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

ORPHEUS
Object-Based Audio Experience

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

D2.1: Initial Reference Architecture Specification Report

Version: v1.0

Deliverable type	R (Document, report)
Dissemination level	PU (Public)
Due date	30/06/2016
Submission date	12/07/2016
Lead editor	Michael Weitnauer (IRT)
Authors	Michael Weitnauer (IRT), Volker Mühle (MAGIX), Andreas Silzle (FHG), Chris Baume (BBC), Jean-Yves Aubie (BCOM), Olivier Warusfel (IRCAM), Niels Bogaards (ECANDY), Benjamin Duval (TRI)
Reviewers	Halid Hrasnica (EURES), Werner Bleisteiner (BR)
Work package, Task	WP2, T2.1
Keywords	Architecture, workflow, specification

Abstract

The deliverable D2.1 provides an overview about the current state of the pilot implementation architecture and its influence on the final reference architecture. The reference architecture will be developed during the project time, based on experiences from the project pilots. A detailed explanation regarding the distinction between reference architecture and pilots is included. The planned workflow of the pilots is described as well as the current state of macroblocks and their components. This document will be updated twice during the project.

[End of abstract]

Document revision history

Version	Date	Description of change	List of contributor(s)
v0.1	23/05/2016	Initial version	IRT
v0.2	02/06/2016	Contributions for Macroblock descriptions included	MAGIX, FHG, BBC, BCOM
v0.3	02/06/2016	Input from IRCAM added and section 1.1 changed	IRCAM, IRT
v0.4	02/06/2016	Comments and abbreviations added	
v0.5	07/06/2016	Updates for macroblock subsections, abstract included, executive summary added	BCOM, MAGIX, BBC, IRCAM, IRT
v0.6	08/06/2016	Section 2.3.5 updated, conclusion added	ECANDY, IRT
v0.7	09/06/2016	Document ready for review	
v0.8	27/06/2016	Review comments, input from BR and BBC	BR, BBC, EURES, IRT
v0.9	28/06/2016	Review comments addressed	
v1.0	31/06/2017	PMC comments addressed	

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[1].

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

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

Executive Summary

One of the major objectives of ORPHEUS is the specification of a reference architecture for an end-to-end object-based production workflow. The intention of such a reference architecture is to provide a guideline for other content producers and broadcasters for building an object-based production and distribution workflow and how to adapt established infrastructure.

To achieve this, the consortium is currently in the process of specifying the pilot implementation architecture which will be used for the pilots in the project. Based upon findings and lessons learned from the project pilots, the reference architecture will be specified during the project. It will be – compared to the pilot implementation architecture – rather format and interface agnostic wherever possible and reasonable in order to be applicable as a general guideline for other broadcasters.

The already defined parts of the pilot implementation architecture are described along with their components. Moreover, the current state and decisions taken so far are enclosed. Even though many decisions regarding interfaces and components or technologies are already taken, several parts of the pilot architecture need to be further refined in the next months of the project.

© 2015 - 2018 ORPHEUS Consortium Parties

Table of Contents

© 2015 - 2018 ORPHEUS Consortium Parties

List of Figures

© 2015 - 2018 ORPHEUS Consortium Parties

Abbreviations

DAW	Digital Audio Workstation
PCM	Pulse Code Modulation
ADM	Audio Definition Model
VST	Virtual Studio Technology
HOA	Higher Order Ambisonics
BW64	Broadcast Wave 64 Bit
IP	Internet Protocol
DAB+	Digital Audio Broadcasting +
DVB-S	Digital Video Broadcasting Satellite
ITU	International Telecommunication Union
MPEG	Moving Pictures Expert Group

© 2015 - 2018 ORPHEUS Consortium Parties

1 Introduction

1.1 Motivation

The object-based approach to media gives the most fundamental opportunity to re-imagine the creation, management and enjoyment of media since the invention of recording and broadcasting audio. One may argue that the core of the object-based approach to media is not new in itself, and that object-based audio has not been fully adopted so far, despite the fact that past and recent developments ([1][2][3]) have utilised some variation or parts of the object-based concept. This is due to the fact that previous attempts did not utilize the full creative and technical process from planning, editing, production and consumption of the object-based audio. This has led to isolated, partial solutions that did not take into account the whole or 'big picture' as required to unlock the full potential of an object-based production approach.

The ORPHEUS project therefore opts for an integrated method, targeting the end-to-end chain from production, storage, and play-out to distribution and reception. Only through this approach can it be ensured that the developed concepts are appropriate for real-world, day-to-day applications and are scalable from prototype implementations to large productions. In order to achieve this, the ORPHEUS project structure has been designed with a full media production chain in mind.

The core of this concept is formed by a pilot and a reference architecture for object-based media production. Thus, the reason for having a real-world pilot in the centre of the concept is not only for demonstration and evaluation purposes. Furthermore, it defines the requirements for an end-to-end chain that utilises the innovative features of object-based media while enforcing the constraints of a real-world, day-to-day production workflow. This approach ensures that every step of the chain will be considered with all shareholders in discussion to make the appropriate development possible.

The envisioned pilot will be implemented using the reference architecture developed and specified within the project. This reference architecture will provide specifications for the interfaces and components required for an end-to-end production chain. This will be beneficial beyond the projects lifetime, as it can serve as a guideline to set up, integrate or migrate an object-based production workflow. Apart from that, it can be used as a benchmark or test bed, ensuring the interoperability of potential future improvements of individual components within the chain. Work on the reference architecture and the implementation of the pilot will lead to an in-depth understanding of an object-based workflow and provide the tools required for creative object-based production.

1.2 Role of T2.1 for the project

WP2 can be seen as the ‘umbrella’ work package of ORPHEUS, as it defines interfaces, receives feedback from other work packages and makes decisions for the specific pilot implementations which are driven by the use cases. Figure 1 illustrates the central role of WP2 and T2.1 for the entire project.

