DRAFT GUIDELINES

Version 0.0draft018 2017-12-22

VR Industry Forum

5177 Brandin Court

Fremont, CA 94538, USA

www.vr-if.org

1 Introduction

The guidelines presented in this document cover all aspects of the distribution ecosystems, including compression, storage and delivery, in order to ensure high quality, comfortable consumer VR experiences. These guidelines are aimed at addressing best practices for VR content distribution as well as advocating interoperability and deployment guidelines based on common technical standards for VR content distribution, including promoting the use of common profiles across the industry.

The targeted audience includes content distributors, service providers, broadcasters, mobile operators, consumer electronics manufacturers, professional equipment manufacturers, software developers and technology companies that aim to enable deployment of VR content distribution services.

The scope of the guidelines presented herein includes:

- Production: Technical aspects of the media formats used in the interface between the content provider and the service provider along with human factors considerations for compelling and usable 360° video experiences.
- Compression: Media codecs for VR, i.e. encoding of different production formats and related media profiles for video, audio and possibly also other media types such as text, graphics, etc.. This includes decoding and rendering of the media based on an abstracted distribution data model.
- Storage: Media formats for VR content (e.g. file/segment encapsulation) for different distribution means, including but not limited to storage, download, adaptive bitrate streaming and broadcasting
- Delivery: Interfaces and protocols for Live, Linear and VOD delivery over streaming (unicast), and broadcast applications
- Security: VR specific threat identification and mitigation techniques as well as methods for implementing security and privacy protection functions.

Contents

Figures

Tables

2 References

[3DA] Information technology – High efficiency coding and media delivery in heterogeneous environments – Part 3: 3D Audio, Second Edition (ISO/IEC 23008-3)

[3DTECH] ANSES, 3D technologies and eyesight

[80211AC] Standard for Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications - Amendment 4: Enhancements for Very High Throughput for Operation in Bands below 6 GHz. (IEEE 802.11ac)

[80211AX] Standard for Information Technology - Telecommunications and Information Exchange Between Systems Local and Metropolitan Area Networks - Specific Requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment Enhancements for High Efficiency WLAN (IEEE 802.11ax)

[ACN] Ambisonic data exchange formats – Ambisonic Channel Number (Wikipedia)

[ADM] Audio Definition Model (ITU-R BS.2076), BBC Software

[ADDSEI] HEVC Additional Supplemental Enhancement Information (JCTVC AC1005)

[AOA-FAQ] American Optometric Association, FAQ

[BS2088] Long-form file format for the international exchange of audio programme materials with metadata (ITU-R BS.2088)

[BT2020] Parameter values for ultra-high definition television systems for production and international programme exchange (ITU-R BT.2020)

[BT709] Parameter values for the HDTV standards for production and international programme exchange (ITU-R BT.709)

[CDNOPT] TiledMedia, CDN Optimization for VR Streaming

[DASH] Information technology – Dynamic adaptive streaming over HTTP (DASH) – Part 1: Media presentation description and segment formats, including ISO/IEC 23009-1:2014/Amd.4:2016 (ISO/IEC 23009-1)

[DASHIFIOP] DASH-IF, Interoperability Points v4.1

[DASHPUSH] Information technology – Dynamic adaptive streaming over HTTP (DASH) – Part 6: DASH with server push and WebSockets (ISO/IEC 23009-6)

[DAW1] Qualcomm, 3D Audio Tools

[DRC] Information technology – MPEG audio technologies – Part 4: Dynamic Range Control (ISO/IEC 23003-4)

[ECP] MovieLabs Specification for Enhanced Content Protection (ECP V1.1)

[EME] W3C, Encrypted Media Extensions

[FRATES] Les casques de réalité virtuelle et de jeux video

[ISOBMFF] Information technology – Coding of audio-visual objects – Part 12: ISO Base Media File Format (ISO/IEC 14496-12)

[ISOCMAF] Information technology – Coding of audio-visual objects Information technology – Multimedia application format (MPEG-A) – Part 19: Common Media Application Format (ISO/IEC 23000-19)

[JAUNT360] Jaunt Studios, The Cinematic VR Field Guide: A Guide to Best Practices for Shooting 360°
[MP4FF] Information technology – Coding of audio-visual objects – Part 14: The MP4 File Format (ISO/IEC 14496-14)

[MP4SYS] Information technology – Coding of audio-visual objects – Part 1: Systems (ISO/IEC 14496-1)

[N16584] MPEG-H 3D Audio Verification Test Report (MPEG N16584)

[NAL] Information technology – Coding of audio-visual objects – Part 15: Carriage of network abstraction layer (NAL) unit structured video in the ISO base media file format (ISO/IEC 14496-15:2017)

[NORMS] Ambisonic data exchange formats – Normalization (Wikipedia)

[OMAFFDIS] Text of ISO/IEC FDIS 23090-2 Omnidirectional Media Format (MPEG W17235)

[PROJCONV] Y. Ye, E. Alshina, J. Boyce (editors) Algorithm descriptions of projection format conversion and video quality metrics in 360Lib Version 4, (JVET G1003, MPEG M41359)

[Pulkki] V. Pulkki, "Virtual sound source positioning using vector base amplitude panning" Audio Engineering Society Journal, vol. 45, no. 6, pp. 456–466, 1997.

[R103] Video Signal Tolerance in Digital Television Systems (EBU R103)

[R128] Loudness Normalization and Permitted Maximum Level of Audio Signals (EBU R128)

[RFC6381] The 'Codecs' and 'Profiles' Parameters for "Bucket" Media Types (IETF RFC 6381)

[TR26918] Virtual Reality (VR) media services over 3GPP (3GPP TR26.918)

[TS1015472] Plano-stereoscopic 3DTV; Part 2: Frame Compatible Plano-stereoscopic 3DTV (ETSI TS 101 572-2)

[TS26234] Packet Switched Streaming Service (3GPP TS 26.234)

[VDVS-QUAL] Subjective Quality Results of Viewport-Dependent Omnidirectional Video Streaming (3GPP S4-170637)

3 Conventions and Terminology

3.1 Definitions

Additional terms are defined in the VRIF Lexicon, available at http://www.vr-if.org/lexicon/

Term	Definition
360° Content	Post produced or Live 360° Video content with limited interactivity
3DOF	3 Degrees of Freedom – a rendering model whereby the viewing pose is only alterable through rotations on the x, y and z axes. These rotations represent roll, pitch and yaw respectively.
6DOF	6 Degrees of Freedom – an extension of the 3DOF rendering model in which translations along the x, y, and z axes are also permitted. These translations represent the forward-backward, left-right and up-down motions respectively.
Equirectangular	Projection of a spherical image in a rectilinear image frame
Haptic Feedback	Palpable feedback, usually servo-driven in human interfaces (hand controllers)
Nadir	Lowest point on a sphere antipodal to the Zenith. More generally, the point (or direction) represented by an elevation of -90° in any spherical coordinate system. Often used to refer to dead-spot in image below camera array. See also "Zenith".
Nodal point	The point at which the optical centers of the camera array are cantered
Simulator Sickness	Uneasiness or disorientation when wearing a Head-Mounted Display
Spatial Audio	Audio which is perceived to have orientation in three possible axes: Left/Right, Up/Down, Back/Front
Spherical Video	Video content captured or displayed simultaneously in all directions
Stitch Lines	Visible junctures of blended or overlapped images from multiple camera arrays
Unity	(Game Engine) A cross-platform engine for game and VR design and implementation
Vection	The visual peripheral information able to induce the illusion of self-motion.
Zenith	Highest point on a sphere, antipodal to the Nadir. More generally, the point (or direction) represented by an elevation of +90° in any spherical coordinate system. Often used to refer to dead-spot in image above camera array. See also "Nadir".

3.2 Abbreviations

Definitions of most of these are available in the VRIF Lexicon, http://www.vr-if.org/lexicon/

Abbreviation	Definition
ϕ	Azimuth (longitude, increasing eastward)
Θ	Elevation (latitude, increasing northward)
AES	Advanced Encryption Standard
BRIR	Binaural Room Impulse Response
CMP	Cube Map Projection
DAW	Digital Audio Workstation
DOF	Degrees of Freedom
ERP	Equirectangular Projection
FOA	First Order Ambisonics
FOV	Field of View
HDCP	High-bandwidth Digital Content Protection
HMD	Head Mounted Display
HOA	Higher Order Ambisonics
HRTF	Head Related Transfer Function
IPD	Interpupillary distance
MCTS	Motion Constrained Tile Set
POV	Point of View
SBA	Scene-Based Audio
SEI	Supplemental Enhancement Information
VOD	Video On Demand
VR	Virtual Reality

4 VR Technologies

4.1 Production Guidelines for VR Audio and Video Content

4.1.1 User Experience

360° content created for display in a Head Mounted Display (HMD) should allow the viewer to engage in a unique and immersive manner. The viewer may be separated from visual cues from the real environment and placed in a virtual environment or experience.

The uniqueness of the 360° video experiences is mostly due to two factors:

- The level of immersion induced by the wider field of view of the HMD
- The interactivity of the scene that respond to the user actions (at the lowest level, such interactivity is reduced to the ability to visually explore (look around) the scene like in the case of 3DOF)

These two elements imply some particular precautions to ensure that the content created should not lead to disorientation and unease.

From an audio perspective, content should be produced with an understanding that visual content is not primarily frontal as is assumed for linear TV consumption. Further, since audio is consumed over headphones, acoustic cues can be created from all directions including below ear level – something not usually done for loudspeaker consumption (since most loudspeaker configurations don't include loudspeakers below ear level).

It should be noted that due to the environmental isolation resulting from the unique and immersive nature of HMDs, viewers may experience discomfort when consuming 360° video experiences while in-transit. This discomfort is caused by either (a) the positional sensors in the HMD being affected by changes in the transit system resulting in unstimulated viewport change, or (b) the incoherence between the vection and the vestibular information.

4.1.2 Considerations for acquisition and creation

The following sub-sections provide guidance on certain factors which play a significant role in the creation of high quality VR content that can be viewed in a comfortable manner. Additional technical and artistic consideration can be found in [JAUNT360].

4.1.2.1 Field of view consideration for video

There is a significant difference between the human Field Of View (FOV) and the one covered by current commercial VR devices. The human horizontal FOV is estimated to be 190° (without eye rotation) and 220° (with eye rotation) while the vertical FOV is 120° (50° upwards and 70° downwards). The head mounted displays vary in the FOV that they can reproduce, but generally have a 60° to 90° per eye FOV range (for instance HTC Vive, Oculus Rift and Sony PlayStation VR claim to have a 110° total field of view, while smartphone based solution usually reach 95°). While these values are still far from the human FOV, they are notably wider than those of traditional devices like TV screens that, when viewed at recommended distances, cover approximately 40° of FOV.

The difference in the wideness of the FOV between HMD and traditional displays has various consequences for the production of the media. In particular a wider field of view implies a bigger amount of “vection” (visual peripheral information able to induce the illusion of self-motion) that could induce discomfort symptoms characteristic of the Virtual Reality sickness.

Virtual Reality sickness is induced by the discrepancy between motion cues coming from the vestibular and the visual systems. It is important to notice that the vestibular system is able to give information concerning the orientation and acceleration (of the head, and consequently of the user) but is unable to distinguish between staying still and moving straight at a constant speed.

4.1.2.2 Content Position

While the human field of view is 190°, the perceptual capacities of the visual system are not the same over the whole visual field. The ability to recognize colors for instance is limited to a central area of about 120°. Shape is recognizable around 60° from the center and texts are readable when they are at less than 20° from the center of the eye (fovea). As a consequence, humans tend to keep the most relevant visual content in the central area of their FOV, by constantly moving their eye gaze and head if necessary. This motion could be uncomfortable in particular when wearing an HMD. During production attention has to be paid to reducing the need for head movement by keeping relevant content in the comfortable viewing area as well as providing some cues as to where any main action is occurring.

For example, content designed to be viewed in a seated position should be created such that the action in the scene requires that the viewer only moves their head through an arc of 90° to 120°. This will give an overall viewable arc of 180° to 210°, with significant action contained within a 90° range of head movement.

For content designed to be viewed in a standing position a wider arc may be considered.

4.1.2.3 Camera Motion

As previously stated, “vection” can induce VR sickness in the user, and a main source of “vection” is the camera motion.

