



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



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

## **D5.1: Document on user requirements**

Version: v1.2

Deliverable type	R (Document, report)
Dissemination level	PU (Public)
Due date	30/09/2016
Submission date	14/10/2016
Lead editor	Olivier Warusfel (IRCAM)
Authors	Mike Armstrong (BBC), Chris Baume (BBC), Werner Bleisteiner (BR), Niels Bogaards (ElephantCandy), Nicolas Epain (b<>com), Benjamin Duval (Trinnov), Andrew Mason (BBC), Martin Ragot (b<>com), Andreas Silzle (FhG), Michael Weitnauer (IRT)
Reviewers	Chris Baume (BBC), Thibaut Carpentier (IRCAM)
Work package, Task	WP5
Keywords	Presentation, Technical requirements, End-user's requirements

---

### *Abstract*

This document aims to identify the main technical and end-user requirements for the reception, presentation and personalised consumption of broadcast object-based content. Its purpose is to serve as guidelines for the design, implementation and assessment of technical solutions that will be developed during the project. Usability specifications are exemplified through various mock-ups and user scenarios. Several hardware and software solutions are considered for the end-user device in order to cover different audio content consumption situations (e.g. domestic use vs mobility). Personalisation and interactivity features are also listed and will require the design and development of user interfaces handling various input devices (e.g. touch screen, GPS sensors, microphone...).

---

### Document revision history

Version	Date	Description of change	List of contributor(s)
V0.0	25/04/2016	Initial structure	Olivier Warusfel
V0.1	16/06/2016	Reception macroblock description + 1.1.3, 1.3.2 updated	Chris Baume, Niels Bogaards, Michael Weitnauer, Olivier Warusfel
V0.2	22/06/2016	1.1.2, 1.1.3, 1.4 updated	Werner Bleistener, Benjamin Duval
V0.3	30/06/2016	Modified structure + 1.1.3, 1.3.2, Appendix update	Mike Armstrong, Niels Bogaards, Martin Ragot, Olivier Warusfel
V0.4	19/07/2016	1.1, 1.2 and 1.3 update	Werner Bleisteiner, Niels Bogaards, Benjamin Duval
V0.5	18/08/2016	1.1.3 update	Benjamin Duval, Olivier Warusfel
V0.6	25/08/2016	1.1.3, 1.4.1, 1.2 updates	Andrew Mason, Werner Bleisteiner, Nikolaus Färber, Andreas Silzle
V0.7	13/09/2016	Update of section 1, modification of the document structure.	Michael Weitnauer, Benjamin Duval, Olivier Warusfel
V0.8-1	25/09/2016	Corrections, Executive Summary and conclusion	Andreas Silzle, Niels Bogaards, Olivier Warusfel
V0.8-2	26/09/2016	review	reviewers
V0.9	29/09/2016	Integrate reviews	Olivier Warusfel
V1.0	30/09/2016	Editing	Olivier Warusfel
V1.1	10/10/2016	Integrate minor corrections	Olivier Warusfel
V1.2	11/10/2016	Minor corrections in Executive Summary and figure 1.	Olivier Warusfel

### Disclaimer

This report contains material which is the copyright of certain ORPHEUS Consortium Parties and may not be reproduced or copied without permission.

All ORPHEUS Consortium Parties have agreed to publication of this report, the content of which is licensed under a Creative Commons Attribution-NonCommercial-NoDerivs 3.0 Unported License<sup>1</sup>.

Neither the ORPHEUS Consortium Parties nor the European Commission warrant that the information contained in the Deliverable is capable of use, or that use of the information is free from risk, and accept no liability for loss or damage suffered by any person using the information.

### Copyright notice

© 2015 - 2018 ORPHEUS Consortium Parties

<sup>1</sup> [http://creativecommons.org/licenses/by-nc-nd/3.0/deed.en\\_US](http://creativecommons.org/licenses/by-nc-nd/3.0/deed.en_US)

## Executive Summary

This document aims to identify the main technical and end-user requirements for the reception, presentation and personalised consumption of broadcast object-based content. Its purpose is to serve as a guideline for the design, implementation and assessment of technical solutions that will be developed during the project. Usability specifications are exemplified through various mock-ups and user scenarios. Several hardware and software solutions are considered for the end-user device in order to cover different audio content consumption situations (e.g. domestic use vs mobility). Personalisation and interactivity features are also listed and will require the design and development of user interfaces handling various input devices (e.g. touch screen, GPS sensors, microphone, ...).

The different reception systems shall support MPEG-H delivery format, with an additional PC browser application that shall support AAC encoded audio streams with associated ADM metadata. The transmission will be based on MPEG-DASH in order to offer adaptive distribution according to the network capacity.

In the personalisation stage, the end-user can interact with the content (audio and associated metadata) thanks to different user interfaces (e.g. touch screen, keyboard, sensors). Several interaction/personalisation features are identified and operate either on mixing/spatial aspects (e.g. speech over ambience ratio) or at a semantic level (e.g. language selection, non-linear consumption).

The rendering stage shall support a large variety of standardised or ad-hoc 2D or 3D loudspeaker layouts, as well as binaural rendering over headphones, and shall be able to render the various types of audio streams described in ADM (DirectSpeakers, Matrix, Binaural, HOA, Objects with positional information).

## Table of Contents

<b>Executive Summary</b> .....	<b>3</b>
<b>Table of Contents</b> .....	<b>4</b>
<b>List of Figures</b> .....	<b>5</b>
<b>Abbreviations</b> .....	<b>6</b>
<b>1 Introduction</b> .....	<b>7</b>
<b>2 Reproduction</b> .....	<b>9</b>
2.1 General features .....	9
2.1.1 Hardware receiver .....	9
2.1.2 Web browser .....	10
2.1.3 Mobile device .....	10
2.1.4 Conventional receiver .....	10
2.2 Technical requirements .....	11
2.2.1 Distribution formats .....	12
2.2.2 Audio rendering formats .....	12
2.2.3 Reverberation, distance and room effect control .....	14
2.2.4 Environmental adaptation .....	15
2.2.5 Distribution adaptation .....	15
<b>3 Interaction and Personalisation</b> .....	<b>16</b>
3.1 Mock-ups and scenarios for pilot 1 .....	16
3.1.1 User story and mock-up of magazine programme [C. Baume , M. Armstrong] .....	16
3.1.2 User-story and mock-up of 'Live Music Concert' [W. Bleisteiner] .....	18
3.2 Mock-ups for pilot 2 .....	20
3.2.1 Pilot 2 – Interactive radio documentary [W. Bleisteiner - BR] .....	20
<b>4 User Interface</b> .....	<b>22</b>
4.1 General Features .....	22
4.1.1 Hardware receiver .....	22
4.1.2 Web browser .....	22
4.1.3 Mobile device .....	23
4.2 End-user requirements .....	23
4.2.1 Programme control .....	23
4.2.2 Audio quality .....	23
4.2.3 Gesture control .....	24
4.2.4 Situational information .....	24
4.2.5 Feedback / interaction .....	24
<b>5 Conclusions</b> .....	<b>25</b>

## List of Figures

Figure 1: Reception stage in the global architecture .....	7
Figure 2: Logical stages of the Reception and Presentation macro-block. ....	11
Figure 3: Pilot1 - User-story 'Journey to work on a personal device' [from C. Baume, M. Armstrong - BBC] .....	17
Figure 4: Pilot1 - Mock-up of 'Live Music Concert' [from W. Bleisteiner - BR] .....	19
Figure 5: Pilot2 – Additional picture/video documents [from W. Bleisteiner - BR] .....	21