The aim of Task 2.1 is the definition of interfaces, requirements and specifications of a reference architecture for an end-to-end object-based audio broadcasting chain. The result will be made available on an open-source basis if possible. The reference architecture will be a guide for future implementations of tools and building blocks, taking into account defined interfaces and best practices, as well as potential pitfalls and newly created standards and working practices by the project partners. Furthermore, it can be used as a reference point to test and evaluate new developments. An initial version of this reference architecture will be the basis for both phases of the pilot and will be refined further based on the results of the integration and testing process. It will also give the clearest indication yet of how media organisations can adapt to operate this technology at the scale required to run broadcast stations/services 24 hours a day, 7 days a week, 365 days a year.

© 2015 - 2018 ORPHEUS Consortium Parties

Although broadcasters are very experienced in this scale of operation, these challenges are not trivial. The technology will fundamentally change monitoring, automation, workflows and asset management and few organisations have the knowledge required to manage the metadata from capture to consumption.

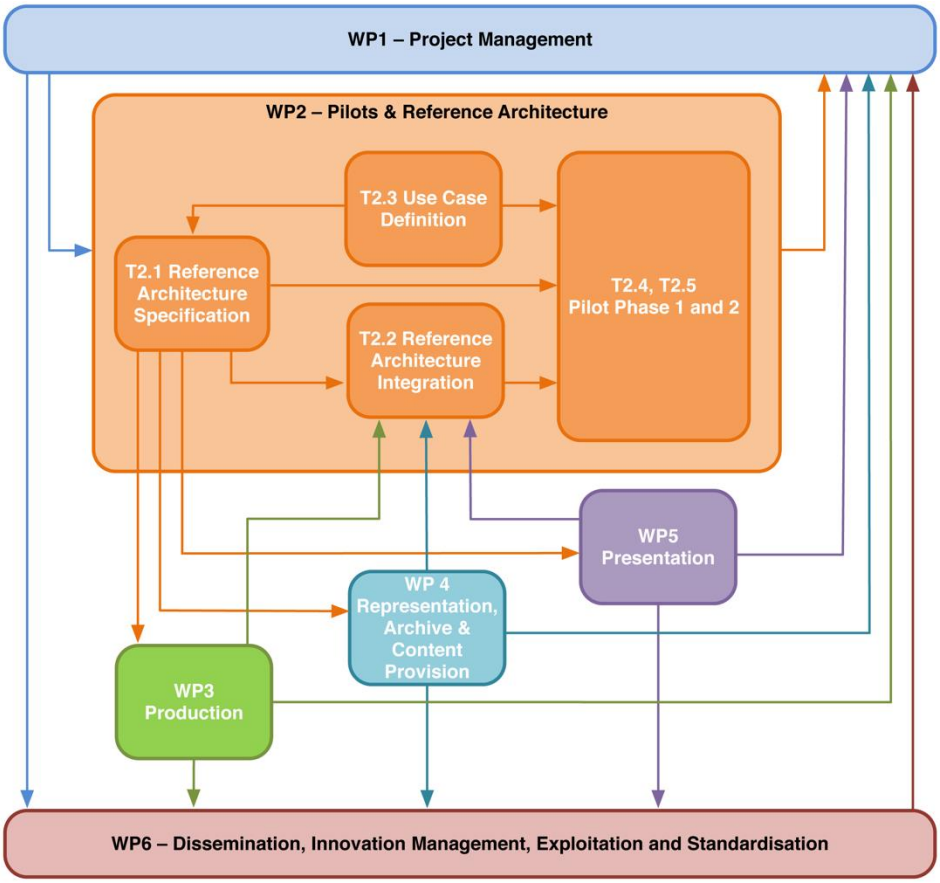

Figure 1: Pert chart of overall ORPHEUS WP structure

© 2015 - 2018 ORPHEUS Consortium Parties

2 Architecture Specification

2.1 Relation between pilots and reference architecture

The ultimate goal of Task 2.1 is the definition and specification of a reference architecture for object-based audio broadcasting. In order to avoid confusion, it has to be distinguished between the reference architecture and the ORPHEUS specific pilot implementation architecture.

The reference architecture will provide specifications of the required interfaces and components for an end-to-end production chain and will be beneficial beyond the projects lifetime, since it can serve as a general guideline to setup, integrate or migrate an object-based production workflow. Further, the reference architecture will be format, interface and protocol agnostic as much as possible and reasonable. On the other hand, the specific pilot implementation architecture will be based on the established infrastructure of the broadcaster partners in the project, on the available tools of other consortium members and on further technical outcomes of the project.

For several practical and economic reasons, the reference architecture will be based as much as possible on existing channel-based workflows since these workflows are the result of long-term experiences of broadcasters and content producers that guarantee the best quality of the final product in an economic manner.

The current focus of WP2 and T2.1 lies on the specification of the pilot implementation architecture with the implications on the reference architecture in mind, because the pilot implementation is currently most critical for the overall project progress and all other work packages and Tasks are influenced by those basic decisions.

Further, within ORPHEUS it will be the first time that a broadcasting workflow will be assembled for a complete end-to-end object-based structure and no experiences are available yet. Hence, for the definition and specification of a reference architecture, the actual pilot implementations are essential to gather the knowledge and experience for the universal, implementation-agnostic reference architecture. This will ultimately lead to a step-by-step development of the final reference architecture.