Viewport changes that are the consequence of user movement (rotations in 3DOF and rotation and translation in 6DOF) could be VR sickness free if the system is reactive (less than 20 milliseconds of Motion-to-Photon delay) and accurate (1:1 scale) because the vection is coherent with the vestibular information.

In all other cases, viewport changes could potentially induce VR sickness in particular when the vection suggests orientations and accelerations that are not coherent with that which is perceived by the vestibular system. Rotational gain can be used in some situations but should only be applied in a predictable or user specified manner to minimize the risk of VR sickness.

To reduce the risk of VR sickness the main approach consists of reducing the amount of “vestibular incoherent” vection through different technique like:

- Avoid unnecessary change in speed and direction when performing camera motion, constant speed translations on a straight line are less disturbing.
- Avoid rotation of the camera that could influence the “horizontality of the horizon.”
- Whenever possible replace motion by adopting other metaphors (teleportation).
- The temporary reduction of the field of view sensibly reduce the strength of vection, peripheral vision is highly sensitive to motion cues[1].
- Avoid moving to close to objects and environmental elements that could induce high level of vection (walking close to a wall or very close to the ground in particular when they have repeated patterns)

[1] Method implemented in the VR-game eagle fight from Ubisoft

- Avoid repeated patterns and sharp lines like stripes, blocks, stairs, threes, poles. Moving close to such patterns create a considerable amount of motion cues.
- Limit flickering and other artefacts that could generate sharp, high contrast, visual patterns that (like stripes) create a considerable amount of motion cues. For the same reason blurred images (and artificial blurring algorithms) could be used to reduce vection.
- The addition of static frames or elements around the view of the moving camera (like a view from inside a vehicle or a fake nose) could partially reduce VR sickness probably due to a combination of perceptual effect (reduction of the vection due to static elements) and cognitive activity (identification of an familiar situation that could help to interpret the incongruous motion cues)
- If the viewer is stationary and the scene moves independently from the viewer there is a perceived disconnect between the viewer's visual system and motion sensing system. This can lead to disorientation and in some cases nausea.
- If motion is to be used as a creative intention then care should be taken. The following techniques can be useful:
  - Placing the viewer in a recognizable situation i.e. inside a vehicle or a flying bird's eye view.
  - If the motion is at ground level, avoid:
    - Accelerating too rapidly
    - Turns that are not relevant to the action
    - Yaw and pitch of the horizon
    - Vertical motion – low frequency - as in replication of human walking POV
  - Where the action follows a relevant person or object, this can reduce motion effects.

4.1.2.4 Image capture rate and motion capture fidelity

Content should be shot or created in a manner that reduces motion blur or motion stepping to comfortable levels as flickering images can lead to rapid fatigue or disorientation.

Image capture at higher frame rates and designing content to work within the target display capabilities is recommended to reduce discomfort.

When capturing moderately fast-paced action such as dance, sport and vehicles a minimum capture frame rate of 50/60fps should be considered.

- If shooting at higher frame rates, consideration of the target display(s) characteristics should be made.

4.1.2.5 Orientation

For 360° content, it is best practice to keep the horizon level at all times unless angled for a specific purpose. The viewer will always tend to match their head inclination to the perceived horizon – this can lead to discomfort and vertigo.

If the horizon oscillates this can prove to be very uncomfortable.

Panning and tilting the camera can also be very uncomfortable.

When transitioning between shots, care should be taken to orientate the action or intended direction of view of the subsequent shot to the outgoing shot. Significant changes in the scene orientation across cuts can cause the viewer to become lost in the experience and may result in physical discomfort while attempting to regain the area of action they were previously following.

4.1.2.6 Perceived Viewer eye height

Camera to scene height should always be relevant to the scene.

A small discrepancy in height can result in surprisingly off-putting perceptions of the relative sizes of subjects.

Ideally the camera height should be coherent with the final user position (seated vs standing up). For 3DOF systems, this could imply the user to select his viewing position (seated vs standing up)[2].

The wide-angle distortion of some camera formats can exacerbate this, a low camera height can make subjects appear unusually tall – similarly a high camera height can make subjects appear unusually small.

4.1.2.7 Proximity of objects in the scene

Objects close to camera have a disproportionate influence on the viewer and can unnecessarily dominate the scene. This effect can impact reaction in attempt to move away from the object.

Wide-angle perspective distortion of some camera formats may make objects approaching the camera appear to accelerate towards the viewer and so should be avoided.

If only 3DOF is supported, nearby objects create occlusion and the apparent loss of 3D effect due to inability to look around the object may break the illusion of immersion. In other words it makes it very clear to the user that there is no 6DOF and this might be disconcerting.

4.1.2.8 Duration of Content

Results from a survey (of 300 respondents) indicate that:

- Shorter content durations are more comfortable to the user
- Up to 20 minute durations are seen as acceptable if the content is not challenging.
- Content with significant action or motion should be shorter.

4.1.2.9 3D Stereo Content

The impression of depth obtained using stereoscopic contents is the consequence of the binocular disparity (the difference in the retinal projection of the 2 eyes) induced by the human inter pupillary distance IPD (on average 6 cm for the adults). The further the objects are from the user the smaller the difference of the retinal projection. In the case of objects that are more than 20 meters from the user, the binocular disparity plays a minor role in the perception of depth.

The vision of stereoscopic content in current HMDs[3] induces a visual fatigue due to a phenomenon called the “vergence/accommodation conflict”. In fact, all the content presented in the HMD are at the same focal plane that is dictated by the lens (usually at 1.5 meters) while the binocular disparity imposes a vergence distance that varies as a function of the “stereoscopic distance (i.e. the horizontal disparity between the 2 retinal projections of a target)” of the object to keep in the line of sight (in the foveal area). The larger the difference between the focal and the vergence distance, the greater will be the fatigue.

[2] 6DOF systems (like the HTC vive) can adapt (at least in part) the video to be coherent with the user head height.

[3] Lightfield displays are supposed to limit this issue.

The creation of stereoscopic content implies some specific Human Factor considerations:

- The long term effects of repeated exposure to stereoscopic content on the development of the human visual systems are still under debate. As of today no longitudinal studies have been conducted to evaluate such effects and the few statements produced by medical and governmental organizations are in part contradictory. The "French Agency for Food, Environmental and Occupational Health & Safety" for instance state that: "children under the age of 6 should not be exposed to 3D technologies" and "children under the age of 13 should only use 3D technologies in moderation, and that both they and their parents should be vigilant concerning any resulting symptoms" [3DTECH] while the "American Optometric Association" state that "By the age of 3 years most children will have binocular vision well enough established to enjoy viewing 3D television, movies or games" [AOA-FAQ].
- Differences in the luminosity, sharpness or colors of two images presented in a stereoscopic way will make it harder for the two images to be perceptually fused, with the most vivid image being predominant. Any lack of fusion may cause additional fatigue for the viewer.
- Stereoscopic depth information should be coherent with the other depth information. In particular, it is important to avoid that objects that appear in front of the "screen" (negative parallax) will move out from the HMD field of view. This problem known in literature as the "frame problem" induces visual discomfort due to the conflict between the stereoscopic depth cues that suggests that the object is close and the monocular (occlusion) information that suggests that the object is far.

One of the most powerful cues which enables the perception of immersion is that of stereo imagery in the experience.

- The experience should have carefully planned and implemented stereo depth regime.
- Care should be taken to avoid too much inappropriate stereo disparity (uncomfortable stereo depth) as this will cause eye-strain and visual discomfort.
- Near-field objects should not be positioned or manipulated so as to cause adverse autonomous defensive reactions in the consumer.

4.1.2.9.1 Additional Health and Safety Issues

Most of Virtual Reality HMD producers include in their "health and safety warning" some statements concerning the risks of use of these devices by children. Examples of such warning are presented in Table 1.

Table 1: Age related health and safety warnings

Device	Statement	Source
Playstation VR	The VR headset is not for use by children under age **12**.	https://www.playstation.com/en-us/network/legal/health-warnings/
Oculus Gear	The Gear VR should not be used by children under the age of **13**, as young children are in a critical period in visual development.	https://scontent-cdg2-1.xx.fbcdn.net/v/t39.2365-6/17640357_1698999383446748_1803373359325511680_n.pdf?oh=10d89b617e1c5f894bd8117bc3470ed8&oe=5A82F8CB

Oculus Rift	This product should not be used by children under the age of **13**, as the headset is not sized for children and improper sizing can lead to discomfort or health effects, and younger children are in a critical period in visual development	https://scontent-cdg2-1.xx.fbcdn.net/v/t39.2365-6/19896829_771660013013643_4087250127671001088_n.pdf?oh=642cb259720ceb661a181142929621d1&oe=5A8836EA
Google Daydream	Daydream View should not be used by children under the age of **13**.	https://support.google.com/daydream/answer/7185037?hl=uk
HTC vive	The product was not designed to be used by children. Do not leave the product within the reach of young children or allow them to use or play with it. They could hurt themselves or others, or could accidentally damage the product.	http://dl4.htc.com/vive/safty_guide/91H02887-08M%20Rev.A.PDF?_ga=2.37961125.787580076.1508146654-1752841129.1507818191

4.1.2.10 Subtitles

Content can be produced with (non-positional) subtitles. These will be rendered by the HMDs with formats and positioned as selected by the end-user.

4.1.2.11 Video Live/Post Production

- Production should take into consideration the aspects described in sections 4.1.2.1 to 4.1.2.9
- Cuts should be consistent with the action and story.
  - Consideration of the viewer motion
  - Use of production techniques such as emphasized lighting or sound to direct the viewpoint
  - Consistent story telling
  - Audio alignment with image content
  - Spatial alignment and temporal synchronization.

4.1.2.12 Audio for VR - overview

One of the essential novelties of VR is that a user is free to change their viewing gaze at will, allowing individual immersive experiences in any viewing direction at any given moment. Consequently, the methods of audio creation, transmission and reproduction applied for VR must be able to accompany the dynamically changing visual perspective.

This means that audio must be reproduced equally well in all directions, allowing the presentation of sounds from below or above the viewer with the same spatial accuracy as sounds from the front.

Further, for a realistic listening experience the audio presentation method must seamlessly adapt the spatial audio processing and recreate the sound scene coherently with respect to the dynamically changing viewer's gaze. Spatial audio is essential for compelling VR experiences. It can be used as a tool to immerse and guide the viewer in the VR scene.

Even when the video content has a limited FOV, (e.g. 180° or 270° content), full spherical audio is a requirement – since an artificial 'silence' from the back, in these cases, will result in the loss of immersiveness and/or the lack of suspension of reality.

4.1.2.13 Audio Formats

Audio for VR can be produced using three different formats. These are broadly known as channels-, objects- and scene-based audio formats. Audio for VR can use any one of these formats or a hybrid of these (where all three formats are used to represent the spherical soundfield.

Channels

Loudspeaker-based audio reproduction such as stereo 2.0 or surround 5.1 has been the de-facto standard for production and audio delivery to consumers for decades. To ensure the intended sound reproduction, channel-based audio requires the same standardized loudspeaker placement at the production facility and the listener's reproduction location. Standardized loudspeaker configurations include simple mono and stereo, horizontal-only (5.1) to immersive 7.1+4H and 22.2.

For faithful reproduction over headphones, a common methodology to use virtual loudspeakers and the corresponding set of HRTF/BRIRs that is relevant for a certain head position of the listener relative to the loudspeaker positions. When the head rotates, the soundfield is rotated by updating the set of HRTF/BRIR. This approach requires an accurate and high-resolution set of HRTF/BRIRs available for all possible head locations relative to the loudspeaker positions as well as careful spatio-temporal interpolation when updating the HRTF/BRIRs in real time.

Objects

For Object-based audio the sound scene is composed of multiple individual sound sources (objects) along with metadata that describes its spatial characteristics (position, width, radiation pattern, room-reflection properties, etc). During playback, the audio scene is constructed or rendered using all the audio sources and the associated metadata. The format is thus agnostic of loudspeaker positions. For loudspeaker playback, the renderer considers the number and position of loudspeakers – and in the case of headphone playback such as VR – renders to headphones.

The use of object-based audio is quite effective for post-produced audio. It involves the use of DAWs to create the audio scene in which some objects might get grouped into a summary element or stem while others such as Dialog can be carried as a discrete object.