## Abbreviations

<b>AAC</b>	Advanced Audio Coding
<b>ACN</b>	Ambisonics Channel Number (related to HOA)
<b>ADM</b>	Audio definition model
<b>AES69</b>	Standard for Spatially Oriented Format for Acoustics (used for HRTF exchange format)
<b>BWF</b>	Broadcast wave format
<b>DAB</b>	Digital Audio Broadcast
<b>DASH</b>	Dynamic Adaptive Streaming over HTTP
<b>DVB</b>	Digital Video Broadcast
<b>FDN</b>	Feedback Delay Network
<b>GPS</b>	Global Positioning System
<b>HOA</b>	Higher Order Ambisonics
<b>HRTF</b>	Head Related Transfer Function
<b>ITD</b>	Interaural Time Delay
<b>ITU-R</b>	International Telecommunication Union, Radiocommunication Sector
<b>MPEG</b>	Moving Picture Experts Group
<b>N3D</b>	Normalised 3D (related to HOA)
<b>OB</b>	Outside broadcasting
<b>PCM</b>	Pulse-code modulation
<b>SID</b>	Single Index Designation (related to HOA)
<b>SN3D</b>	Semi-normalised 3D (related to HOA)

# 1 Introduction

This document is dedicated to the specification, development and assessment of tools and algorithms for object-based audio rendering on multiple platforms and end-user reproduction systems, including mobile devices. The objective of WP5 is to exemplify, through mock-up scenarios, how the listening experience can be broadened and diversified, taking advantage of the modularity and interactivity offered by object-based audio format. Subtask T5.1 will mainly address the design and evaluation of different spatial rendering algorithms according to the reproduction setup. Subtask T5.2 will develop exemplar implementations of object-based user experiences that cover different degrees of personalisation according to the user or listening situations. Subtask T5.3 will address the specifications, design, implementation and assessment of user interfaces dedicated to customised or interactive audio content consumption. Finally, subtask T5.4 will specify and develop specific schemes for the evaluation of user experience in the context of object-based audio, addressing both the perceived quality of the audio itself and the characterisation of the overall user experience.

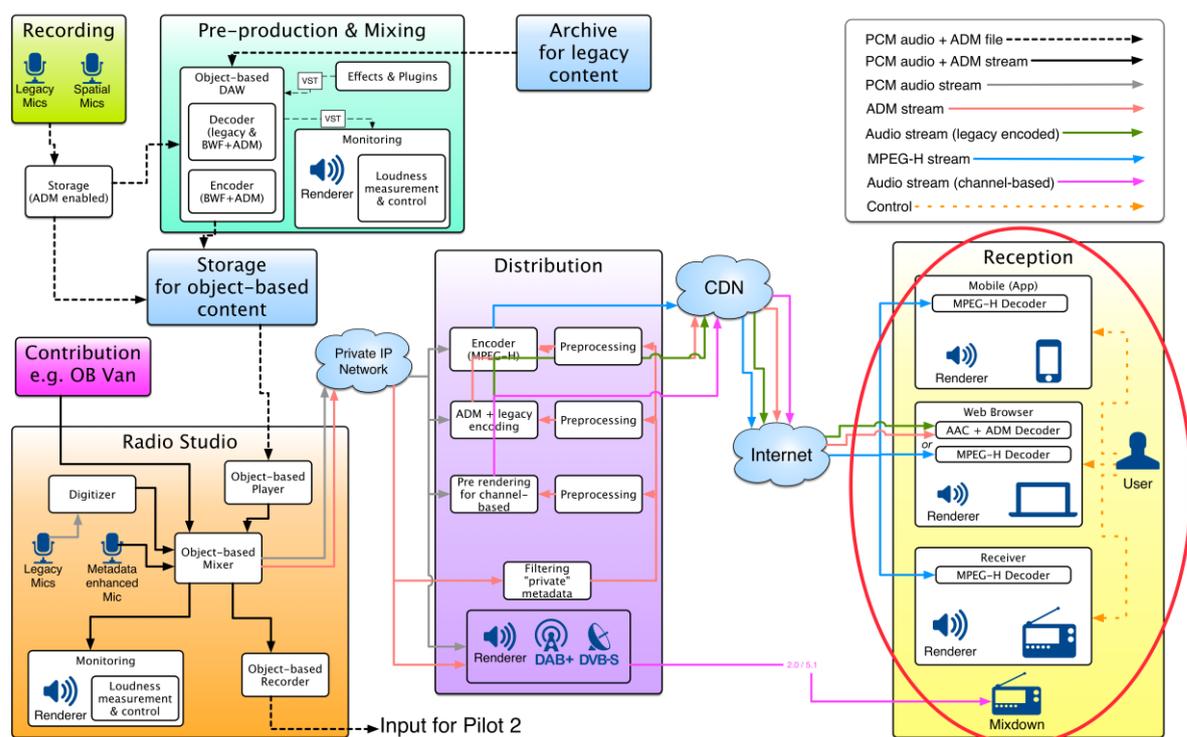


Figure 1: Reception stage in the global architecture

According to the global architecture of the audio-object broadcast workflow (see Figure 1.) these tasks concentrate on the end-user side and are symbolised by the macro-block “Reception”. The main role of this macro-block is to perform the decoding of the audio streams and the rendering of the different audio-objects according to associated meta-data, target platform and possible user interaction. Several hardware and software solutions are considered in order to cover different segments of consumer markets, as well as different listening habits and audio content consumption situations. These solutions differ in terms of rendering capabilities, according to the available network bandwidth, number of audio I/O channels and processing power. They also differ in terms of user interface and proposed personalisation/interactivity features since they are not addressing the same listening situations (e.g. domestic use vs mobility) and are equipped with different sensors and input devices (screens with keyboard, built-in microphones, GPS sensors etc).

The ORPHEUS project opts for an integrated methodology with real-world pilots central to our R&D approach. These pilots will not only serve as demonstrators but will also help to define the requirements for an end-to-end chain that utilises innovative features of object-based media and

takes into account the constraints of a real-world, day-to-day production workflow. Throughout the present document, references will sometimes be made to the pilots #1 and #2 that are currently being defined in the work-package WP2. In summary, Pilot #1 is focusing on live multiplatform object-based audio while Pilot #2 will show enhanced object-based audio for on-demand consumption and advanced 3D rendering.

The document is organised as follows. Section 2 is dedicated to the general features and technical requirements related to the hardware devices or software applications in charge of the reception and rendering of the broadcast audio content. Section 3 presents several mock-ups and user scenarios that illustrate the consumption of representative object-based audio programmes, received and listened to in various situations and involving diverse user interaction. In light of these examples, Section 4 describes a collection of controls and interactivity features that shall be provided to the user.

## 2 Reproduction

In the context of conventional broadcast workflow, many standards, such as Recommendations ITU-R BS.1116, ITU-R BS.775 and EBU Tech. Doc. 3276, have been proposed for studio construction, room response, speaker placement, and downward conversion of multichannel content. However, these recommendations have been specified purposely for channel-based audio content. Today, the need to support 3D audio content and to offer high versatility between rendering setups, from headphones to fixed installations with ever-growing number of loudspeakers, motivates the development of a new broadcast workflow relying on an object-based audio approach.