2.2 Components and signal flow of the pilot implementation architecture

Figure 3 illustrates the current state of the planned pilot implementation architecture, with a signal flow graph. The graph contains the five identified so-called macroblocks:

- Recording
- Pre-production and Mixing
- Radio Studio
- Distribution
- Reception

These blocks are described in more detail in the subsections below. The macroblocks are independent from each other, have several smaller components and are somewhat coherent with the ORPHEUS work package structure. Each macroblock will potentially be situated at the venue of a different project partner. Interfaces and interconnections between macroblocks and their components are also illustrated in Figure 2.

© 2015 - 2018 ORPHEUS Consortium Parties

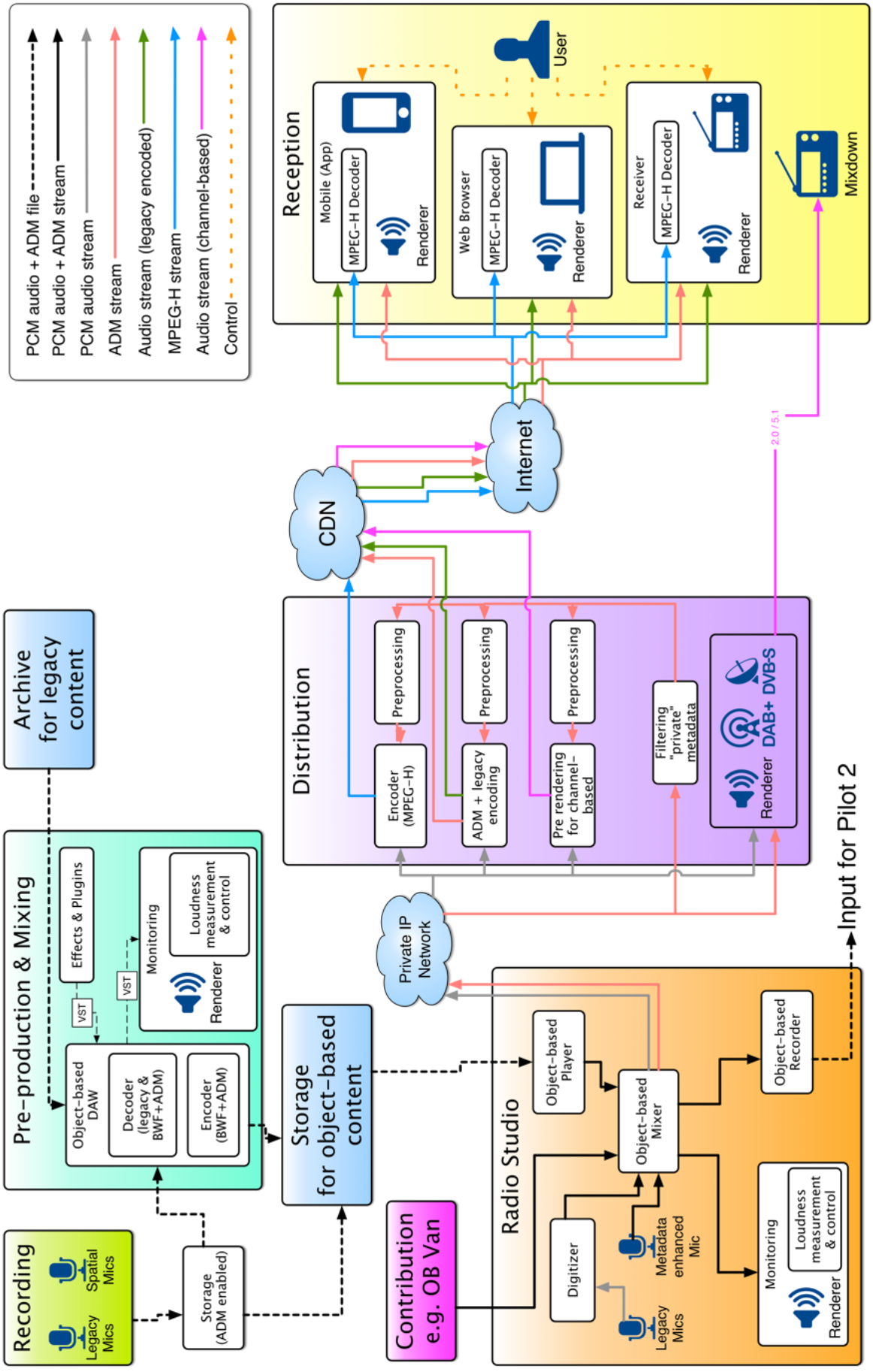

Figure 2: Current status of pilot implementation workflow overview

© 2015 - 2018 ORPHEUS Consortium Parties

2.3 Recording

The purpose of the recording macroblock is to provide the tools and infrastructure required to conduct object-based recordings. Single objects as well as entire audio scenes will be captured for the pilot, using both legacy microphones and novel microphone arrays which will be developed within the scope of ORPHEUS. This macroblock is therefore directly linked to T3.1 ("Capturing") developments and investigations. In the case where complex audio scenes are captured using microphone arrays, it is envisioned that ORPHEUS' object-based DAW will be employed. Thus this macroblock is also closely related to the pre-production and mixing macroblock (see Section 2.4 below). Additionally, the interface with other components and macroblocks will be file-based uncompressed PCM along with ADM metadata.

