover, B. N., Ryan, J. G., and Stinson, M. R., “Measurements of directional properties of reverberant sound fields in rooms using a spherical microphone array,” J Acoust Soc Am, 116 (2004), 2138–2148. doi:10.1121/1.1787525

[10] Guski, M. and Vorländer, M., “Comparison of Noise Compensation Methods for Room Acoustic Impulse Response Evaluations”, Acta Acust. United Ac., 100:2 (2014), 320-327

[11] Guidorzi, P., Barbaresi, L., D’Orazio, D., and Garai, M., “Impulse Responses Measured with MLS or Swept-Sine Signals Applied to Architectural Acoustics: An In-depth Analysis of the Two Methods and Some Case Studies of Measurements Inside Theaters,” Energy Procedia, 78 (2015), 1611–1616, doi:10.1016/j.egypro.2015.11.236

[12] Herre, J., Hilpert, J., Kuntz, A. and Plogsties, J., “MPEG-H 3D Audio – The New Standard for Coding of Immersive Spatial Audio,” IEEE J. Sel. Topics Sig. Proc.; 9:5 (2015), 770-779, doi:10.1109/jstsp.2015.2411578

[13] Hidaka, T., Yamada, Y. and Nakagawa, T., “A new definition of boundary point between early reflections and late reverberation in room impulse responses,” *J. Acoust. Soc. Am*, 122:1 (2007), 326 – 332

[14] Huopaniemi, J., Savioja, L. and Karjalainen, M., “Modeling of reflections and air absorption in acoustical spaces – a digital filter design approach,” in IEEE ASSP Workshop on Applications of Signal Processing to Audio and Acoustics (1997)

[15] ITU-R, “Recommendation BS.2076-0, Audio Definition Model,” International Telecommunication Union, June (2015)

[16] Jot, J.-M. and Chaigne, A., “Digital delay networks for de- signing artificial reverberators,” in Proc. of the 90th AES Convention, Paris, France, Feb. 19-22 (1991)

[17] Jot, J.-M., “An analysis/synthesis approach to real-time artificial reverberation,” In IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP-92), 2 (1992) 221–224, doi:10.1109/ICASSP.1992.226080

[18] Jot, J.-M., "Real-time spatial processing of sounds for music, multimedia and interactive human-computer interfaces", in ACM Multimedia Systems Journal, 7: 1 (1999)

[19] Jot, J.-M., Cerveau, L., and Warusfel, O., "Analysis and Synthesis of Room Reverberation Based on a Statistical Time-Frequency Model," AES 103rd Convention, New York, NY, USA (1997)

[20] Jullien, J.-P., "Structured model for the representation and the control of room acoustic quality," in 15th International Congress on Acoustics (ICA), Trondheim, Norway (1995), 517–520

[21] Kahle, E. and Jullien, J.-P., "Some new considerations on the subjective impression of reverberance and its correlation with objective criteria," in 127th annual Meeting of the Acoustical Society of America, Cambridge, MA, USA (1994), 239–242

[22] Kahle, E., "Validation d'un modèle objectif de la perception de la qualité acoustique dans un ensemble de salles de concerts et d'opéras", Ph.D. thesis, Laboratoire d'Acoustique de l'Université du Maine, Le Mans (1995)

[23] Kahle E., and Jullien, J.-P., "Subjective listening tests in concert halls: Methodology and results," in Proc. of the Int. Congr. on Acoustics, Trondheim, Norway (1995), 521 – 524

[24] Khaykin, D., and Rafaely, B. (2012). "Acoustic analysis by spherical microphone array processing of room impulse responses," J Acoust Soc Am, 132, 261–270. doi:10.1121/1.4726012

[25] Li, K. and Lam, P., "Prediction of reverberation time and speech transmission index in ling enclosures," J Acoust Soc Am, 117 (2005, 3716-3726

[26] Kuttruff, H., "Room Acoustics (4th ed.)", Spon Press (2000)

[27] Lochner, J. P. A. and Burger, J. F., "The subjective masking of short time delayed echoes by the primary sounds and their contribution to the intelligibility of speech.," Acustica, 8 (1958), 1-10

[28] Meerima, J. and Pulkki, V., "Spatial impulse response rendering I: analysis and synthesis," J. Audio Eng. Soc., 53:12 (2005), 1115-1127

[29] Melchior, F., Sladeczek, C., Partzsch, A., and Brix, S., "Design and implementation of an interactive room simulation for wave field synthesis, "in Proc. 40th AES Int. Conf., Tokyo, Japan (2010)

[30] Morgenstern, H., Noisternig, M., Rafaely, B., "Spatial analysis of sound fields in rooms using spherical MIMO systems," The Journal of the Acoustical Society of America, 137:4 (2015), 2195

[31] Morgenstern, H., Klein, J., Rafaely, B., and Noisternig, M., "Experimental investigation of multiple-input multiple-output systems for sound-field analysis," Presented at the International Congress on Acoustics (ICA), Buenos Aires, Argentina (2016)

[32] Morgenstern, H., Rafaely, B., and Noisternig, M., "Design framework for spherical microphone and loudspeaker arrays in a multiple-input multiple-output system," Submitted to J Acoust Soc Am (2016)

[33] Noisternig, M., Katz, B. F., Siltanen, S., and Savioja, L., "Framework for Real-Time Auralization in Architectural Acoustics," Acta Acust United Ac, 94 (2008), 1000–1015 doi:10.3813/AAA.918116

[34] Noisternig, M., Carpentier, T., and Warusfel, O., "Perceptual control of convolution based reverberators", J Acoust Soc Am, 131 (2012), 3215

[35] Noisternig, M., Carpentier, T., and Warusfel, O., "Denoising of Directional Room Impulse Responses Measured with Spherical Microphone Arrays," In 40th Annual German Congress on Acoustics (DAGA), Oldenburg, Germany, (2014) 600–601