Consequently, a new generation of hardware and software solutions have to be designed and implemented for the reception and rendering of the object-based audio content on the end-user side. The decoding of the audio and metadata streams and the rendering modules shall automatically adapt the play-out of the object-based content according to the end-user setup, i.e. from conventional loudspeaker setups to advanced multi-channel immersive audio systems or binaural rendering over headphones. Such hardware or software receivers shall provide spatial rendering algorithms according to the reproduction loudspeaker setup, following standardised or ad-hoc 2D or 3D arrangements. Innovative rendering algorithms shall be designed and compared in order to evaluate aspects such as colouration, objects' 3D position, spatial extent or movements, and reverberation. Scalable rendering and web-based rendering shall also be developed. Binaural rendering over headphones shall receive specific attention since binaural listening could become a major vector for 3D audio access with enhanced possibilities of personalisation and interaction. The rendering shall also be adapted to reproduction system impairments, correcting for speaker deficiencies or misplacement, adapting to room response, possibly measured with a 3D acoustic probe.

### 2.1 General features

The hardware or software solutions will provide means for decoding and rendering audio streams received via content delivery network. Two main delivery formats are considered: MPEG-H streams, and ADM metadata with legacy encoding of audio objects (AAC).

According to the available bandwidth of the network and processing power of the end-user device the receiver shall select/ask for an appropriate streaming format, i.e. where some audio objects are kept independent while others are pre-combined or pre-rendered into a monophonic or multi-channel "proxy object". The streams can also be delivered at various bitrates, depending on the end-user's situation. Additionally, streams shall be selected based on user preference (such as language, available listening time, etc.).

The play-out module shall support a large diversity of standardised 2D or 3D (2.0, 5.1, 9.1, 22.2, ...) or personalised loudspeaker setups as well as binaural playback over headphones with real-time head movement compensation.

*N.B. The ITU-R baseline renderer should be used for the production part of the reference architecture. However it will not be available before mid-2017. Meanwhile, it was decided to use the MPEG-H encoder-renderer solution of FHG, especially for the development of the pilot #1. Next steps will consider the ITU renderer.*

The following hardware and software solutions will be considered in order to cover different segments of consumer markets, as well as different listening habits and audio content consumption situations.

#### 2.1.1 Hardware receiver

The hardware receiver is a standalone processor. It fulfils all the requirements by itself, and does not

need any third party application. The device is CPU-based, and the audio processing is a closed software library running on an embedded computer. Some third-party libraries such as audio decoders may be included as independent modules, linked to the main library and run from the main signal processing function. The object-based audio features provided in ORPHEUS will be added into the signal processing flow. Two solutions will be considered:

- embedding FHG's MPEG-H renderer: the C++ library provided by FHG will be included in the processing software of the device;
- implementing a custom ADM + legacy renderer;

### 2.1.2 Web browser

The web browser is considered the most universally available reception system. Its ubiquity means that a very large audience can be exposed to object-based broadcasting without any required extra effort (such as downloading plug-ins, installing apps, configuring speaker setups etc.). At present, web browsers do not readily support object-based audio rendering. Three solutions are being considered in ORPHEUS:

- embedding an MPEG-H decoder and its included renderer in the web browser. In the pilot phase this will only be feasible by using a modified version of the Chromium browser;
- implementing a rendering system based on JavaScript and the Web Audio API. While a native solution will not be available for all browsers immediately, growing support for, and maturity of, the Web Audio API makes this an attractive option;
- creating a plug-in capable of rendering object-based audio (for instance using MPEG-H). As browsers support for third party plug-ins is declining, this option was rejected as it may not be available for future use.

Both the modified Chromium browser and the Web Audio based solution will be implemented for the ORPHEUS pilots.

### 2.1.3 Mobile device

On mobile devices, it is common to install custom apps. In a native app, advanced and efficient renderers can be included, which offer all or a specific subset of the object-based audio features provided in ORPHEUS. Two possible implementations of a decoding/rendering system in a native mobile app are:

- FHG's MPEG-H decoder with its included renderer. Provided as a C++ library, this renderer, possibly optimised specifically for mobile hardware, will be included in a native app for the ORPHEUS pilots.
- a custom renderer based on ADM + legacy encoded audio files.

### 2.1.4 Conventional receiver

For backward compatibility, conventional audio receivers can play-out pre-rendered channel-based versions, (2.0 or 5.1 format) generated by broadcasters and delivered via DAB or DVB-S/C or hosted on their websites/media libraries. Note: These versions offer very few features and advantages of the original object-based audio content.

## 2.2 Technical requirements

The logical stages of the Reception and Presentation macro-block are presented in figure 2. Compressed audio-streams and associated metadata are first decoded. User interaction features are provided by altering metadata in the personalisation stage. The rendering stage receives the audio streams and metadata and automatically adapts to the playback setup (headphones, 2D or 3D loudspeakers setup). Optionally, an environmental adaptation is proposed with inputs from environment sensors (head-tracking for binaural rendering, compensation for background noise, rendering setup alignment/equalisation).

For Pilot #1, the input stream of the reception part is MPEG-H over MPEG-DASH, except for one PC web browser, which will receive AAC over MPEG-DASH plus a stream of ADM metadata.

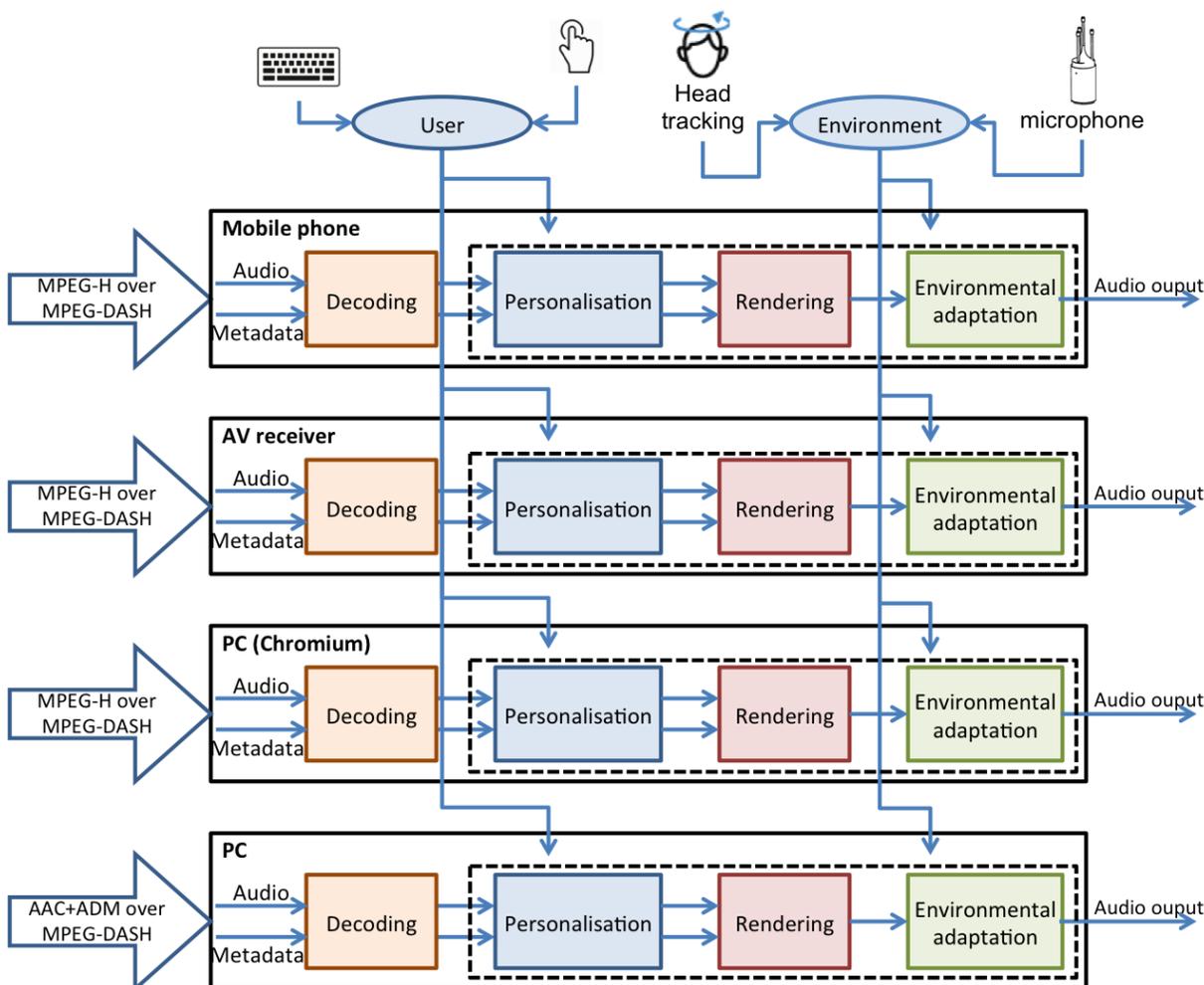


Figure 2: Logical stages of the Reception and Presentation macro-block.