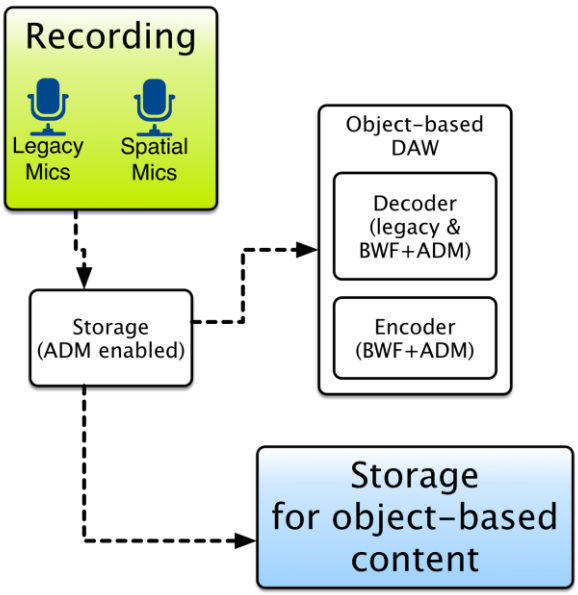

Figure 3: Current status of Recording macroblock

The aim of the recording macroblock is to provide audio recording solutions for the three main capture formats:

- channel-based (including mono, stereo and more spatialized formats like 7.1+4)
- object-based (typically one mono or stereo track for each object)
- scene-based (typically HOA (Higher Order Ambisonics) which can also be seen as an object)

It is worth noting that any capture format may be converted to another representation if required. For instance, a scene-based recording can be turned into audio objects by extracting dominant sources. Similarly, channel-based content can easily be converted to HOA signals, and vice versa.

Accordingly, recording systems must be chosen depending on their capabilities in terms of fidelity and usability, rather than for their native recording format. For instance, conventional mono or stereo microphones, as well as multi-channel capture systems (IRT Cross, Double MS, ABXY, etc.) may be used in the pilot to record ambient or directional sounds. Moreover, spherical or circular microphone arrays will be considered for their compactness and their ability to record the entire sound field from a single viewpoint. In the typical sound capture scenario, it is envisioned to employ both type of microphones, with "legacy" mics used primarily for distinct sound objects (e.g., a person talking) while microphone arrays are mostly used to capture "ambient sound" objects.

Among the technologies that will be investigated in task T3.1 (Capturing), the ones that will be integrated in the pilot are described in the following sections.

© 2015 - 2018 ORPHEUS Consortium Parties

Microphone array processor for HOA conversion

This “microphone array processor” plugin will convert standard microphone signals into HOA signals. As mentioned above, HOA signals will typically constitute audio ambience objects in recording scenarios. This tool will be delivered as a VST plugin for easy integration in a DAW or a mixer. The plugin will typically convert 32 microphone signals into HOA signals up to order 3 or 4, which correspond to 16 and 25 channels, respectively. Therefore, the DAW provided by Magix will be upgraded to support up to 32 channels per track.

Integration of a basic version of the plugin in the DAW is currently under test. The VST plugin should be available for integration in the pilot from July 2016.

Spot and HOA microphone signal analysis plugin for delay and position extraction

This VST plugin facilitates the spatial mixing of spot microphone signals with respect to a main, 3D Ambisonics (or HOA) microphone capture. This helps integrating objects corresponding to distinct sound sources (e.g., recorded with HF microphones) with ambience objects recorded with circular or spherical microphone arrays. Also, it will be possible to generate object trajectories using the delay and position data computed by the plugin.

A real-time version of the algorithm is currently being finalized and the development of a corresponding GUI is considered within the scope of the project, as well as further improvements of the algorithms.

A first VST plugin (with a basic GUI) has been developed and will be available from July 2016.

HOA scene analysis for object extraction and scene manipulation

This VST plugin will offer to the sound engineer the possibility to suppress, extract, move and/or amplify detected sound sources independently. It will thus make it possible to extract one or several sound objects from recordings done with circular or spherical microphone arrays. Also, it will allow sound engineers to edit ambient sound objects by attenuating or reinforcing sounds incoming from certain areas of space.

A first version of a VST plugin implementing such functionalities is currently being built. The use of such kind of effect in pilots is currently hypothetical and has to be confirmed. Nevertheless, a first shareable and demonstrable version of the VST plugin is envisaged in Q3 2016. Further improvements will be expected even after that.

Real-time generation of 7+4 channels / HOA from a circular microphone array recording

This technology allows to generate 7+4 loudspeaker signals (i.e., 7 mid-layer channels and 4 top-layer channels) as well as HOA signals using a compact circular microphone array. The discussion to integrate it as a VST plugin is still ongoing.

An offline version of the algorithm is currently being tested. A real-time version is expected to be ready in Q3 2016.

2.4 Pre-production and Mixing

The purpose of the pre-production and mixing block is to deliver tools for editing existing object-based content or creating such content from legacy audio material or other sources. The core of this block is the object-based DAW, which is extended by several tools and workflows to import, edit, monitor and export object-based content. These tools and workflows are developed and

© 2015 - 2018 ORPHEUS Consortium Parties

implemented in task T3.2. The DAW will need to interact with the components presented in the section below and illustrated in Figure 4.