For live capture and distribution of object-based audio, all audio objects must be tagged with the correct metadata (e.g. location, diegetic, diffusion, width, etc). Dynamic audion objects requite time-varying metadata. Real-time tracking

For post-production, an audio engineer typically pans and adjusts the objects in a 3D scene according to the video scene.

A pure object-based representation can require a multitude of individual audio tracks and their associated time-varying metadata. Typical cinematic content involves the use of several simultaneous objects. The bandwidth necessary for transmitting a sound scene depends on the number of simultaneous objects present at any point in time. Due to limitations in bandwidth for streaming or broadcast services, the high number of objects used in cinematic content typically must be reduced through object grouping (mixing).

Objects can be either individually binauralized using one discrete HRTF convolution process per object, or rendered (e.g., using Vector Base Amplitude Panning [Pulkki]) to a set of virtual loudspeakers which is then binauralized using one HRTF convolution process per virtual loudspeaker (as discussed in the channels section). While the first method results in the best possible rendering quality, the complexity increases

significantly with the number of objects. Alternative techniques trade off quality with complexity in different ways.

Scene-based Audio

Scene-based audio (SBA) represents the acoustic pressure field as a function of space and time using a set of coefficients that are the linear weights (or coefficients) of orthogonal spatial basis functions known as Spherical Harmonics. This is also known as Higher Order Ambisonics (HOA). Like Object-based audio, SBA is agnostic to the loudspeaker configuration. For loudspeaker playback, the renderer adjusts to the number and position of loudspeakers.

First Order Ambisonics (FOA) (also commonly referred to as B format) is a basic form of scene-based audio in which the soundfield is described by only the lowest four spherical harmonic coefficients. Higher Order Ambisonics provides a more accurate sound field representation by using additional spherical harmonic coefficients beyond the lowest four. As the number of spherical harmonic coefficients increases, so does the accuracy of the spatial audio representation. The number of coefficients for full 3D is $(N+1)^2$, where N is the ambisonics order.

SBA provides an efficient and accurate representation of the sound field with a limited number of coefficients. The accuracy of the soundfield is dependent only on the Ambisonics order. Moreover, for a given Ambisonics order, the bandwidth is not a function of the number of sound sources in the scene.

The spherical harmonic based representation enables a matrix based sound field rotation that is efficient and smooth, allowing for compelling VR experiences. The rotation operation does not increase complexity as it is typically integrated in the rendering operation.

Depending on the implementation technique, the computational complexity of the rendering increases with the HOA order. There are algorithms for efficient binauralization of HOA coefficients that are also independent of the complexity of the scene and the number of virtual speakers used in rendering. These computational advantages are invaluable in enabling head-tracked binauralization for VR on consumer devices.

Capturing/acquiring Ambisonics sound fields can be achieved using compact microphone arrays.

A single scene-based audio representation can also represent any number of discrete audio elements without requiring additional metadata or increased bandwidth. In this approach, the audio scene is constructed or augmented by panning pre-recorded or live captured audio elements into the 3D space. The same benefits and limitations of Object-based audio panning, described above, applies here.

Hybrid formats

Hybrid scenes that involve the use multiple formats can be used for VR audio. Example scenarios include the combination of audio channel-beds and objects as well as the use of Ambisonics audio along with discrete audio objects (such as commentaries) and relevant metadata.

4.1.2.14 Audio Recording and Production

- Spatial audio from live scenes can be recorded using a range of microphones distributed in and around the recording scene.
- There are a number of microphone arrays that are available off the shelf that allow the capture of the spherical soundfield in HOA format. For 3DOF content, the positions of these microphones are considered to be relative to the camera position – which is generally considered to be at the origin.
- For 3DOF content, soundfield captured through microphones should be rotated in accordance with camera rotation.

- If the camera is to be translated (not just rotated), then attaching soundfield microphones to the camera is recommended.
- Microphones should be positioned with the following considerations:
  - Sound captured through stationary contact microphone modules provide a correct sampling of the audio at the position of the microphone. As such, it is relatively simple to 'pan' that recorded audio into 3D space by considering the position of the stationary microphone. This can be done through DAW software.
  - Sound captured through shotgun or parabolic microphones are not necessarily representative of the audio at the position of the microphones. To 'pan' that recorded audio into 3D space requires knowledge of the position of the audio object that the microphones are being directed at. If these positions can be estimated (either manually or automatically), the panning into 3D space can be achieved. The same is true for lapel microphones placed on a moving acoustic source. It is essential to ensure that a high degree spatial accuracy is maintained – such that the positional cues from the visual and acoustic scene are not contradictory.
- High spatial resolution can be achieved using audio channels, objects with metadata or Ambisonics audio or a combination thereof.
- Higher Order Ambisonics signals should be recorded and produced in at least full-spherical 4th order HOA. The higher the Ambisonics order, the higher the spatial resolution allowing the localization and tracking of acoustic features – a necessary condition for immersive experiences.
- Recordings from microphone-arrays can be combined with individually recorded and/or produced stems either live or during post-production.
- For Object- and Channel-based production, unintentional audio crosstalk between (spot) microphones (e.g., due to proximity or reverberation) should be avoided.
- For a live and/or post production, the audio goes through a console or Digital Audio Workstation (DAW) that further processes the spatial characteristics of the sound field. This is on top of the basic audio production workflow. Example DAW software that allows the live capture as well as post-production of spatial audio include [DAW1].
- Consideration during production should account for the fact that spatial audio will be reproduced from all directions, including the lower hemisphere.
- The sampling-rate during recording should be at least 48 kHz.
- Diegetic/non-Diegetic Audio
  - Audio elements having a visual counterpart in the scene (Diegetic audio elements) must be spatially aligned and temporally synchronized with the video during head-motion.
  - Audio elements that do not have a visual counterpart in the scene (Non-Diegetic) may be produced so that their position is not compensated with the user's head-motion.

Note: An audio element may alternate over time between having a visual counterpart and not having a visual counterpart. An example is a narrator that is initially not present in the visual scene and then becomes part of the visual scene. Compensation for the user's head motion should only happen when the audio element has a visual counterpart.

- The audio export of the production should be loudness-normalized for consistent loudness across different content.
- Metadata for loudness of audio content should follow the appropriate regional recommendations for broadcast content delivery and exchange. Some examples of the recommendations include EBU R128 [R128] and the US CALM act. VR production requirements are subtly different in that it is

not known which direction and position of the listener is not known at production time, meaning that these recommendations, while useful, may not have the same amount of benefit as traditional TV viewing.

- Presentation/Rendering for mixing and monitoring
  - Consideration during production should account for the fact that audio will be primarily experienced with headphones or in-ear buds.
  - Mixing using headphones should support low latency soundfield rotation with head tracking. A motion to sound latency lower than 30ms is recommended.
    - When mixing over headphones, an optimal binaural audio experience is achieved when the headphone feeds are created using personalized Head Related Transfer Functions (HRTFs) and headphone equalization. It is recommended that HRTFs closely matching those of the mixing engineer be used.
    - If the headphone feeds are created by an intermediate virtual loudspeaker renderer (before HRTF processing, for example), it is recommended that the renderer be indicated using appropriate metadata. Audio emission encoders (such as MPEG-H) often have the option of transmitting the renderer through the emission bitstream. This will allow for the use of the same renderer that was used for mixing - when played to consumers.
  - If mixing is done over a loudspeaker array, it is recommended that the renderer be indicated using appropriate metadata. Audio emission encoders (such as MPEG-H) often have the option of transmitting the renderer through the emission bitstream. This will allow for the use of the same renderer that was used for mixing - when played to consumers.

4.1.2.15 Interactivity and visual exploration

Compared to traditional media where the content presented in the display is mostly independent from user actions, content presented using VR devices react (at least in part) to user actions. At a minimal level (3DOF 360° videos for instance) the images displayed on the HMD are a function of the movement of the head of the user - which decides which part of the 360° environment to visually explore.

- This interactivity has strong implication for the creation of the content to grant an enjoyable and comfortable user experience.

4.1.3 Master Format

4.1.3.1 Projection and Aspect Ratio

The Master Video presentation format should have the following characteristics:

- Projection will be equirectangular projection.
- Co-ordinate system as described in [OMAFFDIS]
- The video may be generated from multiple camera arrays or composited imagery.
- The image should display no apparent stitch lines and occluded or missing picture information.
- Frame motion should not display motion artefacts such as blur or step motion.
- For fully 360° video, 2:1 Aspect Ratio with the following attributes:
  - No Zenith or Nadir Blind-spots are permitted.
- For partial 360° video
  - The metadata element ***Coverage*** as defined in Table 9 describing the partial region must be included.

Figure 1: Example *Coverage* metadata for Partial Panorama

4.1.3.2 Video Master Format

A Master VR Video file from which transcodes for the various end-user platforms is desirable.

4.1.3.2.1 Resolution

- Stereoscopic format: Separate Left/Right eye files.
- Metadata as depicted in Table 9.

For full 360° video

- Monoscopic: minimum 4096 H × 2048 V
- Stereoscopic: minimum 4096 H × 2048 V for each eye

For partial 360° video

- 1:1 pixel ratio
- Horizontal minimum: ((***Coverage.AzimuthMax*** – ***Coverage.AzimuthMin***) / 360) × 4096
- Vertical minimum: ((***Coverage.ElevationMax*** – ***Coverage.ElevationMin***) / 180) × 2048

4.1.3.2.2 Video Metadata

Video Metadata to be included with video content is depicted in Table 9.

4.1.3.2.3 Frame rates

All media should be acquired and processed as Progressive frames. Acceptable progressive frame rates for monoscopic video are:

- 25 – Subject to content motion constraints
- 30 – Subject to content motion constraints
- 50
- 60

Acceptable Progressive Frame Rates for stereoscopic video are:

- 60
- 75
- 90

- 100
- 120

Comfortable frame rates for monoscopic is 50 and for stereoscopic is 100 [FRATES]. A higher frame rate in the case of stereoscopic content is recommended to make the process of binocular fusion more comfortable and reduce visual fatigue.

4.1.3.2.4 File formats

Table 2: Master File formats

Bit Depth	10-bit
Color Sampling	4:2:2
Color Space	ITU Rec.709 (gamut levels within the threshold defined by EBU R103)
Scan	Progressive Frame
Delivery Format – Option 1	MXF Program Contribution (AMWA AS-11X1 as per DPP specification)
Delivery Format – Option 2	IMF Application 2e / JPEG2000 minimum data rate 150Mb/s
Delivery Format – Option 3	DnxHRHQ
Delivery Format – Option 4	ProRes422HQ

4.1.3.3 Audio Master Format

4.1.3.3.1 File Format

An open production format such as the Audio Definition Model (ITU-R BS.2076-1) facilitates content exchange for contribution [ADM] should be used.

4.1.3.3.2 Audio Format

- For production in Scene-based Audio, full-spherical 4th order Ambisonics with or without additional audio objects for scene augmentation is recommended. The higher the Ambisonics order, the higher the spatial resolution and allows the better utilization of the capabilities of todays and future distribution platforms.
- The contribution should be in single tracks or in an interleaved track format with ACN ordering of the ambisonics signals [ACN]
- The normalization of the ambisonics signals should be N3D, or SN3D [NORMS]. N3D normalization is recommended for 32 bit resolution PCM floating point file formats. If legacy 16 bit PCM fixed-point file formats are used, SN3D normalization is recommended.

4.1.3.3.3 Bit depth and sampling rate

- A PCM file format supporting 32 bit floating point (e.g. ITU-R BS.2088 [BS2088]) is recommended. The minimum bit depth is 16 bits.
- The minimum sampling rate should be 48 kHz.

4.2 Media Profiles

Since early 2016, MPEG has worked on a project known as Omnidirectional MediA Format (OMAF) which reached the Final Draft International Standard (FDIS) stage in October 2017 and is expected to be published as ISO/IEC 23090-2 in 2018. OMAF includes two ways of representing an omnidirectional scene in video pictures: a classical "equirectangular" projection like what has been used historically for maps of the globe, and a mapping of the scene onto the faces of a cube. It supports signaling of the metadata required for interoperable rendering of 360° monoscopic and stereoscopic audio-visual data, and provides a selection of audio-visual encoding formats for this application. It also includes technologies to arrange video pixel data in numerous ways to improve compression efficiency and reduce the size of video, a major bottleneck for VR applications and services.