From a logical standpoint, all of the different classes of receivers (mobile phone, AV receiver, Web browser), shall start with a decoding module of the compressed audio streams and associated metadata. A personalisation/interaction module shall allow for adapting the content stream (audio and metadata) according to the user's preference. Then a rendering module shall process the audio streams to feed the audio presentation setup (loudspeakers or headphones). Finally, the adaptation to the environment can be carried out, with inputs from environmental sensors.

The transmission will be based on MPEG-DASH. DASH (Dynamic Adaptive Streaming over HTTP) is an adaptive bitrate streaming technique that enables high quality streaming of media content over the

Internet delivered from conventional HTTP web servers. MPEG-DASH works by breaking the content into a sequence of small HTTP-based file segments, which contain a short interval of playback time of content. The content is made available at different bitrates, and alternative short segments encoded at these different bitrates, covering aligned intervals, are made available. The client selects the next segment with a bitrate corresponding to the current network conditions. Thus, an MPEG-DASH client can seamlessly adapt to changing network conditions.

In the personalisation stage, the user can interact with the content (audio and associated metadata). Objects can be altered or switched on and off (e.g. language selection, speech over ambience ratio, non-linear consumption, ...). This personalisation module will handle inputs from various user interfaces (touch screen, keyboard, sensors, gesture control, etc.).

The rendering will interpret the metadata transmitted with the audio (e.g. direction information) to manipulate the streams accordingly. It may also exploit information coming from integrated or external sensors such as a head-tracking.

The adaptation to the environment ensures the fidelity of the playback. It applies processing to the audio to compensate for the environment in which the audio is heard. The loudspeakers are time-aligned, level-aligned, equalised, and phase-corrected. Thanks to individual filters, the group delay of each loudspeaker is compensated, and it is ensured that all of them will sound as similar as possible, to facilitate the reproduction of objects located in between them, and to increase the fluidity of the motion of moving objects. Real-time adaptation is also provided in order to apply dynamic compression and loudness control according to the environment noise level measured by the integrated microphone.

Note that for the case of MPEG-H, the above-mentioned functionality (Decoding, Personalisation, Rendering, Environmental Adaptation) is tightly integrated into the MPEG-H Decoder.

### 2.2.1 Distribution formats

The general requirements for representation, archiving and provision of object-based audio content are described in the deliverable D4.1. Here follows a summary of the main requirements and recommendations of the formats that will be used in the project.

- **MPEG-H:** The MPEG-H Decoder implements the Low Complexity Profile Level 3. In its current implementation it is limited to a maximum of 16 simultaneously active (=rendered) signals out of a total of up to 32 channel, objects, and/or HOA signals in the bitstream as specified in the standard.
- **ADM:** The Audio Definition Model (ADM) is an EBU specification of metadata that can be used to describe object-based audio, scene-based audio and channel-based audio. It can be included in BWF WAVE files or used as a streaming format in production environments. The ADM is specified by Recommendation ITU-R BS.2076. Work on specifying a way of rendering audio that uses ADM is currently being done to answer Question ITU-R 139/6. This should lead to publication of a Recommendation for a “baseline renderer”. Orpheus developments shall conform to these recommendations along their progressive definition and publication.

### 2.2.2 Audio rendering formats

ADM provides conventions to describe the audio stream content in order to allow for correct decoding and rendering. Several types (typeDefinition attribute) of content may be delivered:

- *DirectSpeakers*, for channel based audio with each track feeding a speaker directly,
- *Matrix*, for channel based audio where channels are matrixed together,

- *Objects*, for object-based audio, where each channel represents an audio object with associated positional information,
- *HOA*, for scene-based content using High Order Ambisonics conventions,
- *Binaural*, for binaural content dedicated to playback over headphones.

#### 2.2.2.1 DirectSpeakers

For the DirectSpeakers type, sub-attributes will provide the labels of the different tracks (e.g. FrontLeft, ...). Optionally, the bounding of the loudspeaker position may be described. When the standardised loudspeaker setup is available, the renderer simply feeds the loudspeakers with the corresponding tracks.

#### 2.2.2.2 Matrix

For the Matrix type, the renderer shall apply the gain factor and phase shift defined in the sub-attributes associated to each channel.

#### 2.2.2.3 Objects

The Object type is the most flexible case and allows for dynamic control. Sub-attributes describe each object in position and spread in azimuth and width, elevation and height, and distance and depth. Yet, no specific rendering algorithms are specified, especially regarding the rendering of distance. However, some flags exist, for instance in order to force an object to be locked to the nearest loudspeaker, or to jump between successive positions instead of being interpolated. Other parameters include description of the diffuseness, the divergence, exclusion zones, the movement of the objects, and their relation to the screen position and geometry. Dialogue related settings are also specified. The renderer shall support:

- Point-like mono sources
- Point-like mono sources with an adjustable degree of diffuseness
- Extended mono sources with an adjustable extent in the horizontal and vertical dimensions
- Extended mono sources with an adjustable degree of diffuseness

#### 2.2.2.4 HOA

Several conventions are used for transmitting HOA signals (N3D, SN3D, FuMa, ...). They are not compatible as they differ in terms of channel ordering and gain normalisation.

Historically, the earliest channel order convention is referred to as SID (Single Index Designation) and is designed to extend the historical B-format (corresponding to 1st order ambisonics) to the higher orders. In the SID convention HOA channels, here denoted by (order, degree), are ordered as follows: (0,0), (1,1), (1,-1), (1,0), (2,2), (2,-2), (2,1), (2,-1), (2,0), (3,3), (3,-3)... In the most recent channel ordering convention, referred to as ACN (Ambisonic Channel Number), channels are ordered by increasing order and degree, as follows: (0,0), (1,-1), (1,0), (1,1), (2,-2), (2,-1), (2,0), (2,1), (2,2), (3,-3), (3,-2)... ACN is rapidly becoming the most widely used channel ordering convention.

HOA signal gain normalisation is driven by the different ways to normalise spherical harmonic functions. Spherical harmonics can be normalised in energy or in amplitude, either in 2D or 3D. The energy normalisation in 3D is denoted N3D (normalised 3D). This normalisation ensures that the signals have the same RMS level in the case where the sound scene is perfectly diffuse. On the other hand, the amplitude normalisation in 3D, denoted SN3D (semi-normalised 3D), ensures that the crest level remains in the range [-1;1] when encoding a plane wave of amplitude lower than 1.

In the current state of ADM recommendations (EBU Tech 3364, ITU-E BS2076-0), it is proposed to provide 'suitable names' to specify which convention is used. It is also recommended that each HOA component be described by its degree and order and an equation (using C-style notation) to avoid confusion. For the ORPHEUS project, the question is still open whether we shall use the ACN ordering with SN3D normalisation (for the above described reasons) or if we use the N3D convention as in MPEG-H in order to be compatible. In any case, conversion can be applied between both conventions.