[36] Peters, N., Lossius, T. and Schacher, J. C., "The Spatial Sound Description Interchange Format: Principles, Specifications, and Examples," Computer Music Journal, 37:1 (2013), 11-22, doi:10.1162/comj_a_00167

[37] Poletti, M. A., "Three-dimensional surround sound systems based on spherical harmonics," Journal of the Audio Engineering Society, 53:11 (2005) 1004–1025

[38] Potard, G. and Burnett, I., "An XML-based 3D audio metadata scheme," in Proc. 25th AES Int. Conf., London (2005), 17-19

[39] Pueo, B., Escolano, J., Roma, M., "Precise control of beam direction and beamwidth of linear loudspeaker arrays," in: IEEE Sensor Array and Multichannel Signal Processing Workshop (2004), 538–541

[40] Remaggi, L., Jackson, P. J. B., and Coleman, P., "Estimation of Room Reflection Parameters for a Reverberant Spatial Audio Object," in Proc. 138th Conv. Audio Eng. Soc., Warsaw (2015)

[41] Romblom, D., Depalle, P., Guastavino, C., and King, R., "Diffuse Field Modeling Using Physically-Inspired Decorrelation Filters and B-Format Microphones: Part I Algorithm," J Audio Eng Soc, 64:4 (2016), 177–193

[42] Rummukainen, O., Romblom, D., and Guastavino, C. "Diffuse Field Modeling Using Physically-Inspired Decorrelation Filters and B-Format Microphones: Part II Evaluation," J Audio Eng Soc, 64:4 (2016), 194–207

[43] Sakuma, T., Sakamoto, S. and Otsuru, T., "Computational Simulation in Architectural and Environmental Acoustics — Methods and Applications of Wave-Based Computation", Springer (2014), doi: 10.1007/978-4-431-54454-8

[44] Stautner, J. and Puckette, M., "Designing multi-channel reverberators", in Computer Music Journal, 6: 1 (1982)

[45] Savioja, L., and Svensson, U. P, "Overview of geometrical room acoustic modeling techniques," J Acoust Soc Am, 138:2 (2015), 708–730. doi:10.1121/1.4926438

[46] Schroeder M. R., "Eigenfrequenzstatistik und Anregungsstatistik in Räumen", Acustica, 4 (1954), 456-468; English translation in "Statistical parameters of the frequency response curves of large rooms", J Audio Eng Soc, 35 (1987), 299-306

[47] Schroeder M. R., "New method of measuring reverberation time", J Acoust Soc Am, 37 (1965), 409-412

[48] Stan, G.-B., Embrechts, J.-J., and Archambeau, D., "Comparison of Different Impulse Response Measurement Techniques," J Audio Eng Soc, 50 (2002), 249–262

[49] Tervo, S., Pätynen, J., and Lokki, T., "Acoustic reflection path tracing using a highly directional loudspeaker," In IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), (2009), 245–248

[50] Tervo, S., Pätynen, J., Kuusinen, A., and Lokki, T., "Spatial decomposition method for room impulse responses", J Audio Eng Soc, 61:1/2 (2013), 17-28

[51] Trivi, J.-M. and Jot, J.-M., "Rendering MPEG-4 AABIFS content through low-level cross-platform 3D audio API," in Proc. IEEE Conf. on Multimedia and Expo (ICME), Lausanne, Switzerland, 1 (2003), 513-516

[52] Väänänen, R., and Huopaniemi, J., "Advanced AudioBIFS: Virtual acoustics modelling in MPEG-4 scene description," IEEE Transactions on Multimedia, 6 (2004), 661–675

[53] Välimäki, V., Parker, J. D., Savioja, L., Smith, J. O., and Abel, J. S., "Fifty years of artificial reverberation," IEEE Audio, Speech, and Language Processing, 20: 5 (2012), 1421 – 1448

[54] Vorländer, M. "Auralization", Springer (2008)

Appendix A Additional info

A.1 RIR processing

The implementation of the above-mentioned reverberation processors requires some pre-processing of measured RIRs, as will be shown in the following sections.

A.1.1 RIR noise compensation

Measured RIRs are usually corrupted by measurement noise and environmental noise, which produce audible artefacts and bias the RIR parameter estimation process. Due to the exponential energy decay the noise floor mainly masks the late part of the RIR. The time-envelope of the energy and the noise floor can be estimated from the Energy Decay Curve (EDC), which is typically computed using the Schroeder integration method ([25][47]). Simple noise compensation methods truncate the RIR at the intersection time (noise crossover time) T_{lim} where the noise starts masking the exponentially decaying RIR (see also Figure 2). More advanced methods replace the tail of the residual room impulse response with an exponentially decaying zero-mean Gaussian noise process; the gain is adjusted to ensure a perfect transition at T_{lim}. A review of RIR noise compensation methods is given in [10].

Jot ([17][19]) proposed a T-F generalisation of the EDC, the so-called Energy Decay Relief (EDR). The EDR is the ensemble average over the T-F representation of the reverberation decay and evaluates the Schroeder backward integration in each frequency band. With reference to the stochastic model of reverberation decays, the T-F envelope is assumed to decay exponentially with time for each frequency f_ℓ and can be characterised by the initial power spectrum $P(\omega_\ell)$ and the reverberation time $T_R(\omega_\ell)$, where $\omega_\ell = 2\pi f_\ell$. The intersection time $T_{lim}(\omega_\ell)$ and noise power spectrum $P_n(\omega_\ell)$ can be iteratively estimated from the EDR ([17][19]). The noise-free late reverberation tail can be synthesised by filtering a zero-mean white Gaussian noise process with the estimated T-F envelope; the gain is adjusted in each frequency band to ensure a smooth transition at $T_{lim}(\omega_\ell)$. Figure 13 depicts the EDR of a measured RIR before and after noise compensation. The white dashed line shows the estimated intersection time in each frequency band.