4.2.1 Introduction

In the following sections, the VR-IF media profiles for video and audio are presented. These media profiles are aimed to provide interoperability points for media codecs and associated metadata as well as media coding and encapsulation configurations that may be used for rendering, compression, streaming, and playback of the omnidirectional media content.

4.2.2 Selected Media Profiles

4.2.2.1 Video

4.2.2.1.1 Overview

This section describes the selected media profiles for video, namely:

1. HEVC-based viewport-independent OMAF video profile, further described in section 4.2.2.1.2
2. HEVC-based viewport-dependent OMAF video profile, further described in section 4.2.2.1.3

For the viewport-independent baseline media profile, the maximum achievable viewport resolution is constraint by the video decoder capabilities, specified by the elementary stream constraints in the video profiles with HEVC Main 10 Profile, Main Tier, Level 5.1. The following table shows the maximum viewport resolutions that can be achieved with the viewport independent baseline media profile for HEVC Main 10 Profile, Main Tier, Level 5.1, 60fps, for a display with an FOV of 90°×90° and a given content coverage.

Content coverage	Maximum viewport resolution
360°×180°	1K×1K
270°×180°	1.2K×1.2K
180°×180°	1.4K×1.4K
180°×120°	1.8K×1.8K

Table 3: Maximum achievable resolution in the viewport using viewport-independent baseline media profile for HEVC Main 10 Profile, Main Tier, Level 5.1, 60fps, for a display with a FOV of 90°×90° and a given content coverage

The HEVC-based viewport-dependent OMAF video profile allows for achieving higher resolutions of the viewport compared to the entries of Table 3 given the same capabilities (HEVC Main 10 Profile, Main Tier,

Level 5.1, 60fps, for a display with an FOV of 90°×90°). This guidelines describe how to generate content for this profile by mixing low and high-resolution tiles and thereby better leveraging the Max luma picture size as defined by the HEVC video profile and level definitions.

In the download and streaming case, the HEVC-based viewport-dependent OMAF video profile can be used if resolutions higher than those achievable by the viewport independent baseline media profile are desired. In the streaming case, the HEVC-based viewport-dependent OMAF video profile can additionally be used to achieve bandwidth savings. However, viewport dependent streaming comes with additional latency requirements, as described in detail in section 5.4.3, that consist of sensor latency, network request delay, origin-to-edge delay (in case of cache miss), transmission delay (accounting for access network delay) and delays incurred in the client device due to buffering, decoding and rendering. These factors, might affect the time needed from the time instant the head movement happens until high quality/resolution content is shown to the user, when viewport dependent baseline media profile is used.

As described in section 5.4.3.1 in more detail, the download use case is considered to be very attractive for VR services. In such a case, on-demand VR content may be included in a single ISOBMFF file which is downloaded before playback. The structure of that file changes depending on the media profile used. In case of HEVC-based viewport-independent OMAF video profile the entire VR content is included in the file as one track. In this case the maximum viewport resolution is limited depending on the content coverage as described in the example above.

In the download case the file size using HEVC-based viewport-dependent OMAF video profile is expected to increase, since the Viewport-Dependent profile emphasizes the quality of a certain region at a time and multiple video bitstreams (or tiles) shall be made available as separate tracks in the same file.

In case of Viewport-Dependent baseline media profile without tiles, the downloaded file contains a number of tracks corresponding to the number of independently encoded viewports, each in full resolution (e.g. 4k resolution per viewport). Such provision of downloadable data may lead to significantly higher file sizes.

If the HEVC-based viewport-dependent OMAF video profile is used with tiles, one ‘hvc1’ track per tile and per resolution is included in the ISO BMFF file. In addition, one ‘hvc2’ track per potential viewing direction is included in the same ISO BMFF file. Therefore, the file contains a number of extractor tracks (viewports) corresponding to the number of viewports and a number of ‘hvc1’ tracks corresponding to the number of encoded tiles. The number of the tracks corresponding to tiles depends on the tiling granularity and the number of different representations (varying in qualities or resolutions) of each of the tiles.

4.2.2.1.2 HEVC-based viewport-independent OMAF video profile

4.2.2.1.2.1 Introduction

This media profile is defined in ISO/IEC 23090-2 [OMAFFDIS] section 10.1.2 and fulfils basic requirements to support omnidirectional video. Both monoscopic and stereoscopic spherical videos up to 360°are supported. The profile does neither require viewport dependent decoding nor viewpoint dependent delivery. Regular HEVC encoders, DASH packagers, DASH clients, encryption technologies, file format parsers and HEVC decoder engines can be used for encoding, distribution and decoding. The profile also minimizes options to support basic interoperability.

This media profile is expected to be supported by HMDs and other devices rendering omnidirectional video powered by platforms released in 2015 and onwards. The key support is the availability of HEVC Main 10 Level 5.1 decoders to process 4k by 2k frames at frame rates up to 60 fps as well as GPU-based rendering. The profile permits improved immersive experiences beyond the basic capabilities.

Figure 2 provides an overview of a possible receiver architecture that recovers the spherical video. Note that this figure does not represent an actual implementation, but a logical set of receiver functions. More implementation aspects are covered later. Based on an OMAF media stream, the receiver parses, possibly decrypts and moves the elementary stream to the HEVC decoder. Either the OMAF Metadata as defined in [OMAFFDIS] or the Decoder Rendering Metadata (SEI messages) may be used by the Texture-to-Sphere Mapping function to generate a spherical video based on the decoded output signal, also known as "texture". The viewport is then generated from the spherical video signal by taking into account viewport position information from sensors, display characteristics as well as possibly other metadata such as initial viewport information. Whereas the decryption and decoding is typically done in hardware on devices, the OMAF restricted scheme permits to use existing texture mapping and rendering functionalities on GPUs to generate the viewport.

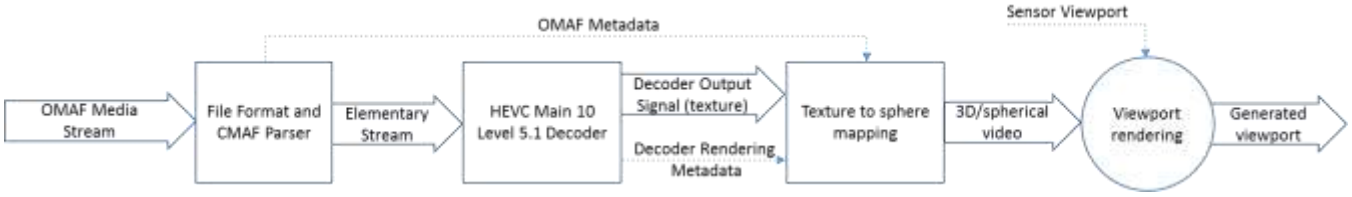

Figure 2: Receiver Model for HEVC-based viewport-independent OMAF video profile

The projection in this profile is restricted exclusively to EquiRectangular Projection (ERP), but permits delivery of less than full 360° spherical video.

The texture signal is restricted to 4096 samples in horizontal and vertical direction.

Details on definitions and coordinate systems can be found in [ADDSEI]. Note that the following terms are used in the remainder of this document following the definitions in this document:

- **coverage sphere region**: *sphere region* that is covered by a *cropped decoded picture*.
- **global coordinate axes**: coordinate axes associated with *omnidirectional video* that are associated with an externally referenceable position and orientation.
- **local coordinate axes**: coordinate axes having a specified rotation relationship relative to the *global coordinate axes*.
- **omnidirectional video**: video content in a format that enables rendering according to the user's viewing orientation, e.g., if viewed using a head-mounted device, or according to a user's desired *viewport*, reflecting a potentially rotated viewing position
- **packed region**: region in a *region-wise packed picture* that is mapped to a *projected region* according to a *region-wise packing*
- **projected picture**: picture that uses a *projection format* for *omnidirectional video*.
- **projected region**: region in a *projected picture* that is mapped to a *packed region* according to a *region-wise packing*.
- **projection**: specified correspondence between the color samples of a *projected picture* and azimuth and elevation positions on a sphere.
- **region-wise packed picture**: decoded picture that contains one or more *packed regions*.
- **region-wise packing**: transformation, resizing, and relocation of *packed regions* of a *region-wise packed picture* to remap the *packed regions* to *projected region*s of a *projected picture*.
- **sphere coordinates**: azimuth and elevation angles identifying a location of a point on a sphere.

- **sphere region**: region on a sphere, specified either by four *great circles* or by two *azimuth circles* and two *elevation circles*, or such a region on a rotated sphere after applying yaw, pitch, and roll rotations.
- **viewport**: region of *omnidirectional video* content suitable for display and viewing by the user.

4.2.2.1.2.2 External Specification

Video elementary streams are encoded following the requirements in ISO/IEC 23090-2 [OMAFFDIS] section 10.1.2.2. In particular, SEI messages describing the omnidirectional video as defined in [ADDSEI] need to be present.

ISO BMFF Tracks are encoded following the requirements in ISO/IEC 23090-2 [OMAFFDIS] section 10.1.2.3 and 10.1.2.4. ISO BMFF files that contain such an encoded track are identified by the brand 'hevi'. The OMAF metadata is equivalent to the information that is present in the SEI messages for omnidirectional video as specified in ISO/IEC 23090-2 [OMAFFDIS] section 10.1.2.3.

DASH Integration is provided following the requirements and recommendations in ISO/IEC 23090-2 section 10.1.2.6. An Adaptation Set including Representations formatted according to this media profile is recommended to be signaled as

- @codecs='resv.podv+erpv.hvc1.1.6.L93.B0'
- @mimeType='video/mp4 profiles="hevi"'
- A Supplemental Descriptor or Essential Descriptor providing the frame packing arrangement may be used.

Note: By the use of the restricted video scheme and the @profiles referring to this media profile, the DASH client has all information to identify if this media profile could be played back. For additional information, the Supplemental Descriptor is used to provide some details on the configuration of the contained Representations.

4.2.2.1.2.3 Quality and Performance

3GPP TR26.918 [TR26918] contains subjective test experiments that investigate the dependency of perceived visual quality on spatial resolution of the omnidirectional video. According to the results reported in 3GPP TR26.918 [TR26918], perceived quality increases with the video resolution. On the tested equipment, 3GPP TR26.918 [TR26918], 4K spatial resolution with ERP provides good quality.

While the tests in 3GPP are limited and not comprehensive, they are the only ones available up to date that can be publicly accessed. It is relevant to understand that no general conclusions can be drawn from the tests. Readers are encouraged to look at the detailed test setup and results before drawing any conclusions.

According to the TR, the coded video bitrate is expected to be in the range of 5-20 Mbit/s, depending on the content and good quality being achieved in the upper range of the bit rate range.

4.2.2.1.3 HEVC-based viewport-dependent OMAF video profile

4.2.2.1.3.1 Introduction

This media profile fulfils basic and advanced requirements to support omnidirectional video. Both, monoscopic and stereoscopic spherical video up to 360° is supported. This profile supports quality emphasis on the actual user viewport, which allows higher resolution in the viewport and/or reduced bitrate compared to the HEVC-based viewport-independent OMAF video profile.

This profile allows streams to have a different quality or resolution for different areas/regions of the omnidirectional video each of them corresponding to a preferred viewport (the one to which the

area(s)/region(s) with highest quality correspond to). When using this profile, different options are available:

1. Areas are encoded with higher or lower quality/fidelity: For example, the quantization step of transform coefficients is adapted spatially in such a way that the visual quality for the regions differ
2. Areas are encoded with different resolution: some areas are downscaled from their original resolution to a lower one.

For each of the two options listed above, there are two possible instantiations:

1. All areas of the omnidirectional video are offered in a single stream/track/Representation
2. Each of the areas of the omnidirectional video is offered using separate stream/track/Representation

Note: Even for the latter case where each of the areas is offered as separate stream/track/Representation all areas corresponding to the whole omnidirectional video can be consumed using a single video decoder conforming to HEVC Main 10 Profile, Main tier, Level 5.1.

Regardless of which of the above-described instantiations is used, the following metadata is applicable:

1. Region-wise quality ranking [OMAFFDIS], enabling to indicate a relative quality order of regions. This metadata is applicable for both the single-resolution and multi-resolution viewport-dependent content.