### 2.2.2.5 Binaural

In the case of the Binaural type, the audio content simply consists of two channels that should be fed directly to the headphones. However, given the increasing usage of headphones, the rendering stage shall devote specific attention to the binaural rendering over headphones of Objects type content, DirectSpeakers content (through the Virtual Speakers paradigm) or HOA content (through Virtual Speakers decoding or direct HOA to binaural conversion).

- FIR or SOS implementation: Several binaural rendering implementations can be used. The simplest implementation is the convolution with HRIRs (Head related impulse response). Other implementations are also possible such as HRIR decomposition into a minimum-phase component and an associated inter-aural time delay. The minimum-phase component may be implemented directly in the time domain (FIR), or in the Fourier domain. Significant processing optimisation can be achieved after modelling the minimum-phase component as a combination of bi-quad filters (SOS for Second Order Sections). The inter-aural delay is implemented as a fractional delay.
- Selection from local set of HRTFs: The receivers shall provide a limited set of HRTFs, locally stored, so that the end-user can choose the filters that match his/her HRTFs best.
- Selection/Download from public database: The receiver may also provide access to a HRTF public database to allow for selecting/downloading/storing various set of HRTFs adapted to the users. HRTFs are stored using the AES69 format.
- Test signals: Test signals shall be provided to assist the end-user in choosing the HRTF filters that best fit his or her ears.
- Parametric adaptation (e.g. ITD): other HRTFs individualisation procedures may propose an individual fine tuning of the Interaural time delay (ITD) associated with a non-individual minimum-phase HRIR set selected from a database. Another possible tuning procedure may let the user rotate non-individual HRTF filters around the inter-aural axis.

### 2.2.3 Reverberation, distance and room effect control

ADM format introduces specific attributes to describe the distance and depth of an audio object. However, no recommendation is given with regard to how these attributes should be rendered. It is well known that an efficient distance rendering requires the control of the energy ratio between direct sound, first reflections and diffuse reverberated field. First reflections also play a major role in controlling the apparent source width, another object attribute considered in the ADM format.

Although not explicitly mentioned, the ADM format provides a general framework that allows for such room effect components to be conveyed in parallel within an AudioPack (a pack of channels that relate together). Having these components produced and transmitted separately (direct sound channels, first reflection channels for each source, plus one or several channels for the late diffuse reverberation) shall guarantee a better independence of the rendering with regard to the different targeted playback setups (headphones with binaural rendering, conventional or ad hoc 2D or 3D loudspeaker setups).

Advanced interaction/personalisation scenarios (Pilot #2) may also take advantage of transmitting separately the room effect components. It would provide the end-user with efficient control on high-

level controls, such as source distance, source orientation or the modification of the listener's standpoint in the scene (e.g. radio drama).

Further investigation is needed in order to decide whether the rendering stage should implement a FDN/convolution based reverberation.

## **2.2.4 Environmental adaptation**

### **2.2.4.1 Head Tracking**

Head tracking has been shown to provide improved 3D sensations and source externalisation. Two scenarios may be envisioned. In the first one, the rendering device is equipped with its own head-tracking module. The binaural rendering shall then compensate for head movements at the rendering stage. In the second situation, recently commercialised headphones equipped with integrated head-tracking are used. They will typically support conventional channel-based format (e.g. 5.1) or HOA scene-based formats. In that case, the renderer should simply provide the headphones with a pre-rendered channel-based (as if addressing a loudspeaker setup) or HOA-encoded scene (as if addressing an external HOA decoder).

### **2.2.4.2 Background noise level sensing**

Loudness compensation is nowadays a common feature provided by audio systems, especially in automotive industry. Such a feature can also be integrated in a smartphone using its microphone(s) to measure the level of background noise in the listening environment (e.g. public transportation). The reproduced level and dynamic range are then adapted according to the noise level. Within the specific context of object-based audio broadcast, the background noise compensation process shall be refined and apply different loudness, filtering or dynamic compression rules according to the audio content conveyed by each object (dialogue, ambient channels, etc.). It is planned to incorporate this feature in Orpheus.

## **2.2.5 Distribution adaptation**

In order to allow for the audio content to be adapted to certain circumstances such as the available network capacity or the end-device capabilities, the distribution should be able to deliver adapted scenes. For both adaptations, the end-device must be capable of sending necessary information to the aforementioned distribution pre-processing component.

### **2.2.5.1 Adaptation to the available network capacity**

One aspect of adaption should be the network capacity. Depending on the complexity of the audio scene or the number of resulting audio objects, the transmission bitrate could be too high for the available distribution channel. In order to meet the available bandwidth, the distribution pre-processing component should provide different versions of the content. This could be done either with reduced encoding qualities (e.g. spending only 48 kbit/s per object instead of 64 kbit/s) transmitted via MPEG-DASH, or with a pre-rendering of the audio scene where the total number of audio objects is reduced. A combination of both options should be also considered.

### **2.2.5.2 Adaptation to the end-device capabilities**

Another aspect of adaption should be the end-device capabilities. Depending on the complexity of the audio scene, the end-device processor might not be able to render all audio objects in real-time. Hence, the distribution pre-processing component should provide different versions of the content with fewer objects. This pre-rendering of the audio scene should take into account semantic information of the objects such as "importance" as well as the actual end-device capabilities.

## 3 Interaction and Personalisation

This section describes different application mock-ups and object-based user experiences. Their intent is to harness the responsive, client-side rendering of the audio objects in order to deliver an experience that has various degrees of personalisation for different users and listening conditions. The particular exemplars are chosen in order to cover a range of levels of curation - ranging from simple provision of alternate audio tracks to interactively-rendered multi-object audio, as well as a number of application domains including: accessibility, headphone/near field rendering and variable duration. These scenarios help to identify the technical requirements for personalisation and presentation to the end-user, options to degrade gracefully in case of devices or delivery constraints, allow the option for different rendering methods (e.g. use of rendering over headphones or loudspeakers) and practical implementation of user interaction and accessibility features (e.g. Dialogue Enhancement). The different mock-ups and user scenarios refer to the different pilots that will be implemented during the Orpheus project. Pilot #1 is focusing on live multiplatform object-based audio while Pilot #2 will demonstrate enhanced object-based audio for on-demand consumption and advanced 3D rendering.

### 3.1 Mock-ups and scenarios for pilot 1

#### 3.1.1 User story and mock-up of magazine programme [C. Baume , M. Armstrong]

About to set out for work, I start listening to an object-based radio programme on my mobile device. I joined after the start of the programme, but I see the topic of discussion and catch up with what has been said by reading the transcript which tells me who has said what so far.

As I leave home, I put my headphones on and walk out onto busy streets with variable noise levels. My device detects high noise levels along with my movement and the fact that I am now paying for mobile bandwidth. The device selects a stereo server rendered version of the programme to reduce bandwidth costs. Also I adjust the foreground/background mix to help me hear the speech over the background music and engage compression and auto level so as to keep the speech level above the busy street noise. Because I am walking it is preferable to listen in stereo so that I can distinguish between the external sounds and the radio programme.

As the mobile signal reduces in quality the device opts for a lower bandwidth mono version to improve reliability.

As I reach the railway station the device picks up a wifi hot spot, and switches to streaming the programme in an object based form.