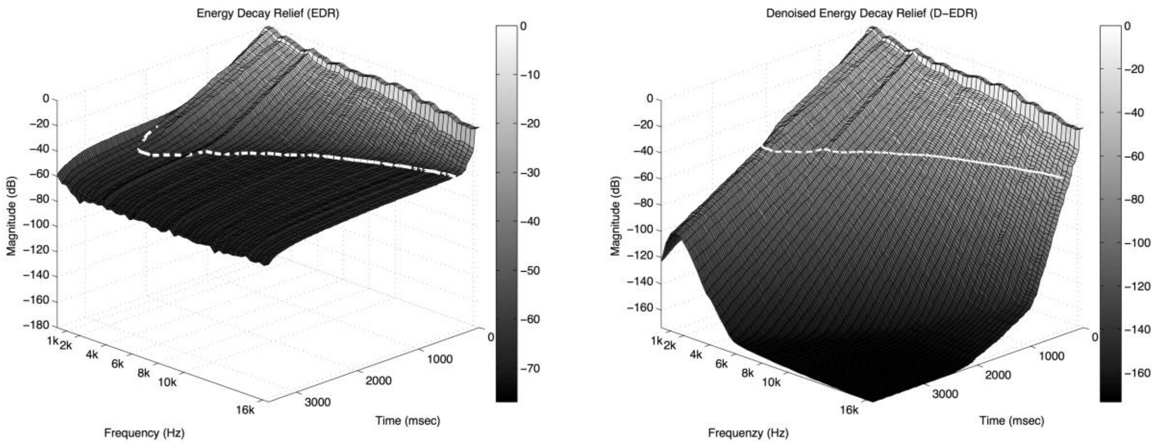

Figure 13: Energy decay relief (EDR) of a measured RIR with estimated intersection time (white dashed line) before (left figure) and after noise compensation (right figure)

Recently, Noisternig et al. [35] extended this approach to a S-T-F noise compensation method. The EDR is applied to the Fourier-Bessel expansion (FBE) of 3-D directional RIRs (DRIRs), which are typically measured with spherical microphone arrays (SMAs). The noisy tail of the DRIR is replaced by a spatially homogenous and isotropic noise field that is filtered by the estimated S-T-F envelope. Thus, not only the temporal but also the spatial characteristics of the late reverberation are considered for the noise compensation process.

A.1.2 FDN parameter estimation

The FDN parameters (see Section 2.2.1) can be estimated from the EDR. The damping filter in each feedback channel is chosen such that the logarithm of its magnitude response is proportional to the delay length and inversely proportional to the reverberation time. By neglecting the channel filter's phase response, the filter equation writes as

$$20\,log_{10}\left|g_i\left(e^{j2\pi f}\right)\right| = \frac{-60}{RT(f)}\tau_i$$

where τ_i is the delay length of the *i*-th channel, and $RT(f)$ is the estimated reverberation time for each frequency. IRCAM Spat~ currently implements the damping as parametric three-band filters with adjustable crossover frequencies. The estimated $RT(f)$ is thus averaged and reduced to three frequency bands. This reduction is motivated by the findings of Kahle [22] who showed that controlling the reverberation time in three frequency bands covers the full range of perceptual attributes for a large set of room acoustic qualities.

As mentioned above, the EDR is not only characterized by the reverberation time $RT(f)$ but also by the initial power spectrum $P(\omega_\ell)$. In theory, the FDN's initial power spectrum is a zero-mean white Gaussian random process that is independent of the energy decay envelope. Due to approximations in the FDN processing, such as the correction filters [16] and air absorption filters [14], this assumption does not always hold in practice and has to be corrected by an additional spectral correction filter. This additional filter aims at matching the initial spectrum of the FDN and the EDR of the measured RIR at the mixing time:

$$h_{corr}(f) = \sqrt{\frac{EDR_h(t_{mix}, f)}{EDR_{FDN}(t_{mix}, f)}}$$

From this equation, a linear-phase filter can be derived, which is then applied to the FDN. This filter guarantees for a smooth and continuous time-frequency envelope of the hybrid reverberation process at the mixing time.

Another essential parameter is the modal density D_m (i.e. the average number of modes per Hz) of the FDN with N feedback channels. It is related to the total length of the delay units by

$$D_m = \sum_{k=1}^{N} \tau_k$$

It can be shown that perceptually convincing artificial reverberation shall satisfy Schroeder's statistical model. With reference to [19] the modal overlap should be at least 3:1, which is equivalent to

$$D_m \geq RT_0$$

where RT_0 denotes the average reverberation time. The latter three equations can be used to adjust the delay times of the FDN's inner loop structure.

Figure 14 shows that the estimated RT and EDC of a measured RIR[7] and the hybrid reverberation synthesis are in good agreement. The FDN for the simulation consists of eight feedback channels and the crossover frequencies were set to 2.5 kHz and 7 kHz. For lower frequencies, an error of approximately $\pm$10% can be observed. This error results from the use of 2nd-order shelving filters in the feedback loop, which cannot approximate the $RT(f)$ of the measured RIR with sufficient accuracy. Higher-order filters are more accurate, but at the expense of a higher computational cost.

[7] The RIR was taken from a commercial data base.

The implemented three-band filter model provides a good trade-off between the model accuracy and the overall computational load.