2. Region-wise packing [OMAFFDIS], which provides a region-wise mapping between packed pictures and projected pictures. This metadata is typically unnecessary for single-resolution viewport-dependent content and needed when regions are coded with different resolutions.

In the configuration provided within the guidelines, only areas with different resolutions and offered using separate streams/tracks/Representations are documented. This is achieved by using HEVC tiles corresponding to different resolutions. These tiles of different resolutions can be combined to cover the entire omnidirectional video, i.e. the whole omnidirectional video is decoded using a single video decoder conforming to HEVC Main 10 Profile, Main tier, Level 5.1 with a resolution/quality emphasis on the actual user viewport. This allows higher resolutions to be displayed in the viewport as compared to the viewport independent media profile.

With the configuration provided within these guidelines, when using this profile one or more HEVC streams of the omnidirectional video are offered at different qualities/resolutions and are encoded comprising HEVC tiles that are encoded as Motion-Constrained Tile Sets (MTCS), i.e. tiles are encoded in such a way that does not reference other tiles. In addition, this profile includes ISO Base Media File Format (ISO BMFF) tools that allow for generating a single HEVC stream that can be decoded by a single HEVC Main 10 Profile, Main tier, Level 5.1 capable decoder and presented.

Minimum receiver capabilities required to support this profile:

- HEVC Main 10 Profile, Main tier, Level 5.1
- ISO BMFF extractors, as defined in ISO/IEC 14496-15 [NAL] specification
- Region Wise Packing, as defined in ISO/IEC 23090-2 [OMAFFDIS] specification

Note: The HEVC-based viewport-dependent OMAF video profile allows considerable freedom in the usage of Region Wise Packing with up to 256 separate regions. Likewise, the HEVC specification allows up to 110 tiles in Level 5.1 of HEVC Main 10 profile. This might lead to undesirable implementation complexity.

4.2.2.1.3.2 OMAF-DASH Viewport-Dependent Streaming and Download Client model

This section provides an overview of the OMAF-DASH streaming client model as well as of the OMAF Download client model and briefly describes their components. A detailed description of these components and the corresponding interfaces can be found in section 5.7.3.

Figure 3 shows a high-level structure of the OMAF-DASH client model with interfaces for streaming. It consists of 5 sub modules and illustrates the interfaces between them.

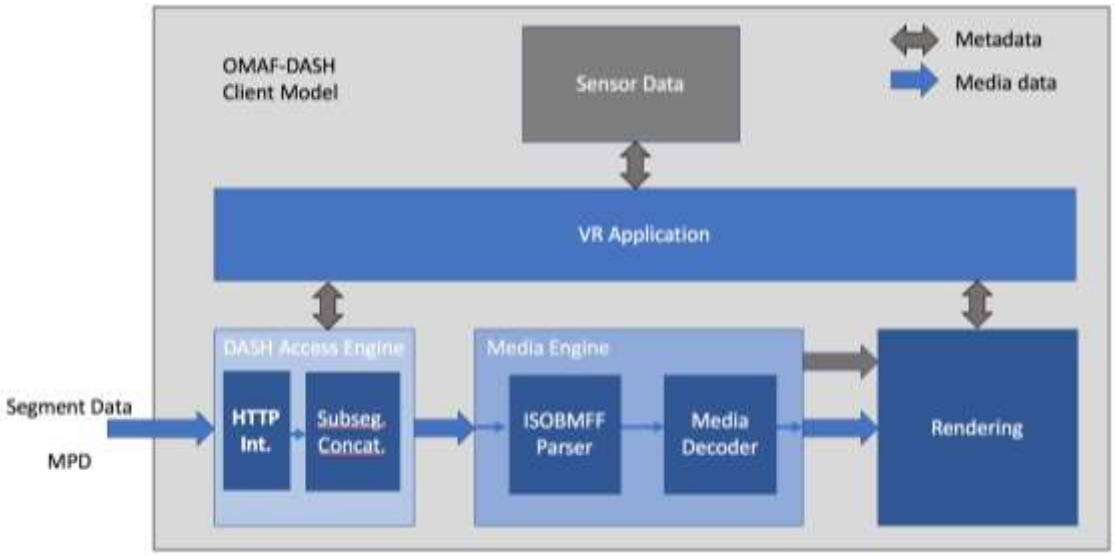

Figure 3: OMAF-DASH Streaming Client model with interfaces

DASH Access Engine: Downloads all OMAF Media streams and generates a single ISO BMFF file by concatenating subsegments as indicated in the figure.

VR application: Determines which OMAF Media streams should be downloaded by the DASH Access Engine. Controls the rendering depending on sensor data and HMD capabilities.

Media Engine: Plays the ISO BMFF file (i.e. plays the extractor track) and outputs a single NAL unit video stream into the Media Decoder that outputs decoded pictures and metadata into to the Renderer.

Renderer: Renders the decoded video pictures to the HMD display.

Sensor Data: User metadata (e.g. viewport position, direction, speed, etc.) taken e.g. from the HMD.

Note: Generally, it is preferable to use hardware supported functionalities to optimize speed, latency, power consumptions and overall performance. Each of those above functions may be accessed with APIs. Specific APIs, possibly supported on SDKs and media frameworks are currently under development for example in Khronos or W3C.

A detailed description of each sub-module and its interfaces is given in section 5.7.3.

Figure 4 shows a high-level structure of the OMAF-Download client model with its interfaces.

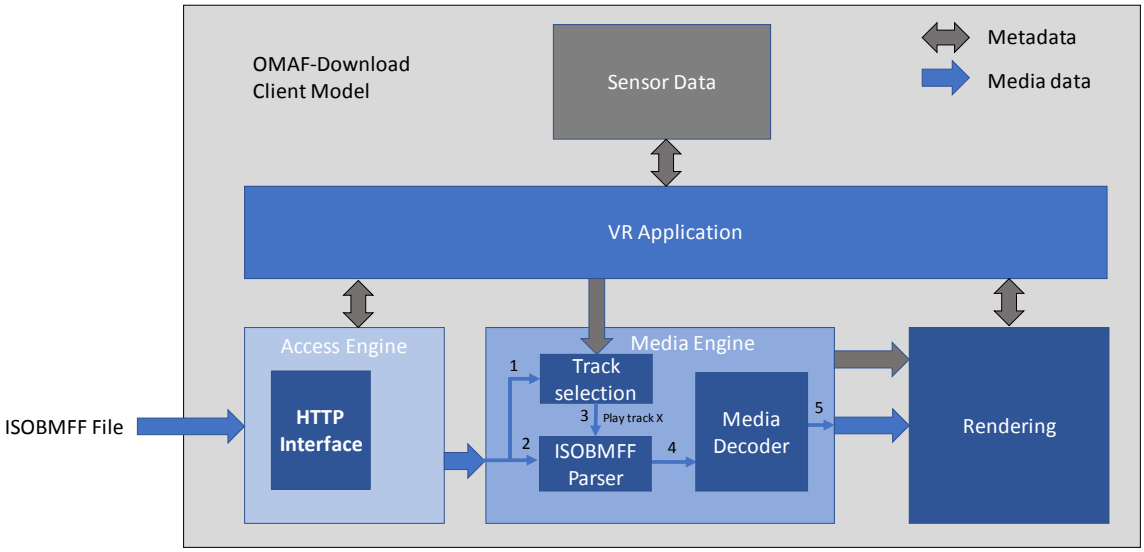

Figure 4: OMAF-Download Client model with interfaces

The main difference to the OMAF-DASH client for streaming as depicted in Figure 2 is the fact that the access engine does carry out any subsegment concatenation, since the download file contains all tracks corresponding to the tiles and all extractor tracks corresponding to the different viewports. The other main difference is that the VR applications has an interface to the track selection function to choose what extractor track to play back (3) based on the sensor data (i.e. based on the current viewport of the user). The extractor track is thereby selected dynamically based on the current user viewport and may change as coding configuration and client implementation allow.

Therefore, OMAF metadata corresponding to the "main" viewport of the tracks (i.e. viewport corresponding to the regions with higher quality/resolution): namely `RegionWisePackingBox`, `SphereRegionQualityRankingBox` or the `2DRegionQualityRankingBox` has to be parsed in the track selection function (1) in order to select the corresponding extractor track (3) based on the current user viewport. The entire ISO BMFF File is passed through to the ISO BMFF Parser (2) where, only the selected extractor track is parsed and a single corresponding HEVC Main10 Main Tier Level 5.1 elementary stream (4) is passed to the Media Decoder. Then the stream is decoded and decoded pictures (5) are forwarded to the renderer along the necessary metadata for rendering.

4.2.2.1.3.3 External specifications

Video elementary streams are encoded following the requirements in ISO/IEC 23090-2 [OMAFFDIS] section 10.1.3.2.

ISO BMFF Tracks are encoded following the requirements in ISO/IEC 23090-2 [OMAFFDIS] section 10.1.3.3. ISO BMFF files that contain such an encoded track are identified by the brand '`hevd`'. The OMAF metadata is equivalent to the information that is present in the SEI messages for omnidirectional video as specified in ISO/IEC 23090-2 [OMAFFDIS] section 10.1.2.3.

DASH Integration is provided following the requirements and recommendations in ISO/IEC 23090-2 [OMAFFDIS] Annex D.1.2. An Adaptation Set including Representations formatted according to this media profile is recommended to be signaled as

- either (when corresponding to the Main Adaptation Set if `Preselection` is used or the Adaptation Set contains dependent Representation with `@dependencyId`)
  - `@codecs='resv.podv+ercm.hvc2.1.6.L93.B0'`

    - @mimeType='video/mp4 profiles="hevd"'
    - A Supplemental Descriptor or Essential Descriptor providing the frame packing arrangement may be used
- or (otherwise)
    - @codecs='resv.podv+erpv.hvc1.1.6.L93.B0' or 'resv.podv+ercm.hvc1.1.6.L93.B0'
    - @mimeType='video/mp4 profiles="hevd"'
    - A Supplemental Descriptor or Essential Descriptor providing the frame packing arrangement may be used

Note: Signaling of other parameters such as applied color transform and transfer characteristics is under discussion in MPEG and may be added.

4.2.2.1.3.4 Quality and Performance

Firstly, according to the results presented in 3GPP SA4 in 3GPP TR 26.918 [TR26918] in section 7.3, e.g. table 7.3 [VDVS-QUAL], significant bitrate reduction can be achieved by tiling the omnidirectional video and mixing different resolution as enabled through this viewport dependent baseline media profile using HEVC Tiles.

Secondly, and as pointed out in section 2 of [VDVS-QUAL] with examples in section 10.1, compared to viewport-independent distribution methods, higher resolution can be obtained in the viewport by used viewport-dependent distribution methods such as this viewport dependent baseline media profile using HEVC Tiles.

4.2.2.2 Audio

4.2.2.2.1 Overview

This section defines media profiles for audio. Table 4 provides an overview on the supported features, but is not considered to be comprehensive. The detailed specification is subsequently provided in the referred section.

Table 4: Overview of OMAF media profiles for audio

Media Profile	Codec	Profile	Level	Max. Sampling Rate	Brand	Section
3D Audio Baseline	MPEG-H Audio	Low Complexity	1, 2 or 3	48 kHz	'oabl'	10.2.2

4.2.2.2.2 OMAF 3D Audio Baseline Media Profile

4.2.2.2.2.1 Introduction

This media profile fulfils the requirements to support omnidirectional audio. Channels, objects and Higher-Order Ambisonics (HOA) are supported, as well as combinations of those. The profile is based on MPEG-H 3D Audio [3DA].

MPEG-H 3D Audio [3DA] specifies coding of immersive audio material and the storage of the coded representation in an ISO Base Media File Format (ISO BMFF) track. The MPEG-H 3D Audio decoder has a constant latency, see Table 1 – "MPEG-H 3DA functional blocks, internal processing domain and delay numbers" of ISO/IEC 23008-3 [3DA]. With this information, content authors can synchronize audio and

video portions of a media presentation, e.g. ensuring lip-synch. When orientation sensor inputs (i.e. pitch, yaw, roll) of an MPEG-H 3D Audio decoder change, there will be some algorithmic and implementation latency (perhaps tens of milliseconds) between user head movement and the desired sound field orientation. This latency will not impact audio/visual synchronization (i.e. lip synch), but only represents the lag of the rendered sound field with respect to the user head orientation.