Whilst I am sitting in the waiting room the device detects that I am sitting still, that the room is quiet and because my train is 10 minutes late this morning, I will have 15 minutes to enjoy this experience, so I select a fully immersive audio experience with a wide dynamic range and enable the head tracking. The item includes a new song which I really enjoy, so I hit the "thumbs up" button to let the presenter know how much I enjoyed it and to encourage the station to play it again in the future. After the song ends, the presenter says it was the most liked song this week.

Just before the train arrives I pause the programme at the end of the current section. Once I am seated on the train the device switches back to the mobile signal, but because the environmental noise is not too high and the signal level is good it opts for a pre-rendered binaural experience with normal foreground/background mix and resumes with the introduction to the next item.

The pre-rendered version is better adapted to the network bandwidth capacity in public transports. Although, pre-rendered binaural does not allow for head tracking compensation, it is not a limitation in such circumstances since, anyway, I don't want the soundscape to revolve as the train goes round a curve.

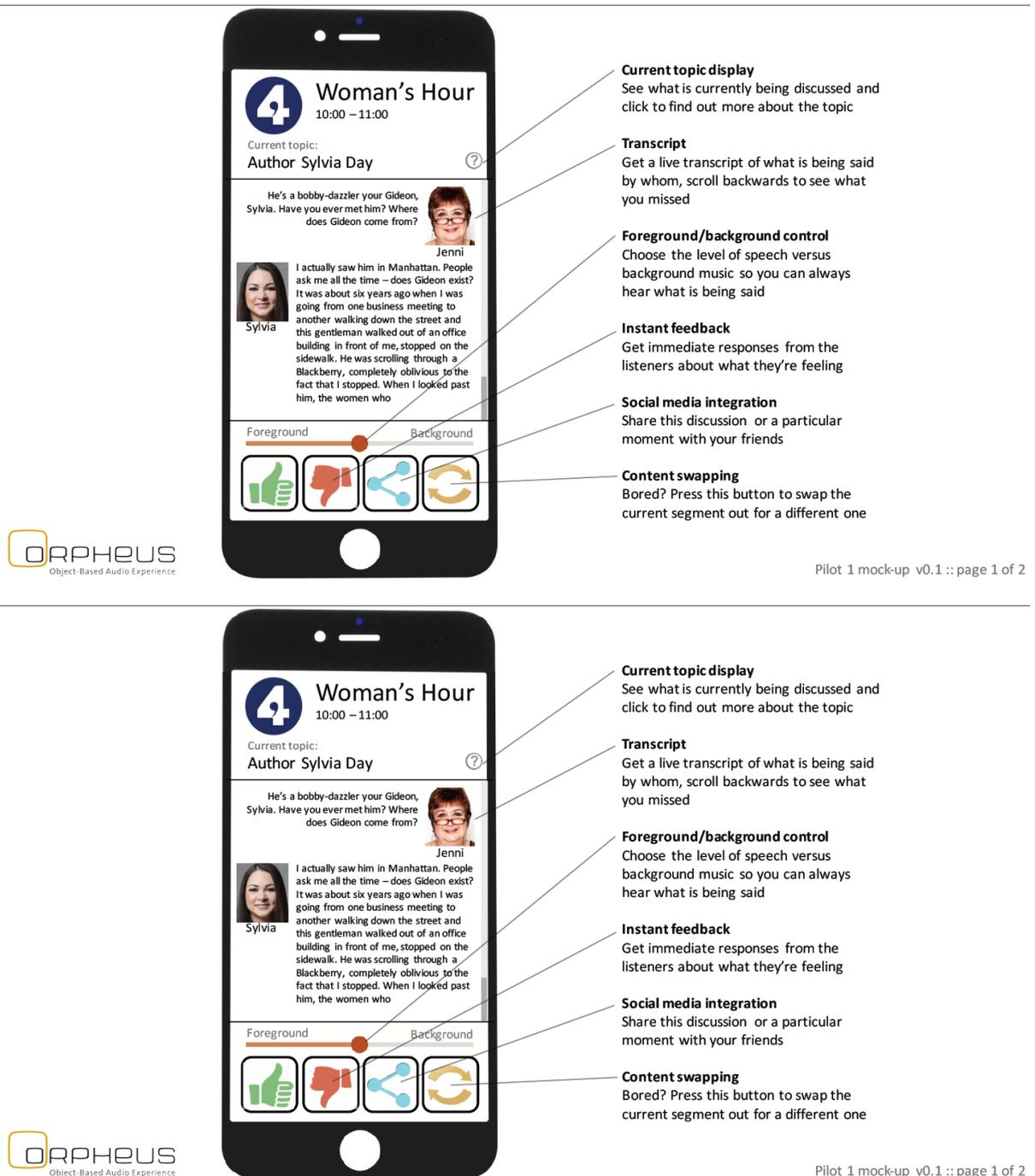


Figure 3: Pilot1 - User-story 'Journey to work on a personal device' [from C. Baume, M. Armstrong - BBC]

The programme moves on to an item on a topic I'm not at all interested in, so I hit the swap button and I get to listen to an alternative item which is much more interesting, so I request more information on the topic and the device brings this up on the display in text and picture form. The device also offers other related content for me to listen to later, which I accept.

At the end of this alternative item the device rejoins the live programme as they start a new item. This item is about the environment and involves interviews with people in 3 different languages, English, French and Spanish. The device knows that I can understand English and can get by in French, but not Spanish, so it switches to partial object based version with the English translation for the interview in Spanish, but gives me the interview in French in the original language.

As the train pulls into the station my device also alerts me that my bus is about to depart so I pause the programme till I am safely on the bus. Now the bus is far noisier than the train so the device selects a compressed stereo stream and resumes at the beginning of the item that had just started. I

still find it difficult to follow because of the noise so I move the foreground/background slider to increase the volume of the speech relative to the background music. The item is really interesting, and will interest a number of my friends. I hit the share button, which posts the item on my social network as audio and transcript, along with a marker to indicate at what point I shared it.

As the device detects that I am arriving at work it continues to the end of the current item and then switches to my favourite music network, storing up any other items of interest in the remainder of the current programme. It also switches to WiFi and requests the fully object based version of the programme.

As I am now in the office and working at my desk, I prioritise the music over the speech to enable me to concentrate on my work and the device gives me information about each music item so I can glance over at it if I am interested and an immersive musical soundscape to help me relax.

### 3.1.2 User-story and mock-up of 'Live Music Concert' [W. Bleisteiner]

Peter had to work late today. He couldn't get out of the office in time to be at home for tonight's live radio transmission of Gustav Mahler's 5th Symphony with the Bavarian Radio Symphony Orchestra, conducted by Mariss Jansons. But now on the commuter train, with his smartphone app and headphones on, he chooses the binaural rendered version, that makes him feel just as being there in Munich's Philharmonie at Gasteig.

As the train takes some bends and Peter loves to look out of the window to see the beautiful summer sunset, he has switched off the Head Tracking option, but he certainly needs the Loudness Adaption, so that some of the ppp (triple pianissimo) passages, levelled below -30dB dynamically remain audible in a noisy train compartment.

Along with the music, Peter is able to read some background articles on the composer Gustav Mahler and the music piece, just like in the printed concert programme for the audience in the hall. Some extra audio clips are also available at any time he wants: an interview with the conductor and also the corresponding episode from the series "Masterpieces" on Mahler's 5th produced already some years ago for BR-KLASSIK and now straight available from their archive. If the editorially curated content still doesn't quench his knowledge thirst, he can start searching with semantic web tools like DPedia, as there are tags provided in the context information.

After 20 minutes Peter gets off the train at a station and picks up his car for some 7 kilometres to his house. The car is fairly new and has a Carplay built-in, so when he connects his smartphone with it, the app switches automatically to the stereo rendering, optimised for the car's loudspeakers.