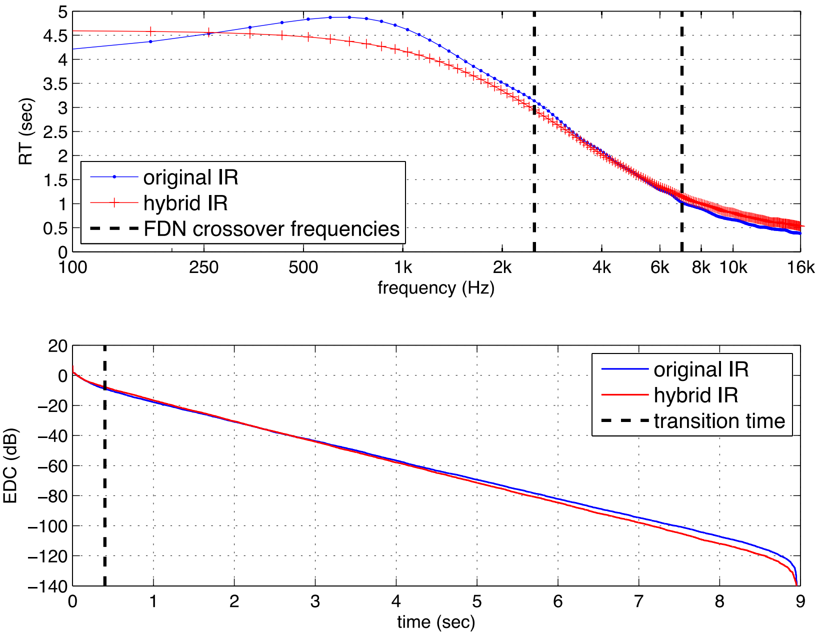

Figure 14: Estimated reverberation time (top) and energy decay curve (bottom) of a measured RIR and the hybrid FDN reverberation processor.

Figure 15 shows that the EDR of the measured and synthesised RIR are in good agreement, although the RT of the hybrid reverberation processor is too short in the frequency range from 500Hz to 1.5kHz. However, it becomes clear that the spectral correction filter shapes the spectrum of the FDN to match the spectrum of the RIR at the transition time.

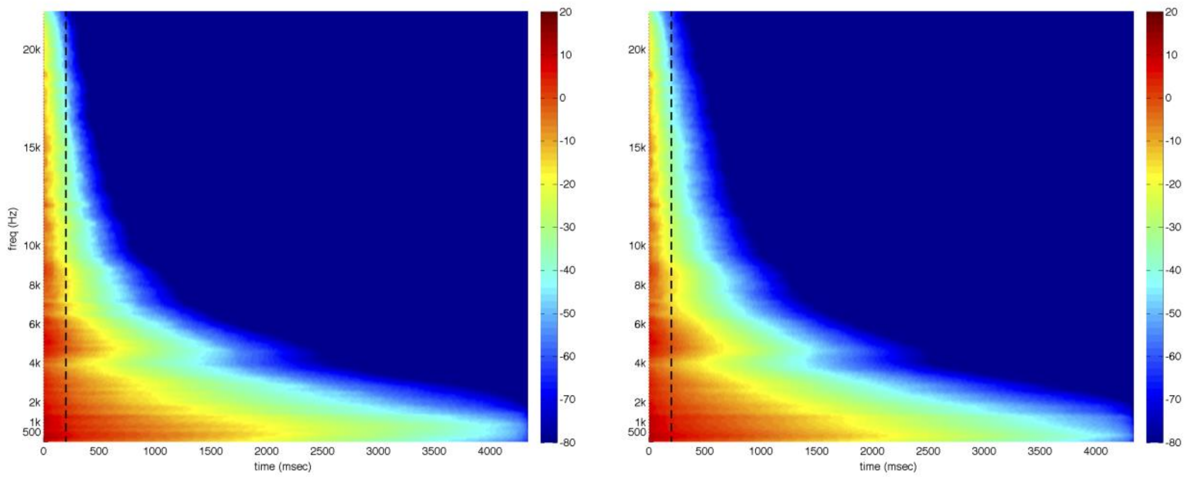

Figure 15: Energy decay relief of a measured RIR (left) and the hybrid FDN reverberation processor (right). The dashed line represents the transition time.

Informal listening tests confirm that the hybrid reverberation processor is perceptually indistinguishable from the direct convolution for various test signals (e.g., Dirac impulse, percussive sounds, male/female speech, music, etc.).

A.2 ITU Audio Definition Model (ITU-R BS.2076) metadata scheme implementation of reverberation

A.2.1 Channel-based / scene-based related approach

The following ADM metadata scheme represents the channel-based / scene-based related approach as in Section 3.2-A.

- [R0] is represented as a single-channel “direct sound” audio object with directional metadata that corresponds to the DOA of the direct sound.
- [R1, R2, R3] are grouped together and represented as a “room effect” audio pack that either contains M audio channels of type `Objects` or N channels of type `HOA`.