MPEG-H 3D Audio specifies methods for binauralizing the presentation of immersive content for playback via headphones, as is needed for 360° VR presentations. MPEG-H 3D Audio specifies a normative interface for the user's orientation, as Pitch, Yaw, Roll, and 3D Audio technology permits low-complexity, low-latency rendering of the audio scene to any user orientation.

4.2.2.2.2.2 External Specification

Audio elementary streams are encoded following the requirements in ISO/IEC 23090-2 [OMAFFDIS] section 10.2.2.2.

ISO BMFF Tracks are encoded following the requirements in ISO/IEC 23090-2 [OMAFFDIS] section 10.2.2.3. The ISO BMFF track should be identified by the brand '`oabl`'.

Note: The Media Profile specified in MPEG does not contain yet a compatibility brand. The '`oabl`' brand is used for compatibility with CMAF Media Profile for OMAF 3D Audio Baseline Media Profile. It is envisioned that this will be corrected at the next MPEG meeting, and the '`oabl`' brand will be used for the OMAF 3D Audio Baseline Media Profile while the '`cabl`' brand will be used for compatibility with CMAF Media Profile for OMAF 3D Audio Baseline Media Profile, as specified in current document.

DASH Integration is provided following the requirements and recommendations in ISO/IEC 23090-2 [OMAFFDIS] section 10.1.3.4. An Adaptation Set including Representations formatted according to this media profile should provide the following signaling according to [RFC6381] and ISO/IEC 23008 3 [3DA] section 21 as shown in Table 5.

Table 5: MPEG-H Audio MIME parameter according to RFC 6381 and ISO/IEC 23008-3

Codec	Media type	codecs parameter	profiles	ISO BMFF encapsulation
MPEG-H Audio LC Profile Level 1	audio/mp4	mhm1.0x0B	'`oabl`'	ISO/IEC 23008-3
MPEG-H Audio LC Profile Level 2	audio/mp4	mhm1.0x0C	'`oabl`'	ISO/IEC 23008-3
MPEG-H Audio LC Profile Level 3	audio/mp4	mhm1.0x0D	'`oabl`'	ISO/IEC 23008-3
MPEG-H Audio LC Profile Level 1, multi-stream	audio/mp4	mhm2.0x0B	'`oabl`'	ISO/IEC 23008-3
MPEG-H Audio LC Profile Level 2, multi-stream	audio/mp4	mhm2.0x0C	'`oabl`'	ISO/IEC 23008-3
MPEG-H Audio LC Profile Level 3, multi-stream	audio/mp4	mhm2.0x0D	'`oabl`'	ISO/IEC 23008-3

4.2.2.2.2.3 Quality and Performance

MPEG-H Audio LC Profile provides excellent sound quality (as assessed per ITU-R BS.1534) for 2D and 3D program material as shown in 3D Audio Verification Test Report [N16584].

4.3 Content Security

This section presents guidelines for protecting Virtual Reality content. The word "content" is used advisedly: there are non-content VR assets that are not in the scope of VR content security. In these guidelines, VR content is defined as a new audio-visual media type that shares many characteristics of traditional linear audio-visual content but differs in significant ways (mainly rendering and display).

Accordingly, these guidelines highlight those differences from traditional video that have a material impact on VR content protection. This approach is based on the assumption that existing Digital Rights Management systems will be used as a baseline. The format will indicate the limitations of using existing DRM on VR content, allowing informed judgment to be made as to where additional protection mechanisms may be required.

Differences notwithstanding, the goals of protecting traditional and VR content are fundamentally the same. That is to protect the rights of the copyright holders and distributors. Of particular concern are unauthorized distribution (by whatever means), unauthorized modification (delivering an inferior experience), and violation of user privacy. There are also new usage rights VR content owners may want to control, e.g. entitlements for resolution, output control, or 3DOF vs 6DOF navigation.

4.3.1 Scope

VR experiences come in many forms. There is a diversity of presentation formats (e.g. augmented, immersive), display devices (e.g. HMDs, tablets), and distribution methods (e.g. streamed vs. downloaded). The landscape is constantly changing, and keeping up the security requirements of the entire ecosystem is not practical.

The targets of this section's guidelines are those designing or specifying commercial systems for viewing VR content. Guidelines are therefore confined to the use cases identified by the VRIF's Requirements and Guidelines committees (see the appendix for relevant documents). They are further restricted to those parts of the ecosystems developed enough to be commercially realizable, and based on standards.

A small set of characteristics are used to distinguish the systems discussed in the security guidelines. They are content type, distribution system, and interaction model, as shown in Table 6.

Table 6: VR Characteristics

Category	Video Format	Audio Format	Distribution Format	Duration	Live	Client Type
360° Video	Spherical	Spatial	DASH	Fixed	No	Player
	Spherical	Spatial	DASH	Open-ended	Yes	Player
	Spherical	Spatial	Downloaded ISO BMFF	Fixed	No	Player
Theatrical VR	Spherical	Spatial	n/a	Fixed	Either	Player

Interactive VR	Mixed[4]	Spatial	Downloaded	Scripted	No	Game Engine[5]

The primary focus is 360° video that is streamed to clients equipped with a player (including web browsers extended to support view-dependent projection). The bulk of this content will be streamed, but downloading content in its entirety before playback will also have use cases.

Live 360° content is similar but will always be streamed. The major difference is that potentially time-consuming operations such as encoding or watermarking must be done in near real-time (introducing no more than a few seconds of latency).

Theatrical VR is a special case of 360° video where a traditional piece of traditional high-value content (e.g. a movie or TV show) is embedded in a virtual theatre environment. It is a special case because 1) the video source may have its own DRM (e.g. streaming video or Blu-Ray disk; 2) a compositing operation of high-value content and background is required; 3) the high-value content is by definition attractive to pirates.

Interactive VR refers to content intended for high-end VR systems such as the Oculus Rift, although it will increasingly be found on mobile devices as well. The key difference is that instead of only a fixed 360° video sequence, the images sent to the display are under the control of software, and may be a composite of multiple media types including CGI imagery. These non-video media types may have a value which the owner or creator wants to protect or track. Hence, they may have their own security requirements and corresponding solutions which may or may not be distinct solutions from the video element's security solutions.

Any of this content may be viewed with a HMD (fully immersive), or on a non-immersive device such as a tablet. In the former case the viewport is determined by the orientation of the HMD. In non-immersive displays, orientation may also be determined by sensors, or may be controlled directly by the user (e.g. finger swipes). This sort of display is sometimes referred to as a "magic window."

4.3.2 MovieLabs ECP 1.1 Deltas

The MovieLabs Specification for Enhanced Content Protection (ECP) describes a set of high-level requirements for securing content. They are intended to be general enough to be applicable to any content distribution system (including future ones), but specific enough to serve as a template for evaluating a specific instance of such a system. No document can substitute for a certification process, but the ECP permits rapid identification of potential problem areas.

Although the ECP is a good starting point for defining guidelines for systems distributing VR content, additional work is required because VR is a fundamentally different media type than traditional audio-visual content. Some of these differences require new mechanisms (or modifications of existing ones) to secure the VR content. In this section those differences are noted as deltas to the ECP. They are enumerated following the format of the ECP, which is divided into four major sections: 1) threats to content; 2) requirements for DRM systems; 3) Platform Specifications; and 4) End-to-End System Specifications.

[4] Visualizable media types besides spherical video include CGI Models, textures, point clouds, etc.

[5] Or some other stand-alone application to handle interactivity.

4.3.2.1 Problems/Threats

The ECP enumerates several threats to protected traditional content. These are also applicable to VR content, with the following notes:

- **Ripping Software (360° video):** For 360° video, the most common distribution format will be adaptive streaming, which is not strictly speaking ripping. Circumvention efforts will focus on these new formats.
- **Availability of Rips (360° video):** If the video is not tiled, transcoding it to a standard format and putting it on file-sharing and torrents is the logical path. If tiles are used to distribute, such reconstruction will be much more difficult because of the difficulty of getting all the tiles and dealing with padding.
- **Ripping Software (Interactive Applications):** These are similar to AAA games, and similar attacks can be expected. These are focused on reverse-engineering the application well enough to remove the section involved with authentications.
- **Availability of Rips (Interactive Applications):** As with AAA games, hacked versions with authentication logic removed will be created and distributed.
- **Output Capture:** Intercepting the output of VR content may take more work to make useful because only the current viewport is displayed. There is an exception in the case of live sports and other events: others may be willing to accept another person's view of a premium event such as a championship game.

4.3.2.2 New Problems/Threats

VR has some issues that are new and not covered by the ECP. These are listed below:

- **Return Path Data:** Users engaging in VR experiences generate several types of return path data, including the HMD tracking data and controller input. This data is a form of "digital exhaust": it must be protected from interception and properly anonymized, otherwise it could reveal information about the user.
- **Degradation of Experience:** A user in an immersive experience is more vulnerable to disruption caused by targeted (e.g. spoofing input data) or generic (DDoS) attacks.

4.3.3 DRM System Specifications

The ECP lists requirements for DRM systems hosting protected content. The same requirements apply to systems hosting VR content, with the following notes:

- **Connection:** Not exclusive to VR, but note that broadcast environment includes scenarios that may require different content protections techniques, such as:
  - Online full 2-way communication
  - Online 1-way (broadcast)
  - Intermittent
  - Completely offline
- **Outputs & Link Protection:** DRM on the output (e.g. HMD view) needs to be selectable to account for use cases where re-transmission of a user's output (e.g. Twitch) is desirable. Also, many existing HMDs use chipsets that only support HDCP 1.4.

4.3.4 Platform Specifications

The ECP lists requirements for hardware platforms hosting protected content. The same requirements apply to platforms hosting VR content, with the following notes:

- **Encryption:** VR content has the same requirements for encryption as traditional content.
- **Secure Media Pipeline:** there are several issues that must be taken into account:
  - VR content that requires special hardware such as a GPU should be part of the SMP (note that. this may not be possible on some current platforms).
  - Interactive VR content is not a simple decode, it may composite different media types.
- **Secure Computation Environment:** VR content has the same requirements for a Secure Computation Environment as traditional content.
- **Hardware Root of Trust:** VR content has the same requirements for a Hardware Root of Trust as traditional content.

4.3.5 End-to-End System Specifications

This section captures requirements whose implementation extends across multiple system components (e.g. client and server), with the following notes:

- **Forensic Watermarking:** VR audio-visual content is created and rendered differently from traditional content so current algorithms will not work, for example:
  - The Field of View in the HMD may only be a small percentage of the full 360° frame. This violates the assumption that a full frame is displayed.
  - The various VR display transforms (equirectangular projection, spherical lens warping, correcting for chromatic distortion, foveated rendering) may make the watermark difficult to recover.

  This is an ongoing area of investigation; forensic watermarks are required to respond to breaches.
- **Playback Control Watermark:** recovering audio watermarks may be hindered by spatial audio processing. It is not clear this is relevant to VR content.
- **Breach Response:** VR content has the same requirements for breach responses as traditional content.

4.3.6 Encrypted Media Extensions

Encrypted Media Extensions (EME) is a recommendation developed by the W3C [EME]. It provides a framework for web browsers to support playback of HTML5 video protected by DRM without the use of plug-ins. Although controversial because it enables proprietary technology to be used in the otherwise open browser ecosystem, the W3C has published EME as a web standard. EME is recommended for securely displaying VR content in an HTML5-based browser. For non-browser playback (i.e. native applications), traditional DRM systems should be deployed.

5 Vertical 1: OTT Download or Streaming of VR360 Content

5.1 Description of Vertical

This vertical primarily addresses the economically feasible distribution of VR360 content to emerging devices.

5.2 Guiding Example Use Cases

A service provider (content aggregator) offers a library of 360° A/V VR content. The library is a mixture of content formats from user generated content, professionally generated studio content, VR documentaries, promotional videos, as well as highlights of sports events. The content enables to change the field-of-view based on user interaction.

The service provider wants to create a portal to distribute the content to a multitude of devices that support 360° A/V and VR processing and rendering. This device may implement functions in hardware for reduced power and battery consumption, optimized processing, minimal thermal impacts and minimized latencies. Some solutions may be embodied in software (such as apps). Typically, VR applications make use of well-defined interfaces to hardware functionality, notably decoders.