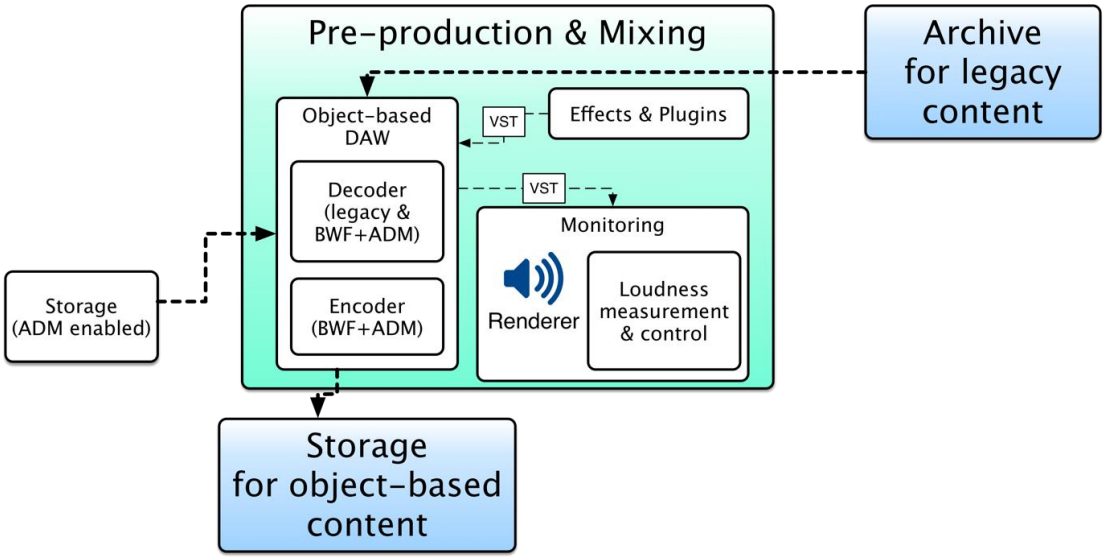

Figure 4: Current status of Pre-production & Mixing macroblock

Recording

The recording of object-based content may be done independently from the DAW. In this case the recorded content is transferred in form of multichannel BW64 files with ADM metadata. Depending on the established broadcast architecture, these files may be stored in a temporary **ADM enabled storage** or directly in the **storage for object-based content.**

Additionally, there may be a way to record object-based content directly within the DAW. This implicates further requirements for the DAW (e.g. 32 input channels for the microphone array processing described in the recording macro block).

Storage for object-based content

The DAW will be able to import and export object-based content in the form of uncompressed multichannel BW64 files with ADM metadata. This involves **decoding** and **encoding** these files including their metadata. There will be tools to edit existing ADM metadata or additional metadata required for broadcasting.

Further the DAW may support direct import and export of MPEG-H as object–based distribution format, as covered in WP4. This allows additional workflows, e.g. for quality control.

Archive for legacy content

For integration into an established broadcasting architecture the access to existing content is very important. For this purpose, there have to be tools inside or outside the DAW to convert legacy content to object-based content by adding the required metadata to it. This process should at least partially be done automated - if possible.

Effects and plugins

A typical application for a DAW is the appliance of effects or plugins onto audio tracks. Basically, applying an effect consists in taking a given sound signal and changing it somehow. Typical effects used in audio production are filtering effects (changing the frequency content of a sound) or dynamic effects (limiter or compressor, to change the dynamic of an audio track). Plugins are extensions for

© 2015 - 2018 ORPHEUS Consortium Parties

the DAW to be used with audio tracks such as special filter effects.

Usually effects and plugins can be applied to audio content within the DAW. Due to the nature of the ADM format there are some limitations to consider here. Especially there is no final mastering stage for a certain channel format, but effects can be applied to individual audio objects only.

For the scope of ORPHEUS, it is decided that any relevant plugin should be made available as VST 2 plugins – at least for the pilots. This may also include format conversions e.g. converting 3D microphone input to HOA format or converting HOA format to channel-based surround formats. Plugins for these use cases already exist and can be delivered e.g. by B-COM.

A special use case is applying reverb to the content. Currently reverb could only be included as a fixed part of the individual objects or as an additional audio object, which partially limits the possibilities of ADM. An object-based reverb algorithm and inclusion into ADM format is investigated by IRCAM in the scope of T3.2.

Monitoring

For previewing the object-based content and simulating the user experience it is important to allow monitoring with different speaker setups (including binaural monitoring). For this purpose, an ADM renderer needs to be implemented in the DAW.

The ITU-R baseline renderer should be taken for the reference architecture. However, it will not be available before mid-2017. Meanwhile, it is decided to use the MPEG-H renderer solution of FHG, especially for the development of the pilots.

An essential part of the monitoring is **loudness measurement and control.** This may be supported by additional VST plugins in the DAW and is researched in T3.4.

In addition to pure audio monitoring, other parts of the user experience may be simulated within the DAW, e.g. transport control or exchanging language specific objects. This will be researched in T3.4 and possibly in T5.4.

Another solution for the monitoring is to stream the content as an ADM compatible stream (as dealt with in T4.2), e.g. to support alternative real-time monitoring renderers or integration into live production environments (T3.3/3.4).

2.5 Radio Studio

The studio is where all of the different source material comes together and is produced into a ‘radio programme’, consisting of many different audio elements. The technical requirements in the control room of a radio studio include: capturing of audio signals, routing of external audio sources, playback of pre-recorded material, monitoring of audio quality and adding ‘content metadata’. Each of these elements are discussed in details below and illustrated in Figure 5:

© 2015 - 2018 ORPHEUS Consortium Parties

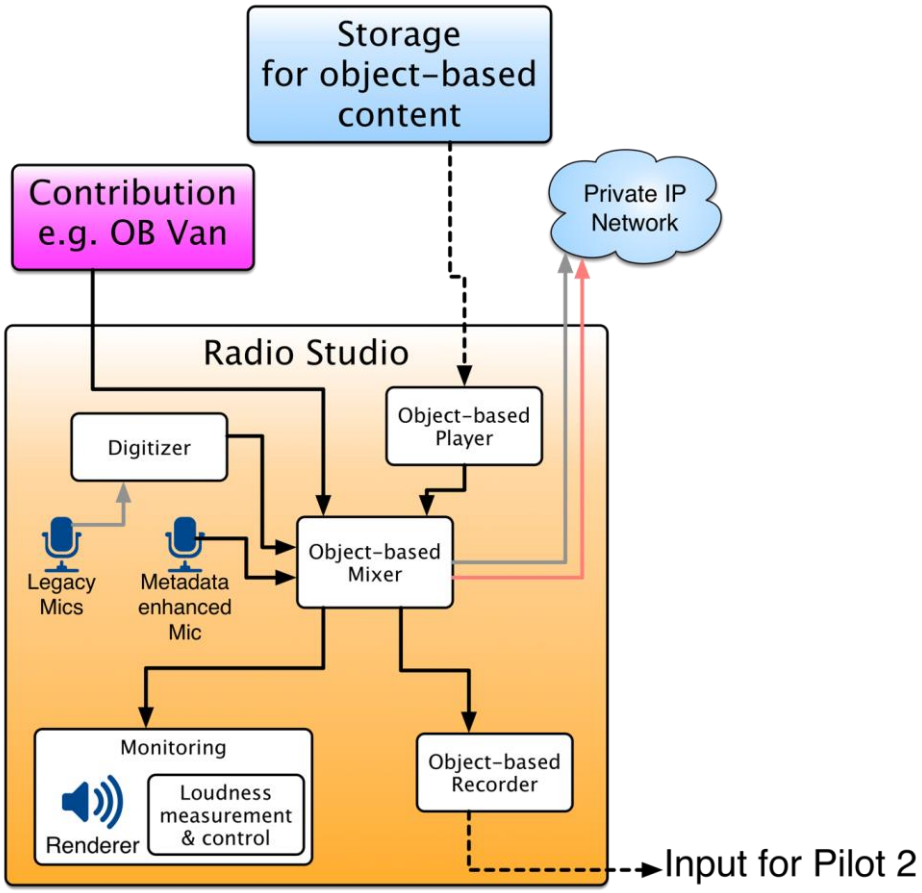

Figure 5: Current status of Radio Studio macroblock

Audio capture:

Within the studio, audio is captured using microphones, typically one per person. However, unlike traditional studios the microphone signals will be enhanced with additional metadata, which for example will include the model of the microphone, its location within the building, and the identity of the person currently speaking into the microphone. The capture and transmission of this information is covered in Task 3.3.

For many programmes, all of the audio content is sourced from the studio, so this macroblock can operate independently of any other macroblock. However, most programmes contain an element of pre-produced material (from Task 3.2) or outside sources (from Task 3.1 for instance).

Initially the pilot studio will use traditional microphones with IP convertors to capture audio content. Screen-based interfaces (GUIs) will be used to capture additional metadata about the audio content, including the identity of the person using the microphone. Later in the project, we will investigate the use of RFID badges to identify users. We will also investigate the possibility of using native IP-connected microphones, as and when they become available.

Object-based mixer:

Traditionally, the 'mixer' is where audio channels are mixed together to form a single audio stream of the programme. In an object-based production, this concept is radically different in that elements are not usually mixed, but kept separate. The 'mixer' in this context performs the same overall function of a mixing desk, which is bringing together multiple sources into a single experience, but doing so through the manipulation of metadata.

© 2015 - 2018 ORPHEUS Consortium Parties

The mixer allows the engineer to select which audio sources form part of the programme, and to set the gain and panning (in 3D) through the adjustment of each source's metadata. It also adds additional content metadata to the programme such as the programme's name, and a title and description of the current segment. This additional data must be captured as part of the production process through the use of appropriate interfaces within the workflow. All of this work is covered in Task 3.3.

ORPHEUS will develop a system that enables producers to create a running order and script for their programme. This system will capture information about the programme and its elements, including the segments within the programme and the contributors. The data captured by the system will integrate with other parts of the studio, and help to deliver an improved audience experience.

Furthermore, the project will investigate the possibility of using IP-connected mixing consoles to ingest and control audio signals. The success of this will rely on being able to extract the appropriate level of metadata (such as fader/pot positions) and integrate this with the IP Studio framework.

For the purposes of the pilots, we will attempt to use an MPEG-H renderer to listen to the output of the mix. The pilot studio will have a surround-sound monitoring set-up with height (depending on physical limitations). This will later be enhanced with binaural rendering.

Object-based player/recorder:

The studio is where audio content is produced live and there needs to be a mechanism to record and replay material. Traditionally this is achieved by recording the stereo/5.1 output of the mixing desk. In an object-based production each source is recorded individually along with the associated metadata. It is expected that this will be achieved by capturing the data stream from the mixer, with additional functionality to import/export from/to BW64 audio files with ADM-described metadata. The object-based player/recorder is covered in Task 3.3.

We will use the existing IP Studio[4][5] sequence store to capture and replay object-based content. Later, we will develop software to export (and possible import) this data to/from ADM-described BW64 files.

Monitoring:

In order to ensure quality standards of the programme, the output of the studio is to be monitored. This includes monitoring of audio quality , i.e. signal levels, perceptual loudness (including 3D panned sources), quality of both 'content' and format metadata which also forms part of the audience experience and finally, the quality of the network connections.

These activities are made more challenging through new, typically 'object-based' interactive elements of the programme, as there may be more than one experience to monitor at any point in time. All of this work is covered in Task 3.4.

We will start by investigating existing tools available for monitoring perceptual loudness of 3D panned audio, and other metrics that are significant to the listening experience. We will use the existing network monitoring tools that are part of the IP Studio package.

2.6 Distribution

This macroblock contains the modules and tools needed for distribution, illustrated by Figure 6. The distribution block received the object-based audio data via a private IP network (PCM audio data plus an ADM metadata stream).