ADM metadata scheme:


```xml
<?xml version="1.0" encoding="UTF-8"?>
<ebuCoreMain xmlns:dc="http://purl.org/dc/elements/1.1/" xmlns="urn:ebu:metadata-schema:ebuCore_2015"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:ebu:metadata-schema:ebuCore_2015
https://www.ebu.ch/metadata/schemas/EBUCore/20150522/ebucore_20150522.xsd" xml:lang="en">
  <coreMetadata>
    <format>
      <audioFormatExtended>
        <audioProgramme audioProgrammeID="APR_1001" audioProgrammeName="Speech in a Cathedral">
          <audioContentIDRef>ACO_1001</audioContentIDRef>
        </audioProgramme>
        <audioContent audioContentID="ACO_1001" audioContentName="Speech in a Cathedral">
          <audioObjectIDRef>AO_1001</audioObjectIDRef>
          <audioObjectIDRef>AO_1002</audioObjectIDRef>
        </audioContent>
        <audioObject audioObjectID="AO_1001" audioObjectName="Speech" duration="00:05:00.00000" start="00:00:00.00000">
          <audioPackFormatIDRef>AP_00031001</audioPackFormatIDRef>
          <audioTrackUIDRef>ATU_00000001</audioTrackUIDRef>
        </audioObject>
        <audioObject audioObjectID="AO_1002" audioObjectName="SpeechReverb" duration="00:05:00.00000"
start="00:00:00.00000">
          <audioPackFormatIDRef>AP_00031002</audioPackFormatIDRef>
          <audioTrackUIDRef>ATU_00000002</audioTrackUIDRef>
          <audioTrackUIDRef>ATU_00000003</audioTrackUIDRef>
          <audioTrackUIDRef>ATU_00000004</audioTrackUIDRef>
          <audioTrackUIDRef>ATU_00000005</audioTrackUIDRef>
        </audioObject>
        <audioPackFormat audioPackFormatID="AP_00031001" audioPackFormatName="Speech" typeDefinition="Objects"
typeLabel="0003">
          <audioChannelFormatIDRef>AC_00031001</audioChannelFormatIDRef>
        </audioPackFormat>
        <audioPackFormat audioPackFormatID="AP_00031002" audioPackFormatName="SpeechQuadReverb"
typeDefinition="Objects" typeLabel="0003">
          <audioChannelFormatIDRef>AC_00031002</audioChannelFormatIDRef>
          <audioChannelFormatIDRef>AC_00031003</audioChannelFormatIDRef>
          <audioChannelFormatIDRef>AC_00031004</audioChannelFormatIDRef>
          <audioChannelFormatIDRef>AC_00031005</audioChannelFormatIDRef>
        </audioPackFormat>
        <audioChannelFormat audioChannelFormatID="AC_00031001" audioChannelFormatName="Speech"
typeDefinition="Objects" typeLabel="0003">
          <audioBlockFormat audioBlockFormatID="AB_00031001_00000001">
            <position coordinate="azimuth">0.000000</position>
            <position coordinate="elevation">0.000000</position>
            <position coordinate="distance">1.000000</position>
          </audioBlockFormat>
        </audioChannelFormat>
        <audioChannelFormat audioChannelFormatID="AC_00031002" audioChannelFormatName="SpeechReverbFrontLeft"
typeDefinition="Objects" typeLabel="0003">
          <audioBlockFormat audioBlockFormatID="AB_00031002_00000001">
            <position coordinate="azimuth">45.000000</position>
            <position coordinate="elevation">0.000000</position>
            <position coordinate="distance">1.000000</position>
          </audioBlockFormat>
        </audioChannelFormat>
        <audioChannelFormat audioChannelFormatID="AC_00031003" audioChannelFormatName="SpeechReverbFrontRight"
typeDefinition="Objects" typeLabel="0003">
          <audioBlockFormat audioBlockFormatID="AB_00031003_00000001">
```

```
      <position coordinate="azimuth">-45.000000</position>
      <position coordinate="elevation">0.000000</position>
      <position coordinate="distance">1.000000</position>
     </audioBlockFormat>
    </audioChannelFormat>
    <audioChannelFormat audioChannelFormatID="AC_00031004" audioChannelFormatName="SpeechReverbBackLeft"
typeDefinition="Objects" typeLabel="0003">
     <audioBlockFormat audioBlockFormatID="AB_00031004_00000001">
      <position coordinate="azimuth">135.000000</position>
      <position coordinate="elevation">0.000000</position>
      <position coordinate="distance">1.000000</position>
     </audioBlockFormat>
    </audioChannelFormat>
    <audioChannelFormat audioChannelFormatID="AC_00031005" audioChannelFormatName="SpeechReverbBackRight"
typeDefinition="Objects" typeLabel="0003">