The service provider wants to target two types of applications:

- Primarily, view in a HMD with head motion tracking.
- As a by-product, the content provider may permit viewing on a screen with the field-of-view for the content adjusted by manual interaction (e.g. mouse input or finger swipe)

The service provider expects different types of consumption and rendering devices with different capabilities in terms of decoding and rendering. However, it wants to target devices that fulfil a certain quality threshold expressed by decoder and rendering capabilities.

The service provider has access to the original footage of the content and is permitted to encode and transcode to appropriate distribution formats.

The footage includes different types of 360° A/V VR content, such as

- For video:
  - One of
    - Pre-stitched monoscopic video, i.e. a (360° and possibly less than 360°) spherical video without depth perception, with Equirectangular Projection (ERP).
    - Pre-stitched stereoscopic video, i.e. a spherical video using a separate input for each eye, typically with ERP.
  - Original content
    - Original content, either in on original uncompressed domain or in a high-quality mezzanine format.
    - Basic VR content: as low as 4k × 2k (ERP), 10bit, BT.709, as low as 30fps
    - High-quality: up to 8k × 4k (ERP), 10 bit, possibly advanced transfer characteristics and color transforms, sufficiently high frame rates, etc.
  - Sufficient metadata is provided to appropriately describe the A/V content

- For audio:
  - Spatial audio content for immersive experiences, provided in the following formats:
    - Channel-based audio
    - Object-based audio
    - Scene-based audio
    - Or a combination of the above
  - Sufficient metadata for encoding, decoding and rendering the spatial audio scene permitting dynamic interaction with the content. The metadata may include additional metadata that is also used in regular TV applications, such as for loudness management.
  - Diegetic and non-diegetic audio content.

The service provider is responsible for monetizing the content and fulfilling necessary accessibility requirements. Subtitles are considered to be important and need to be supported in a standardized way.

The service provider is also responsible for securing the content, if required by the content provider, including DRM systems.

The service provider wants to

- reach as many devices as possible
- minimize the number of different formats that need to be produced and distributed
- ensure that the content is presented in highest quality on the different devices.

The service provider provides an application (e.g. browser-based, native app) or makes use of an installed third party application, and may rely on the decoding and rendering capabilities of the device, typically in hardware or by pre-installed or downloaded software decoders.

The service provider wants to reach devices that are already in the market or are expected to be in the market by end of 2017.

The service provider wants to avoid testing each device, but rather prefers simple interoperability, e.g. standardized interfaces.

The service provider wants to enable that some of the library items can be downloaded to devices, primarily through HTTP, and is played back on the device after downloading. The service provider wants to ensure that a device downloads only content that it can decode and render while providing the best user experience for the device capabilities.

For certain library items, the service provider wants to ensure that content is rendered instantaneously after selection, so a DASH-based streaming is considered. The service provider wants to ensure that a device accesses only content that it can decode and render while providing the best user experience for the device capabilities. The service provider also wants to ensure that the available bandwidth for the user is used such that the rendered content for the user is shown in the highest quality possible.

5.3 Reference Architectures

5.3.1 Distribution Architecture

The architecture introduced in this section addresses service scenarios for the distribution of VR content in file or segment based download and streaming services, including DASH-based services.

The role of the VR Content Provider, the VR Service provider, the distributor, the application and the service platform are differentiated.

Figure 5 considers a functional architecture for such scenarios. VR Content is captured by a VR Content provider and split in audio Ba and video in Bv on the interfaces. Both media come with metadata and are synchronized in time and space. The content is uploaded to a VR Service Provider Portal which stores the original footage. Then the content is prepared for distribution by pre-processing, encoding and file format/DASH encapsulation. Interfaces Da and Dv provide formats that enable encoding by existing media encoders. After media encoding, the content is made available to file format encapsulation engine as elementary streams E and the file format may generate a complete file for delivery or segmented content in individual tracks for DASH delivery over interface F. Metadata may be added. Content may be made available in different viewpoints, so the same content may be encoded in multiple versions. Content may also be encrypted.

At the receiving end, there is an expectation for the availability of a VR application that communicates with the different functional blocks in the receiver's VR service platform, namely, the delivery client, the file format decapsulation, the media decoding, the rendering environment and the viewport sensors. The reverse operations are performed. The communication is expected to be dynamic, especially taking into account the dynamics of sensor metadata in the different stages of the receiver. The delivery client communicates with the file format engine, and different media receivers decode the information and provide also information to the rendering.

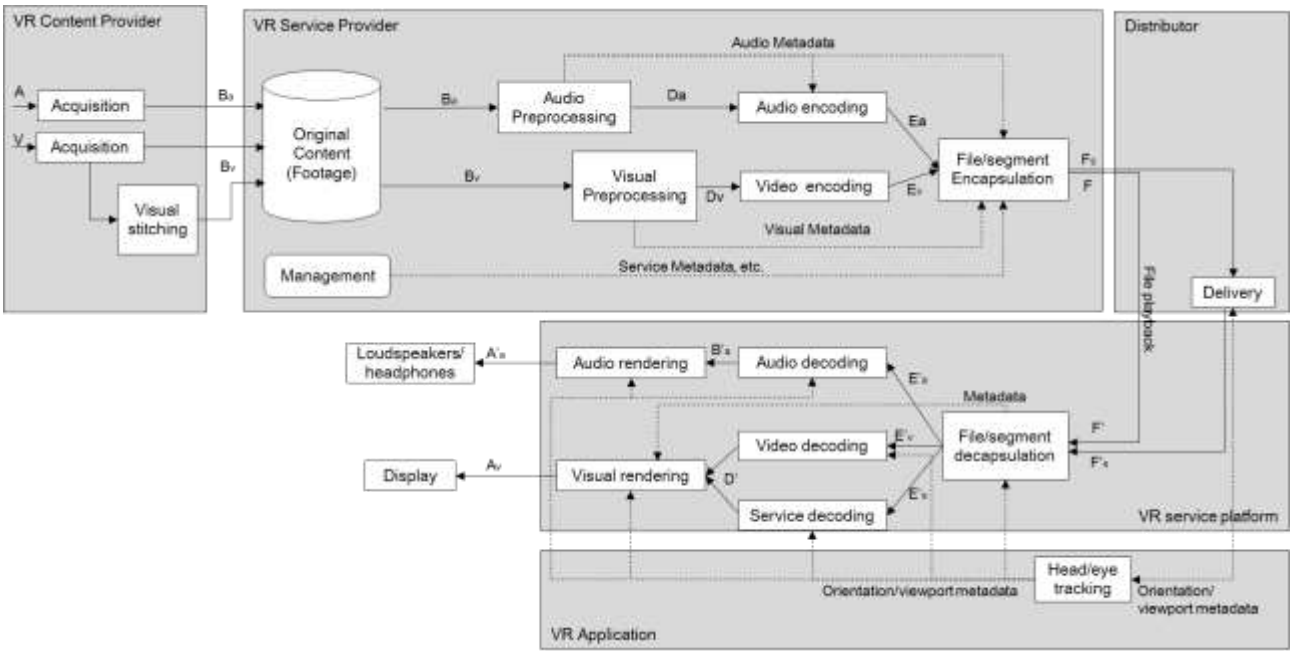

Figure 5: Example Architecture for simple VR Streaming Services

Note that certain functionality (such as audio decoding and audio rendering) depicted in the VR Service Platform box above may in certain circumstances take place within the VR Application box, depending on the VR Service Provider's needs and platform capabilities. However, there are benefits in enabling VR Applications to use a native VR Service platform for decoding and rendering to minimize latency, thermal impact, processing power and power consumption.

Figure 6 provides an attempt for an encrypted system. It is important that the interfaces to the secure decoding and rendering platform are limited and therefore require that well defined conforming bitstreams are provided to the secure media pipeline.

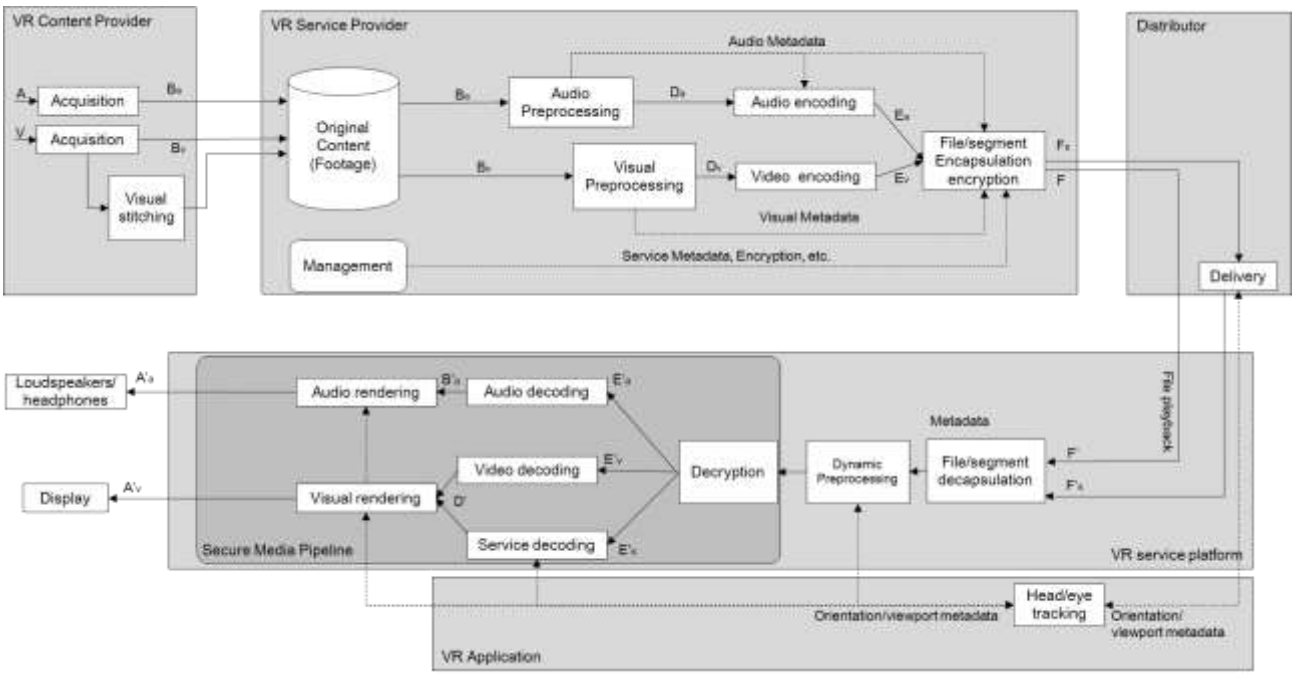

Figure 6: Example Architecture for encrypted VR Streaming Services

5.3.2 Client Architecture

In the distribution architectures depicted in Figure 5 and Figure 6, a VR compute platform (e.g., PC, console, tablet, smartphone, etc.) receives the delivered VR content, performs processing toward de-capsulation, decoding and rendering of the VR content, and forwards the processed input to the displayed at the VR device (e.g., HMD). VR device includes loudspeakers/headphones, display and sensors for head/eye tracking. Available communication technologies to realize the interface between the VR compute platform and VR device include USB and HDMI. This is depicted in Figure 7.

Note: It should be noted that an exception to the client-side depicted in Figure 7 is the integrated HMD platform where VR compute platform is part of the VR device.