© 2015 - 2018 ORPHEUS Consortium Parties

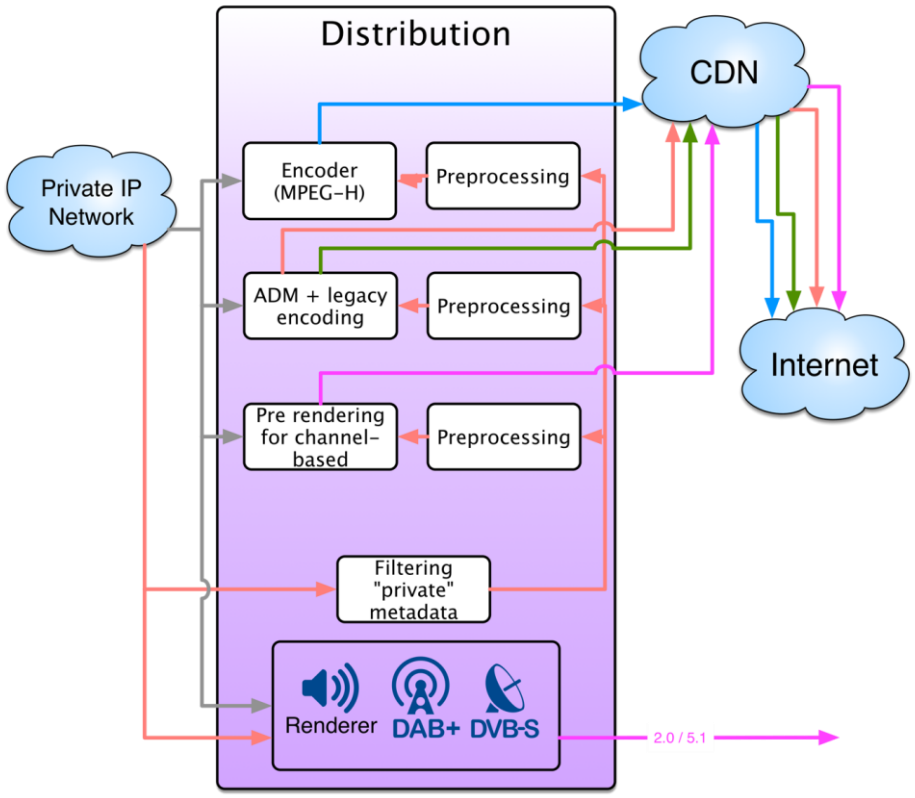

Figure 6: Current status of Distribution macroblock

Two distribution paths are considered:

- A) The distribution of the object-based content via a content delivery network (CDN)
- B) The distribution of a channel-based downmix of the object-based content, i.e. a pre-rendered version, delivered via DAB+ and/or DVB-S.

For case A), the received metadata needs to be pre-processed as a preparation for distribution. The pre-processing depends on the distribution format. First, any information that is 'private' or used solely for internal purposes needs to be removed from the input metadata. As a second pre-processing step, the ADM metadata might need to be either translated into a different metadata representation or a new/modified ADM metadata stream needs to be created (e.g. due to a needed reduction of the overall number of objects), based on the capabilities and needs of the distribution formats.

Several formats are considered:

- Distribution as an MPEG-H stream: The ADM metadata needs to be translated to MPEG-H metadata. A reduction of the total number of objects or a combination of several objects into one audio track might be needed, dependent on the chosen MPEG-H configuration.
- Distribution as ADM metadata plus legacy audio encoding: A reduction of the total number of objects or a combination of several objects into one audio track might be needed, dependent on the chosen legacy encoding and the maximum number of supported tracks.
- Distribution as a pre-rendered channel-based version: Rendering is needed to create a channel-based version.

For case B), a renderer is needed to produce the pre-rendered version.

This block is mainly linked to the Tasks of Work Package 4 (Representation, Archive & Content Provision), especially T4.1 (Codecs and Formats) and T4.3 (Delivery to the end user).

© 2015 - 2018 ORPHEUS Consortium Parties

2.7 Reception

The purpose of this macroblock is to provide hardware and software solutions for the reception and reproduction of object-based audio content for the end-users. All solutions are comprised of two major components, a **decoding/rendering module** and a **user interface**. The decoding and rendering module automatically adapts the play-out of the object-based content according to the end-user environment's setup, i.e. from conventional loudspeaker setups to advanced multi-channel immersive audio systems or binaural rendering over headphones. The user interface allows for various personalised and interactive reproduction of audio content. Figure 7 illustrates the current state of the macroblock.

Several hardware and software solutions have to be considered in order to meet different segments of the consumer electronics market, as well as different end-user listening habits and audio content consumption contexts. 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 contexts (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 general features of the decoding/rendering module and user interface are described hereafter. When relevant, information specific to each hardware or software solutions are detailed.