In order not to miss anything Peter pauses the programme at the end of the first movement of the symphony and resumes it when he's in the car for the 2nd movement.

Arriving at home, Peter rushes into his living room, where he has a decent AV-system installed with a 5.1 loudspeaker setup. As the smartphone is already running low on battery, he puts it into the docking station, to recharge it. This docking station now connects it directly via HDMI with the audio system and switches to 5.1 surround sound rendering. Also the Loudness Adaption is reduced here to a minimum, offering full range dynamic.

But Peter already has plans to improve his home set-up. Pretty soon he will buy a new AV-Receiver which supports 'object-based broadcasting' out of the box. The one he has chosen is not only scalable in loudspeaker outputs from stereo to 5.1 and up to 22.2, it also includes room-optimisation, reproducing the best sound according to the acoustic parameters of his living room. So he can experiment extensively, what loudspeaker set-up serves his needs (and budget) best. The receiver also uses his smartphone or tablet as user interface and second screen to control various technical functions and delivering all information straight to his fingertips. But that's yet another user story...

Live music concert



**3DAudio Logo**

An original surround sound program is on air

**Currently playing display**

Basic information from program planning/EPG system. ("Now playing" in live situations mostly unsolved yet)

**More about the music**

Background articles and audio clips or video clips with artists are usually preproduced and available

**Foreground/background control**

Choose the level of speech versus background music so you can always hear what is being said  
- greyed or disappeared when not available in music

**Instant feedback**

Get immediate responses from the listeners about what they're feeling

**Social media integration**

Share this discussion or a particular moment with your friends

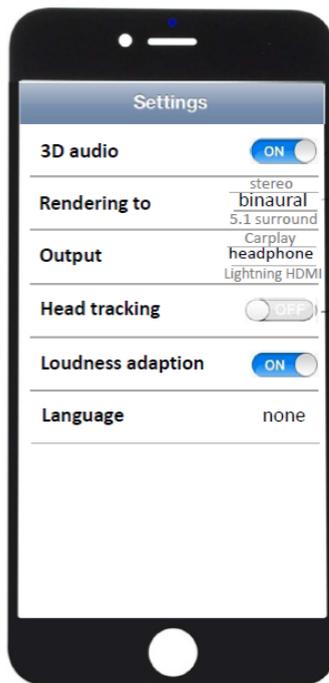
**Content swapping**

Bored? Press this button to swap the current segment out for a different one - for live program a dedicated alternative playlist with material has to be made available

Pilot 1 mock-up v0.2 p.



Live music concert



**Three-dimensional audio**

Get a fully immersive experience using the latest spatial audio rendering

**Rendering**

Various formats might be available- making the smartphone a universal receiver for various locations: at home, on the go, in the car....

**Output**

...therefore various outputs are assignable

**Head tracking**

If you're using a compatible headset, head tracking means the sound stays where it should

**Loudness adaption (dynamic compression)**

Using your device's microphone, the sound is automatically adjusted to your environment's noise level - in classical music low levels are accordingly increased

**Multiple languages**

When different languages are available, such as in some interviews, you will hear your preferred language - if not applicable "none", grey out or not available at all



Pilot 1 mock-up v0.2 page 4 of 4

Figure 4: Pilot1 - Mock-up of 'Live Music Concert' [from W. Bleisteiner - BR]

## 3.2 Mock-ups for pilot 2

### 3.2.1 Pilot 2 – Interactive radio documentary [W. Bleisteiner - BR]

interactive radio documentary



**3DAudio Logo**

An original surround sound program is on air  
**interactive logo**  
shows that this is not just a linear program but offers extra features

**Currently playing display**

Basic information from program planning/EPG system. ("Now playing" in live situations mostly unsolved yet)

**More about the item**

Background articles and audio clips or video clips with artists are usually pre-produced and available

**Foreground/background control**

Choose the level of speech versus background music so you can always hear what is being said

**Play interactive version**

If the content offers specific interactive features additional to a linear play-out it can be activated

**Instant feedback**

Get immediate responses from the listeners about what they're feeling

**Social media integration**

Share this discussion or a particular moment with your friends

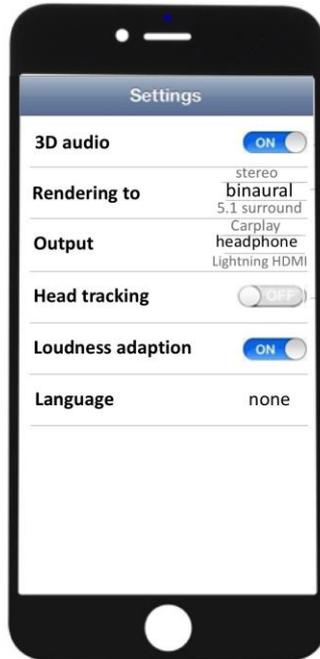
**Content swapping**

Bored? Press this button to swap the current segment out for a different one – for live program a dedicated alternative playlist with material has to be made available

Pilot 2 mock-up v0.3 page 1 of



interactive radio documentary



**Three-dimensional audio**

Get a fully immersive experience using the latest spatial audio rendering

**Rendering**

Various formats might be available- making the smartphone a universal receiver for various locations: at home, on the go, in the car.....

**Output**

...therefore various outputs are assignable

**Head tracking**

If you're using a compatible headset, head tracking means the sound stays where it should

**Loudness adaption (dynamic compression)**

Using your device's microphone, the sound is automatically adjusted to your environment's noise level - in classical music low levels are accordingly increased

**Multiple languages**

When different languages are available, such as in some interviews, you will hear your preferred language – if not applicable "none", grey out or not available at all

Pilot 1 mock-up v0.2 page 4 of 4



interactive radio  
documentary



**video and picture player with 360° function**  
navigation possible with motion sensors or  
finger wiping

**panorama mode**

**touch point**  
identifies audio objects and makes them  
selectable for amplified or solo listening.

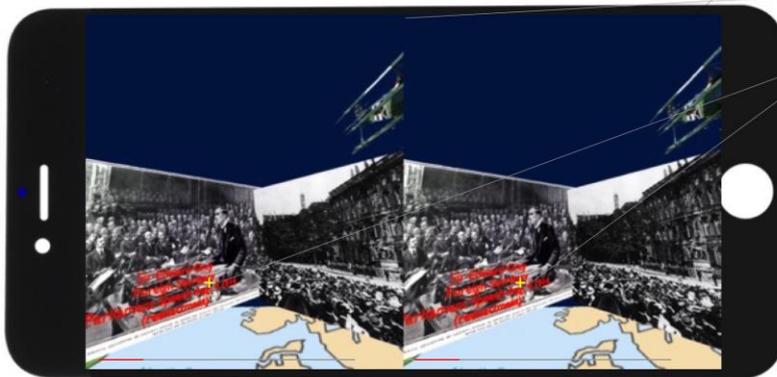
**subtitles to identify speaker/soundbites**

**live transcription from audio stream**  
possibly from selected audio objects only



Pilot 2 mock-up v0.3 page 4 of 4

interactive radio  
documentary



**video and picture player with 360° function**  
navigation possible with motion sensors or  
finger wiping

**stereoscopic mode for VR glasses**

**focus point**  
identifies audio objects and makes them  
selectable for amplified or solo listening.



Pilot 2 mock-up v0.3 page 4 of 4

Figure 5: Pilot2 – Additional picture/video documents [from W. Bleisteiner - BR]

## 4 User Interface

The object-based broadcasting system developed in Orpheus provides a wealth of new features, not only to radio producers, but to the listener (end-user) as well. While some of these features may be relatively straightforward and easy to understand, others are totally new and can be quite complicated to control. Additionally, the relevance of these new features may depend both on the content of the programme (e.g. the number of speakers in an interview, format of the music that is being played) and on the listening conditions of the consumer: a high-end home reproduction system may provide different options from a mobile app used during commuting.