     <audioBlockFormat audioBlockFormatID="AB_00031005_00000001">
      <position coordinate="azimuth">-135.000000</position>
      <position coordinate="elevation">0.000000</position>
      <position coordinate="distance">1.000000</position>
     </audioBlockFormat>
    </audioChannelFormat>
    <audioStreamFormat audioStreamFormatID="AS_00011001" audioStreamFormatName="PCM_Speech"
formatDefinition="PCM" formatLabel="0001">
     <audioChannelFormatIDRef>AC_00031001</audioChannelFormatIDRef>
     <audioTrackFormatIDRef>AT_00011001_01</audioTrackFormatIDRef>
    </audioStreamFormat>
    <audioStreamFormat audioStreamFormatID="AS_00011002" audioStreamFormatName="PCM_SpeechReverbFrontLeft"
formatDefinition="PCM" formatLabel="0001">
     <audioChannelFormatIDRef>AC_00031002</audioChannelFormatIDRef>
     <audioTrackFormatIDRef>AT_00011002_01</audioTrackFormatIDRef>
    </audioStreamFormat>
    <audioStreamFormat audioStreamFormatID="AS_00011003" audioStreamFormatName="PCM_SpeechReverbFrontRight"
formatDefinition="PCM" formatLabel="0001">
     <audioChannelFormatIDRef>AC_00031003</audioChannelFormatIDRef>
     <audioTrackFormatIDRef>AT_00011003_01</audioTrackFormatIDRef>
    </audioStreamFormat>
    <audioStreamFormat audioStreamFormatID="AS_00011004" audioStreamFormatName="PCM_SpeechReverbBackLeft"
formatDefinition="PCM" formatLabel="0001">
     <audioChannelFormatIDRef>AC_00031004</audioChannelFormatIDRef>
     <audioTrackFormatIDRef>AT_00011004_01</audioTrackFormatIDRef>
    </audioStreamFormat>
    <audioStreamFormat audioStreamFormatID="AS_00011005" audioStreamFormatName="PCM_SpeechReverbBackRight"
formatDefinition="PCM" formatLabel="0001">
     <audioChannelFormatIDRef>AC_00031005</audioChannelFormatIDRef>
     <audioTrackFormatIDRef>AT_00011005_01</audioTrackFormatIDRef>
    </audioStreamFormat>
    <audioTrackFormat audioTrackFormatID="AT_00011001_01" audioTrackFormatName="PCM_Speech"
formatDefinition="PCM" formatLabel="0001">
     <audioStreamFormatIDRef>AS_00011001</audioStreamFormatIDRef>
    </audioTrackFormat>
    <audioTrackFormat audioTrackFormatID="AT_00011002_01" audioTrackFormatName="PCM_SpeechReverbFrontLeft"
formatDefinition="PCM" formatLabel="0001">
     <audioStreamFormatIDRef>AS_00011002</audioStreamFormatIDRef>
    </audioTrackFormat>
    <audioTrackFormat audioTrackFormatID="AT_00011003_01" audioTrackFormatName="PCM_SpeechReverbFrontRight"
formatDefinition="PCM" formatLabel="0001">
     <audioStreamFormatIDRef>AS_00011003</audioStreamFormatIDRef>
    </audioTrackFormat>
    <audioTrackFormat audioTrackFormatID="AT_00011004_01" audioTrackFormatName="PCM_SpeechReverbBackLeft"
formatDefinition="PCM" formatLabel="0001">
     <audioStreamFormatIDRef>AS_00011004</audioStreamFormatIDRef>
    </audioTrackFormat>
    <audioTrackFormat audioTrackFormatID="AT_00011005_01" audioTrackFormatName="PCM_SpeechReverbBackRight"
formatDefinition="PCM" formatLabel="0001">
     <audioStreamFormatIDRef>AS_00011005</audioStreamFormatIDRef>
    </audioTrackFormat>
    <audioTrackUID UID="ATU_00000001" bitDepth="24" sampleRate="48000">
     <audioTrackFormatIDRef>AT_00011001_01</audioTrackFormatIDRef>
     <audioPackFormatIDRef>AP_00031001</audioPackFormatIDRef>
    </audioTrackUID>
    <audioTrackUID UID="ATU_00000002" bitDepth="24" sampleRate="48000">
     <audioTrackFormatIDRef>AT_00011002_01</audioTrackFormatIDRef>
     <audioPackFormatIDRef>AP_00031002</audioPackFormatIDRef>
    </audioTrackUID>
    <audioTrackUID UID="ATU_00000003" bitDepth="24" sampleRate="48000">
     <audioTrackFormatIDRef>AT_00011003_01</audioTrackFormatIDRef>
     <audioPackFormatIDRef>AP_00031002</audioPackFormatIDRef>
    </audioTrackUID>
    <audioTrackUID UID="ATU_00000004" bitDepth="24" sampleRate="48000">
```

```
        <audioTrackFormatIDRef>AT_00011004_01</audioTrackFormatIDRef>
        <audioPackFormatIDRef>AP_00031002</audioPackFormatIDRef>
      </audioTrackUID>
      <audioTrackUID UID="ATU_00000005" bitDepth="24" sampleRate="48000">
        <audioTrackFormatIDRef>AT_00011005_01</audioTrackFormatIDRef>
        <audioPackFormatIDRef>AP_00031002</audioPackFormatIDRef>
      </audioTrackUID>
    </audioFormatExtended>
  </format>
 </coreMetadata>
</ebuCoreMain>
```