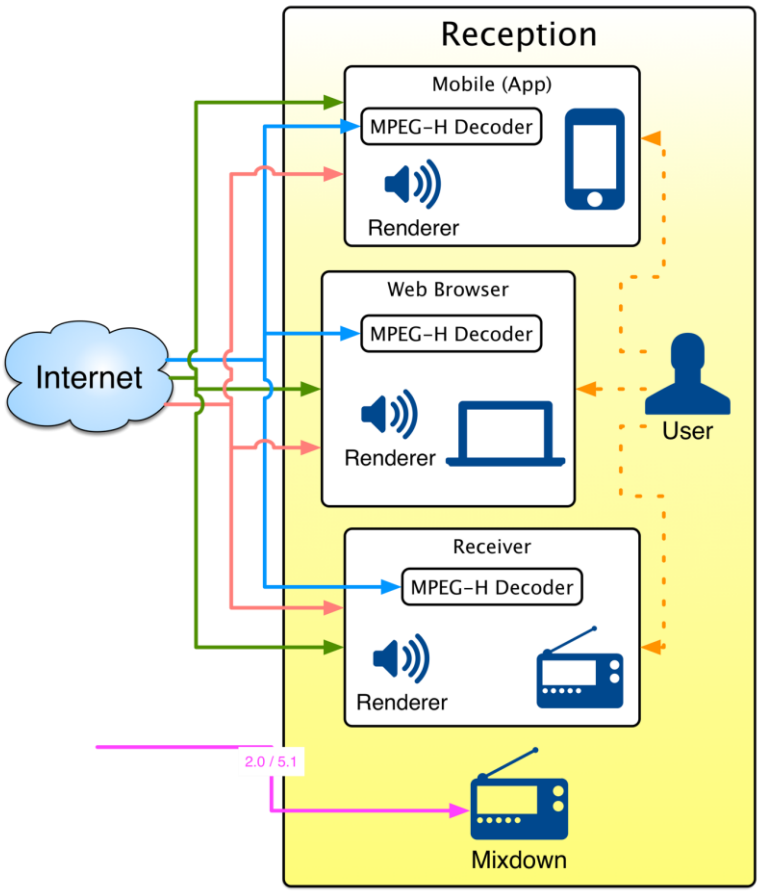

Figure 7: Current status of Reception macroblock

© 2015 - 2018 ORPHEUS Consortium Parties

Decoding/Rendering module

The hardware or software solutions 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 + legacy encoding of audio objects. Ultimately, the receiver should also provide means for rendering pre-rendered channel-based content.

According to the available bandwidth of the network and processing power of the end-user device, the receiver selects/asks for an appropriate streaming format, i.e. where some audio-objects are kept independent while others are pre-combined or pre-rendered as 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 may be selected based on user preference (such as language, available listening time, etc.). The compatibility of existing solutions such as MPEG-DASH standard for adaptive streaming of object-based audio still needs to be checked.

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

The ITU-R baseline renderer should be taken for the reference architecture. However, it will not be available before mid-2017. Meanwhile, it is decided to use the MPEG-H encoder-renderer solution of FHG, especially for the development of the pilot #1. FHG is delivering MPEG-H decoders/renderers as a C/C++ library. Next steps will consider the ITU solution.

BBC has developed an audio toolbox (bbcat) available on github. It includes ADM streaming (ADM support library) but does not support neither HOA nor channel-based formats.

Hardware receiver

The hardware receiver is a standalone processor. It fulfils all the requirements by itself, and does not need any third party application. Thus, every processing is controlled, and no compatibility issues need to be addressed.

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 are already 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 existing signal processing flow in the same way. Two solutions are considered:

- embedding FHG's MPEG-H renderer based on the C++ library provided by FHG
- implementing a custom ADM + legacy renderer; the ITU-R baseline renderer could serve this purpose as well.

Web browser

Today, a HTML5 capable web browser can be considered as the most universally available reception system. Its ubiquity means that a very large audience can be exposed to object-based broadcasting without any extra effort required (such as downloading plugins, 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:

- FHG is currently developing solutions to embed an MPEG-H renderer in the web browser. In the pilot phase, this will only be feasible for the open source Chromium browser.
- implementing a rendering system based on JavaScript and the Web Audio API. Such solutions are currently under development at different partners (BBC, IRCAM, IRT). While a native

© 2015 - 2018 ORPHEUS Consortium Parties

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 plugin capable of rendering object-based audio (for instance using MPEG-H). As browser support for third party plugins is declining, this option was rejected as not suitable for future use.

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 renderer. Provided as a C++ library, this renderer, possibly optimized 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. The ITU-R baseline renderer could serve this purpose as well.

Conventional receiver

For backward compatibility, conventional audio receivers can play-out pre-rendered channel based downmix (2.0 or 5.1 format) delivered on legacy broadcast distribution channels via DAB+ or DVB-S/DVB-C (see above section 2.6).

User interface

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 context (e.g. propose 3D fully immersive audio or more conventional play-back, 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 (if applicable), displaying various types of 'content metadata' (i.e. music title, performer, composer, author, names of radio program participants) as well as text transcription etc.

The design and development of the user interface will follow on-going discussions on the definition of use cases (T2.3) and personalisation features envisioned for pilot #1 and #2 (T5.2).

Hardware receiver

The 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 controlling. The object-based features will be addressed on a dedicated page, as well as the parameters of each of the decoders/renderers.
- A simplest 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 adjusting the most important parameters. A sub-menu will be dedicated to the object-based renderer parameters.

© 2015 - 2018 ORPHEUS Consortium Parties

Web browser

While many extensions and plugins exist that allow for the use of 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. With a clear visual language, the various novel interaction possibilities of object-based audio consumption will be presented.

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

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 broadcasted content as the end-user's preferences. The inclusion of advanced sensors in the user interface will allow for innovative interaction scenario's that expose the full potential of object-based audio.

© 2015 - 2018 ORPHEUS Consortium Parties

3 Conclusions

To specify a reference architecture for an end-to-end infrastructure for object-based audio, the ORPHEUS consortium defines a pilot implementation architecture to be used for the dedicated pilots during the project. This pilot implementation architecture shall serve as basis for the reference architecture by taking into account findings and lessons learned during the pilot phases.

Even though many parts of the pilot architecture are already defined, several decisions regarding interfaces and components need to be taken in the next months of ORPHEUS. This document will be updated twice during the project running time (M14 and M27).

© 2015 - 2018 ORPHEUS Consortium Parties

References

[1] Scheirer E., Väänänen R. 1999. AudioBIFS: Describing Audio Scenes with the MPEG-4 Multimedia Standard, in IEEE Transactions On Multimedia 1, Nr. 3

[2] Geier M., Spors, S. 2008. ASDF: Audio Scene Description Format. International Computer Music Conference (ICMC), Belfast, UK

[3] Peters N. et al. 2012. SpatDIF: Principles, Specification, and Examples. Proceedings of SMC 2012

[4] Peter Brightwell, Jonathan Rosser, Robert Wadge, Phil Tudor, "The IP Studio", BBC R&D White paper WHP 268, Sep 2013

[5] http://www.bbc.co.uk/rd/projects/ip-studio

[end of document]

© 2015 - 2018 ORPHEUS Consortium Parties