The proposed user interfaces should provide understandable and engaging access to relevant features in a wide range of scenarios. In T5.3, these scenarios will be researched, developed and implemented for a range of platforms, broadly categorised as Hardware Receiver, Web Client and Mobile App.

### 4.1 General Features

The user interface will provide means for exploiting both mixing/spatial related metadata as well as semantic metadata. For instance, the user interface shall provide means for adapting the audio quality to the rendering setup or to the listening conditions (e.g. propose 3D fully-immersive audio or more conventional playback, propose automatic level compression according to the estimation of the background noise). On a semantic level, the user interface will provide means for selecting the preferred language, for displaying the names of radio program participants, text transcription, etc.

Additionally, the user interface will provide support for object-based broadcasting's non-linear capabilities, such as variable-length playback, selection of alternate or extended content and programme navigation.

#### 4.1.1 Hardware receiver

The user interface of the receiver allows the user to control the parameters of all the embedded features. Two interfaces are proposed:

- The main graphical interface runs as a set of dynamic web pages that can be opened in an embedded or a distant browser. This allows a user-friendly experience as well as both local and remote control. The object-based features will be addressed on a dedicated page, as well as the parameters of each of the decoders/renderers.
- A simpler interface, including a subset of controls, is displayed on an embedded OLED screen on the front panel of the device. This interface can be controlled with a few buttons and knobs on the front panel, and allows adjustment of the most important parameters. A sub-menu will be dedicated to the object-based renderer parameters.

#### 4.1.2 Web browser

While many extensions and plug-ins exist that allow for the use a wide range of interaction systems (head tracking devices, graphical tablets, external microphones etc.), the main goal of a web-based user interface for object-based audio consumption should be to provide a common feature set for all users. Learning and instruction time should be minimal. The various novel interaction possibilities of object-based audio consumption will be presented with a clear visual language.

As browser screens may vary dramatically between various platforms (from dual monitor setups in a desktop PC situation to tiny screens on mobile and wearable devices), the web-based user interface should be able to adapt itself in a responsive way.

### 4.1.3 Mobile device

Mobile devices typically offer a plethora of advanced input devices and sensors, which can be accessed without additional installations (gyroscopes, microphone, touchscreen, GPS, etc.). In a mobile app it is possible to control the entire user experience, allowing for innovative and dynamic user interfaces that can adapt to both the broadcast content and the end-user's preferences. The inclusion of advanced sensors in the user interface will allow for innovative interaction scenarios that expose the full potential of object-based audio.

The mobile app will not just deliver audio through headphones or built-in loudspeakers, but as is already commonly the case, can forward the audio to home-cinema or hi-fi sets, which may be capable of multi-loudspeaker immersive reproduction.

## 4.2 End-user requirements

### 4.2.1 Programme control

All radio reception systems will feature some form of programme selection, ranging from the tuning dials on conventional analogue receivers to the typically playlist-based systems in internet applications. In Orpheus, the focus will be on the development of user interfaces that present the unique and innovative possibilities of object-based broadcasting.

- *Programme pause/resume button*: in a non-linear consumption scenario, it will be possible to pause and resume playback in a flexible way. For a live broadcast, object-based audio supports the temporary buffering of content or near-live recall of an object after it has been paused, or the skipping of part of a programme to resume at a relevant point in the programme.
- *Jump back (index point)/rewind button*: along similar lines, it should be possible to go back in time and jump to the start of the currently-broadcast section, such as the start of an interview or a piece of music.
- *Swap button - for alternative content*: the extensive metadata that can be provided along with audio objects make it possible to identify the type and style of a programme section, and information on the time that the next segment of the programme will start. One thing this enables is to swap out an object for another, for instance to change the style of music that is played between the sections of an interview.
- *More information request button*: the audio object's metadata may also provide links to additional resources, such as websites or additional audio objects, or contain keywords that can be used in a web search. The client's user interface can therefore provide means to delve deeper into an object and listen to related content or display related textual information or images.
- *Language option selection - which languages do I prefer/understand setting*: a common problem in conventional broadcasting is how to handle languages other than the native one, for instance when interviews are done on location in a news program. With Orpheus it will be possible to choose between the original and a translation. In the user interface the listener can provide a list of preferred and understood languages, so the client program can automatically select the correct audio object. Similarly, in sports reporting, a commentator supporting the preferred club or country can be selected.

### 4.2.2 Audio quality

As object-based audio in Orpheus is rendered at the consumer side, it is possible to influence the rendering, using the metadata associated with the different objects.

- *Foreground/background mix adjustment*: one such application is the adjustment of the balance between foreground and background signals, either manually using a slider or using analytic means, for instance to increase the intelligibility of speech.
- *Auto-level - on/off (keep audio level above external noise)*: selection of automatic level and compression based on analysis of external / environmental noise (exploiting input from the integrated microphone) should be presented to the user in cases where this is relevant.
- *Dynamic range compression profile*: another option the user interface should provide is the selection between several compression profiles, to render optimally for specific scenarios.
- *Adaptation to network bandwidth*: in mobile scenarios, the selection of the number and quality or bitrate of audio objects is needed in cases where bandwidth is limited or expensive. The user interface should inform about the current quality and use of audio streams.

#### 4.2.2.1 Social interactivity

While the definition of a user model and associated infrastructure is beyond the scope of Orpheus, the pilots will experiment with some social / sharing features. Emphasis will lie on the new features of object-based broadcasting, such as the metadata provided, which allows for rich sharing. In particular, the timing information can be exploited, for instance if one wants to share a specific moment or a segment of a programme. Another question is how to account for the sender and receiver personalisation preferences when content is shared.

#### 4.2.3 Gesture control

- *Gesture sensors*: some of the new features proposed in Orpheus are quite complex and may be best interfaced using gestures as opposed to conventional buttons and lists. Navigation in a sound scene could for instance use information from the inertial module. User interface specifically designed for visually impaired people could also take advantage of such gesture controls.
- *Head tracking*: A specific input that will be supported in the Orpheus user interface is head tracking, such as provided by devices that can be mounted on headphones. Head tracking will provide a unique way to interact with spatialised audio. In some cases it might be desirable to turn the head tracking off. For instance, when sitting on a bus the objects should not move around the user as the bus turns. This issue could also be alleviated by a “cooldown” effect that slowly adapts the audio scene’s front direction to the direction the user is pointing to.

#### 4.2.4 Situational information

The Orpheus user interface will be able to use location and movement as inputs, for instance to select a specific rendering setup (like streaming to a multi-loudspeaker home hi-fi system or automatically compressing the sound to compensate for external noise on public transport). The end-user should be able to define locations or modes that are triggered by situational information.

#### 4.2.5 Feedback / interaction

In the object-based programme, calls to action may be included to incite the end-user to for instance respond to quizzes/questionnaires etc. or allow a direct dial-in to the radio studio. Orpheus will develop interfaces for these scenarios. While the handling of feedback and the integration into a content management system will be up to individual broadcasters, the Orpheus’ time-synchronized metadata enables the triggering and contextualizing of such interaction.

## 5 Conclusions

The technical and end-user requirements listed in the document represent a first basis for the development of the devices and applications for the reception and personalised rendering of broadcast object-based content. Although the proposed guidelines intend to demonstrate the large potential of object-based broadcasting, some practical limitations have been taken into account in order to achieve concrete results within the scope of the project. The on-going work on the design and the implementation of the pilots (WP2) will likely bring additional requirements.