A.2.2 Spat metadata compatible approach

The following ADM metadata scheme represents the channel-based / scene-based related approach as in Section 3.2-C.

- [R0] is represented as a single-channel “direct sound” audio object with directional metadata that corresponds to the DOA of the direct sound.
- [R1] is represented as a two-channel “early reflections” audio object with directional metadata that corresponds to the angular deviation of the DOA of the direct sound.
- [R2, R3] are grouped together and represented as a “late room effect” audio pack that either contains M audio channels of type `Objects` or N channels of type `HOA`.

ADM metadata scheme:

```
<?xml version="1.0" encoding="UTF-8"?>
<ebuCoreMain xmlns:dc="http://purl.org/dc/elements/1.1/" xmlns="urn:ebu:metadata-schema:ebuCore_2015"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:ebu:metadata-schema:ebuCore_2015
https://www.ebu.ch/metadata/schemas/EBUCore/20150522/ebucore_20150522.xsd" xml:lang="en">
 <coreMetadata>
  <format>
    <audioFormatExtended>
      <audioProgramme audioProgrammeID="APR_1001" audioProgrammeName="Speech in a Cathedral">
        <audioContentIDRef>ACO_1001</audioContentIDRef>
      </audioProgramme>
      <audioContent audioContentID="ACO_1001" audioContentName="Speech in a Cathedral">
        <audioObjectIDRef>AO_1001</audioObjectIDRef>
        <audioObjectIDRef>AO_1002</audioObjectIDRef>
        <audioObjectIDRef>AO_1003</audioObjectIDRef>
      </audioContent>
      <audioObject audioObjectID="AO_1001" audioObjectName="Speech" duration="00:05:00.00000" start="00:00:00.00000">
        <audioPackFormatIDRef>AP_00031001</audioPackFormatIDRef>
        <audioTrackUIDRef>ATU_00000001</audioTrackUIDRef>
      </audioObject>
      <audioObject audioObjectID="AO_1003" audioObjectName="SpeechEarlyReflections" duration="00:05:00.00000"
start="00:00:00.00000">
        <audioPackFormatIDRef>AP_00031003</audioPackFormatIDRef>
        <audioTrackUIDRef>ATU_00000002</audioTrackUIDRef>
        <audioTrackUIDRef>ATU_00000003</audioTrackUIDRef>
      </audioObject>
      <audioObject audioObjectID="AO_1002" audioObjectName="SpeechReverb" duration="00:05:00.00000"
start="00:00:00.00000">
        <audioPackFormatIDRef>AP_00031002</audioPackFormatIDRef>
        <audioTrackUIDRef>ATU_00000004</audioTrackUIDRef>
        <audioTrackUIDRef>ATU_00000005</audioTrackUIDRef>
        <audioTrackUIDRef>ATU_00000006</audioTrackUIDRef>
        <audioTrackUIDRef>ATU_00000007</audioTrackUIDRef>
      </audioObject>
      <audioPackFormat audioPackFormatID="AP_00031001" audioPackFormatName="Speech" typeDefinition="Objects"
typeLabel="0003">
        <audioChannelFormatIDRef>AC_00031001</audioChannelFormatIDRef>
      </audioPackFormat>
      <audioPackFormat audioPackFormatID="AP_00031003" audioPackFormatName="SpeechEarlyReflections"
typeDefinition="Objects" typeLabel="0003">
        <audioChannelFormatIDRef>AC_00031006</audioChannelFormatIDRef>
        <audioChannelFormatIDRef>AC_00031007</audioChannelFormatIDRef>
      </audioPackFormat>
      <audioPackFormat audioPackFormatID="AP_00031002" audioPackFormatName="SpeechQuadReverb"
typeDefinition="Objects" typeLabel="0003">
        <audioChannelFormatIDRef>AC_00031002</audioChannelFormatIDRef>
```

```xml
        <audioChannelFormatIDRef>AC_00031003</audioChannelFormatIDRef>
        <audioChannelFormatIDRef>AC_00031004</audioChannelFormatIDRef>
        <audioChannelFormatIDRef>AC_00031005</audioChannelFormatIDRef>
      </audioPackFormat>
      <audioChannelFormat audioChannelFormatID="AC_00031001" audioChannelFormatName="Speech"
typeDefinition="Objects" typeLabel="0003">
        <audioBlockFormat audioBlockFormatID="AB_00031001_00000001">
          <position coordinate="azimuth">0.000000</position>
          <position coordinate="elevation">0.000000</position>
          <position coordinate="distance">1.000000</position>
        </audioBlockFormat>
      </audioChannelFormat>
      <audioChannelFormat audioChannelFormatID="AC_00031006" audioChannelFormatName="SpeechEarlyLeft"
typeDefinition="Objects" typeLabel="0003">
        <audioBlockFormat audioBlockFormatID="AB_00031006_00000001">
          <position coordinate="azimuth">20.000000</position>
          <position coordinate="elevation">0.000000</position>
          <position coordinate="distance">1.000000</position>
        </audioBlockFormat>
      </audioChannelFormat>
      <audioChannelFormat audioChannelFormatID="AC_00031007" audioChannelFormatName="SpeechEarlyRight"
typeDefinition="Objects" typeLabel="0003">
        <audioBlockFormat audioBlockFormatID="AB_00031007_00000001">
          <position coordinate="azimuth">-20.000000</position>
          <position coordinate="elevation">0.000000</position>
          <position coordinate="distance">1.000000</position>
        </audioBlockFormat>
      </audioChannelFormat>
      <audioChannelFormat audioChannelFormatID="AC_00031002" audioChannelFormatName="SpeechReverbFrontLeft"
typeDefinition="Objects" typeLabel="0003">
        <audioBlockFormat audioBlockFormatID="AB_00031002_00000001">
          <position coordinate="azimuth">45.000000</position>
          <position coordinate="elevation">0.000000</position>
          <position coordinate="distance">1.000000</position>
        </audioBlockFormat>
      </audioChannelFormat>
      <audioChannelFormat audioChannelFormatID="AC_00031003" audioChannelFormatName="SpeechReverbFrontRight"
typeDefinition="Objects" typeLabel="0003">
        <audioBlockFormat audioBlockFormatID="AB_00031003_00000001">
          <position coordinate="azimuth">-45.000000</position>
          <position coordinate="elevation">0.000000</position>
          <position coordinate="distance">1.000000</position>
        </audioBlockFormat>
      </audioChannelFormat>
      <audioChannelFormat audioChannelFormatID="AC_00031004" audioChannelFormatName="SpeechReverbBackLeft"
typeDefinition="Objects" typeLabel="0003">
        <audioBlockFormat audioBlockFormatID="AB_00031004_00000001">
          <position coordinate="azimuth">135.000000</position>
          <position coordinate="elevation">0.000000</position>
          <position coordinate="distance">1.000000</position>
        </audioBlockFormat>
      </audioChannelFormat>
      <audioChannelFormat audioChannelFormatID="AC_00031005" audioChannelFormatName="SpeechReverbBackRight"
typeDefinition="Objects" typeLabel="0003">
        <audioBlockFormat audioBlockFormatID="AB_00031005_00000001">
          <position coordinate="azimuth">-135.000000</position>
          <position coordinate="elevation">0.000000</position>
          <position coordinate="distance">1.000000</position>
        </audioBlockFormat>
      </audioChannelFormat>
      <audioStreamFormat audioStreamFormatID="AS_00011001" audioStreamFormatName="PCM_Speech"
formatDefinition="PCM" formatLabel="0001">
        <audioChannelFormatIDRef>AC_00031001</audioChannelFormatIDRef>
        <audioTrackFormatIDRef>AT_00011001_01</audioTrackFormatIDRef>
      </audioStreamFormat>
      <audioStreamFormat audioStreamFormatID="AS_00011006" audioStreamFormatName="PCM_SpeechEarlyLeft"
formatDefinition="PCM" formatLabel="0001">
        <audioChannelFormatIDRef>AC_00031006</audioChannelFormatIDRef>
        <audioTrackFormatIDRef>AT_00011006_01</audioTrackFormatIDRef>
      </audioStreamFormat>
      <audioStreamFormat audioStreamFormatID="AS_00011007" audioStreamFormatName="PCM_SpeechEarlyRight"
formatDefinition="PCM" formatLabel="0001">
        <audioChannelFormatIDRef>AC_00031007</audioChannelFormatIDRef>
        <audioTrackFormatIDRef>AT_00011007_01</audioTrackFormatIDRef>
      </audioStreamFormat>
      <audioStreamFormat audioStreamFormatID="AS_00011002" audioStreamFormatName="PCM_SpeechReverbFrontLeft"
formatDefinition="PCM" formatLabel="0001">
        <audioChannelFormatIDRef>AC_00031002</audioChannelFormatIDRef>
        <audioTrackFormatIDRef>AT_00011002_01</audioTrackFormatIDRef>
```

```
    </audioStreamFormat>
    <audioStreamFormat audioStreamFormatID="AS_00011003" audioStreamFormatName="PCM_SpeechReverbFrontRight"
formatDefinition="PCM" formatLabel="0001">
     <audioChannelFormatIDRef>AC_00031003</audioChannelFormatIDRef>
     <audioTrackFormatIDRef>AT_00011003_01</audioTrackFormatIDRef>
    </audioStreamFormat>
    <audioStreamFormat audioStreamFormatID="AS_00011004" audioStreamFormatName="PCM_SpeechReverbBackLeft"
formatDefinition="PCM" formatLabel="0001">
     <audioChannelFormatIDRef>AC_00031004</audioChannelFormatIDRef>
     <audioTrackFormatIDRef>AT_00011004_01</audioTrackFormatIDRef>
    </audioStreamFormat>
    <audioStreamFormat audioStreamFormatID="AS_00011005" audioStreamFormatName="PCM_SpeechReverbBackRight"
formatDefinition="PCM" formatLabel="0001">
     <audioChannelFormatIDRef>AC_00031005</audioChannelFormatIDRef>
     <audioTrackFormatIDRef>AT_00011005_01</audioTrackFormatIDRef>
    </audioStreamFormat>
    <audioTrackFormat audioTrackFormatID="AT_00011001_01" audioTrackFormatName="PCM_Speech"
formatDefinition="PCM" formatLabel="0001">
     <audioStreamFormatIDRef>AS_00011001</audioStreamFormatIDRef>
    </audioTrackFormat>
    <audioTrackFormat audioTrackFormatID="AT_00011006_01" audioTrackFormatName="PCM_SpeechEarlyLeft"
formatDefinition="PCM" formatLabel="0001">
     <audioStreamFormatIDRef>AS_00011006</audioStreamFormatIDRef>
    </audioTrackFormat>
    <audioTrackFormat audioTrackFormatID="AT_00011007_01" audioTrackFormatName="PCM_SpeechEarlyRight"
formatDefinition="PCM" formatLabel="0001">
     <audioStreamFormatIDRef>AS_00011007</audioStreamFormatIDRef>
    </audioTrackFormat>
    <audioTrackFormat audioTrackFormatID="AT_00011002_01" audioTrackFormatName="PCM_SpeechReverbFrontLeft"
formatDefinition="PCM" formatLabel="0001">
     <audioStreamFormatIDRef>AS_00011002</audioStreamFormatIDRef>
    </audioTrackFormat>
    <audioTrackFormat audioTrackFormatID="AT_00011003_01" audioTrackFormatName="PCM_SpeechReverbFrontRight"
formatDefinition="PCM" formatLabel="0001">
     <audioStreamFormatIDRef>AS_00011003</audioStreamFormatIDRef>
    </audioTrackFormat>
    <audioTrackFormat audioTrackFormatID="AT_00011004_01" audioTrackFormatName="PCM_SpeechReverbBackLeft"
formatDefinition="PCM" formatLabel="0001">
     <audioStreamFormatIDRef>AS_00011004</audioStreamFormatIDRef>
    </audioTrackFormat>
    <audioTrackFormat audioTrackFormatID="AT_00011005_01" audioTrackFormatName="PCM_SpeechReverbBackRight"
formatDefinition="PCM" formatLabel="0001">
     <audioStreamFormatIDRef>AS_00011005</audioStreamFormatIDRef>
    </audioTrackFormat>
    <audioTrackUID UID="ATU_00000001" bitDepth="24" sampleRate="48000">
     <audioTrackFormatIDRef>AT_00011001_01</audioTrackFormatIDRef>
     <audioPackFormatIDRef>AP_00031001</audioPackFormatIDRef>
    </audioTrackUID>
    <audioTrackUID UID="ATU_00000002" bitDepth="24" sampleRate="48000">
     <audioTrackFormatIDRef>AT_00011006_01</audioTrackFormatIDRef>
     <audioPackFormatIDRef>AP_00031003</audioPackFormatIDRef>
    </audioTrackUID>
    <audioTrackUID UID="ATU_00000003" bitDepth="24" sampleRate="48000">
     <audioTrackFormatIDRef>AT_00011007_01</audioTrackFormatIDRef>
     <audioPackFormatIDRef>AP_00031003</audioPackFormatIDRef>
    </audioTrackUID>
    <audioTrackUID UID="ATU_00000004" bitDepth="24" sampleRate="48000">
     <audioTrackFormatIDRef>AT_00011002_01</audioTrackFormatIDRef>
     <audioPackFormatIDRef>AP_00031002</audioPackFormatIDRef>
    </audioTrackUID>
    <audioTrackUID UID="ATU_00000005" bitDepth="24" sampleRate="48000">
     <audioTrackFormatIDRef>AT_00011003_01</audioTrackFormatIDRef>
     <audioPackFormatIDRef>AP_00031002</audioPackFormatIDRef>
    </audioTrackUID>
    <audioTrackUID UID="ATU_00000006" bitDepth="24" sampleRate="48000">
     <audioTrackFormatIDRef>AT_00011004_01</audioTrackFormatIDRef>
     <audioPackFormatIDRef>AP_00031002</audioPackFormatIDRef>
    </audioTrackUID>
    <audioTrackUID UID="ATU_00000007" bitDepth="24" sampleRate="48000">
     <audioTrackFormatIDRef>AT_00011005_01</audioTrackFormatIDRef>
     <audioPackFormatIDRef>AP_00031002</audioPackFormatIDRef>
    </audioTrackUID>
   </audioFormatExtended>
  </format>
 </coreMetadata>
</ebuCoreMain>
```