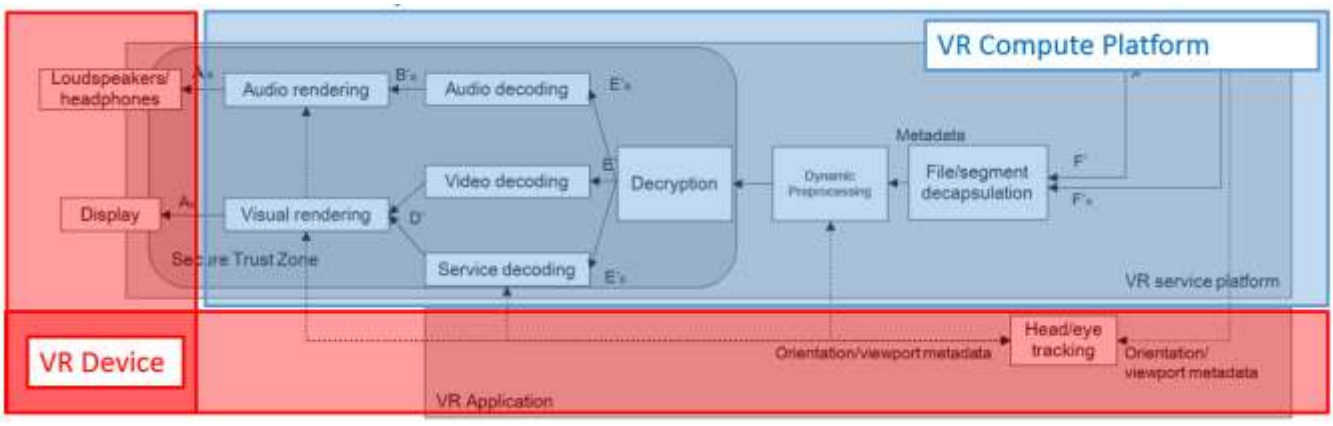

Figure 7: Client side processing on the example architecture

An example high-level VR software stack is depicted in Figure 8, depicting the relationships between VR application, VR runtime and VR hardware / device drivers on the VR compute platform. Here VR hardware for instance includes HW capabilities (e.g., CPU/GPU) on the VR platform for decoding and rendering. Device drivers for instance include drivers for display and head/eye tracking sensors, and thus provides access to devices.

A bi-directional application interface (i.e., APIs) between the VR application and VR runtime carries information both ways. Another directional driver (HW) interface between the VR runtime and VR hardware / device drivers carries information both ways. Data mostly flows upward in this diagram from the drivers to the application, but some requests (like haptics on controllers) flows back down to the driver, and is then passed on to the actual device.

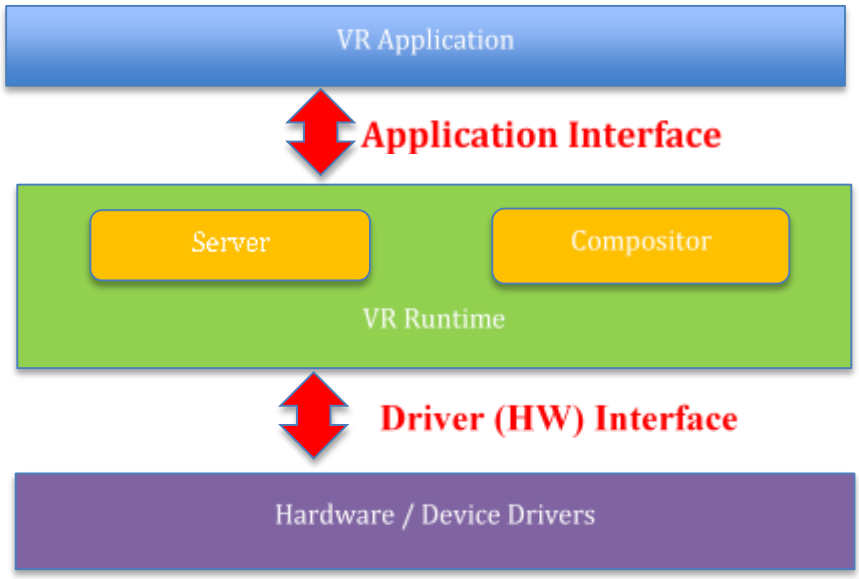

Figure 8: High-level VR software stack

The VR runtime interacts with both the VR application and device drivers via the APIs through the application and driver (HW) interfaces, and contains functions such as server and compositor functions. As such, VR applications do not have to interact with the devices directly, but instead the interaction occurs through the logical abstractions as enabled by the VR runtime. The VR runtime may also manage simultaneous interactions with multiple physical devices (e.g., display, controller, sensors, etc.) and may activate only a subset of devices to optimally use the VR platform resources.

The server function in the VR runtime includes functions such as predictive tracking to estimate and project the head pose based on sensor data. The compositor in the VR runtime performs functions such as Asynchronous Time Warp (ATW) and/or Asynchronous Space Warp (ASW) to display smooth motion even if the application is unable deliver new frames on time, by processing previously rendered frames based on predicted changes in user's head orientation. Other operations performed by the VR runtime includes taking pre-distorted images from the VR application and applying of barrel distortion to correct for pincushion distortion caused by the lenses. Moreover, the VR runtime further can interact with the CPU/GPU on the platform to perform other operations such as decoding, decryption and graphics rendering.

For the application interface, API examples on the information from the VR application to the VR runtime includes pre-distortion image to display and haptics information. API examples on the information from the VR runtime to the VR application include predicted poses and controller/peripheral states.

For the driver (HW) interface, API examples on the information flowing from the VR runtime to the VR device drivers include the output of the audio rendering and visual rendering, and haptics information. API examples on the information flowing from the VR device to the VR runtime include head/eye tracking information, controller/peripheral state, and sensor data, e.g., hand/foot sensor information.

It is desirable to standardize APIs for the above mentioned two interfaces, i.e., application interface and driver (HW) interfaces. Without a cross-platform standard, VR applications, games and engines must port to each vendors' APIs. In turn, this means that each VR device can only run that apps that have been ported to its SDK. The result is high-development costs and confused customers – limiting market growth. To address this gap, the Khronos OpenXR working group is currently standardizing the APIs for these two interfaces with the goal of avoiding market fragmentation.

5.4 Technical Enablers

5.4.1 Suitable Media Profiles

5.4.1.1 Video

The HEVC-based viewport-independent OMAF video profile as presented in section 4.2.2.1.2 may be used to fulfil the use case and provide broad interoperability.

The HEVC-based viewport-dependent OMAF video profile as introduced, may be used to fulfil the use case and provide broad interoperability, while achieving a higher resolution at the viewport than that of the HEVC-based viewport-independent OMAF video profile.

5.4.1.2 Audio

The OMAF 3D Audio Baseline Media Profile may be used to fulfil the use case and provide broad interoperability.

5.4.2 Suitable Presentation Profiles

The presentation profiles defined in clause 11 of [OMAFFDIS] are applicable to this use case. Any other media information that is added to the presentation should utilize the same coordinate system as the video and audio formats to ensure proper rendering and presentation.

5.4.3 Distribution Systems

5.4.3.1 OTT Distribution using HTTP

This section primarily addresses interface F on distribution in Figure 5. In the context of this use case, streaming and download are considered. In recent years the use of HTTP-CDNs for distribution of content over the open Internet has gained more and more attraction and is nowadays the premium OTT distribution means. The popularity of HTTP is manifold, but primarily the scalability of CDNs with distributed caching architectures, the ability to pass firewalls and NATs as well as the ability of HTTP protocol stacks on many different devices as well as in browser endpoints makes HTTP the most far-reaching distribution protocol. The downsides of TCP/IP and HTTP (such as variable bitrates, object-based delivery, download latencies, etc.) have been compensated by smart formats and the usage of the formats in different applications. Predominantly download and adaptive bitrate (ABR) streaming is used for

distributing new media services. Furthermore, the formats and the HTTP-based APIS also more and more have found support in not only unmanaged distribution, but also in IP multicast as well as in broadcast, e.g. in 3GPP MBMS or ATSC3.0. In addition, HTTP architectures and CDNs are continuously improved adding new protocols such as HTTP/2.0 or combine unicast and multicast protocols.

While for regular TV applications, downloading an entire content is considered less and less appealing (except for recent services that enable offline access to content libraries, for example in air planes), for VR services and content download may still be a very attractive option, in particular if the content is short, the content is not live and/or the real-time access to high-quality content is not possible due to bitrate restrictions. Hence, download will remain to be an attractive option and with the use of HTTP-based download, CDN capabilities can be fully exploited. It is beneficial that the application can check the content before downloading it and HTTP headers and capability exchange can be used for this purpose together with the formats well defined Internet media type using the `Content-Type` HTTP header.

The most popular standardized protocols for streaming are HTTP Progressive Download and Dynamic Adaptive Streaming over HTTP (DASH). Both enable to provide services to deliver on-demand VR content over HTTP protocols, including the metadata and media data composing the on-demand VR service. As such, again standard HTTP servers and standard HTTP caches can be used for hosting and distributing on-demand VR content. Note that in the context of this distribution model, it is expected that the delivery network is typically unaware of the content, whether it is VR or any other media content. This makes HTTP-based delivery attractive for launching services.

An example VR distribution system is depicted in Figure 9. On-demand VR content including media content and metadata may be stored on one or more media origin servers, along with the DASH media presentation description (MPD). The MPD contains the relevant information on the different encoded versions of the DASH VR content, including VR-related content information such as those on available viewports, projection and region-wise packing metadata, along with the traditional MPD parameters on codecs, bitrates and resolutions. The media origin server is typically an HTTP server such that the MPD and media segments related to on-demand VR content can be requested via clients and be delivered via HTTP. A DASH client in the user terminal obtains a current viewing orientation or viewport e.g. from the HMD that detects the head orientation and possibly also eye orientation. By parsing metadata from the MPD, the DASH client concludes which Adaptation Set and Representation cover the current viewing orientation at the highest quality and at a bitrate that can be afforded by the prevailing estimated network throughput. The DASH client issues (Sub)Segment requests accordingly. In case of HTTP progressive download, the on-demand VR content may be included in an ISO base media file format (ISO BMFF) file as one track and the entire ISO BMFF file may be offered on an HTTP server or on a CDN for downloading.

The massively scalable distribution of on-demand VR content is typically enabled via content delivery networks (CDNs) that consist of a geographically distributed set of HTTP proxy caches, with the goal of enabling content access to end users with high availability, proximity and high performance. At the network edge, VR content may be delivered through the client devices via different access networks, such as 4G/5G access and cable/WiFi access. Client compute devices (e.g., PC, tablet, smartphone, etc.) typically perform processing toward de-capsulation, decoding and rendering of the VR content, and forward the

processed input to be displayed at the HMD.

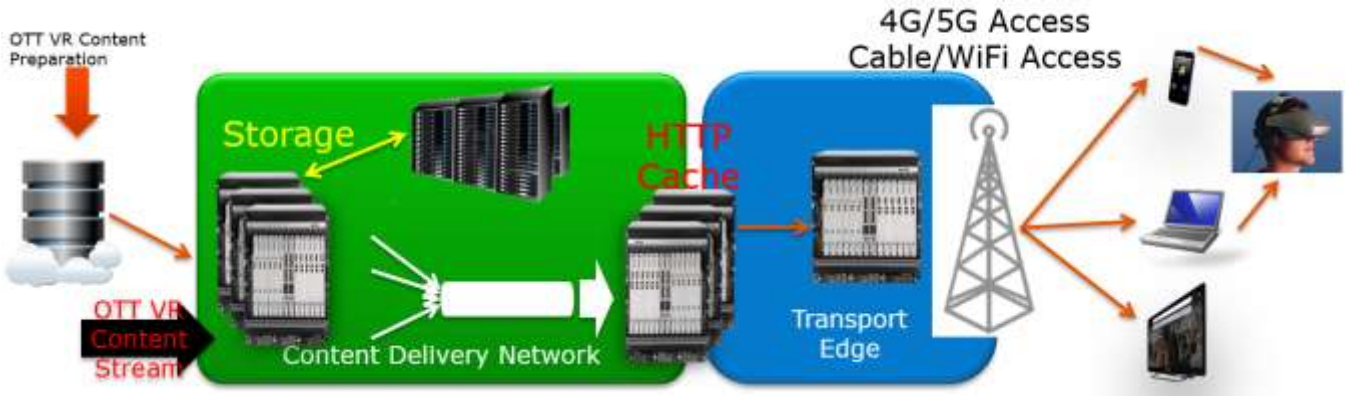

Figure 9: Example VR distribution system

Figure 10 shows an example protocol stack for on-demand VR content delivery services. OMAF-based media content, associated metadata, DASH-formatted MPD and media segments can be made accessible through HTTP.

<table>
<tr><td>Video
Audio</td><td rowspan="2">DASH Media Presentation Description</td></tr>
<tr><td>OMAF File Format</td></tr>
<tr><td colspan="2">HTTP</td></tr>
<tr><td colspan="2">TCP</td></tr>
<tr><td colspan="2">IP</td></tr>
</table>

Figure 10: Example protocol stack for VR content distribution

5.4.3.2 Download

If the HEVC-based viewport-independent OMAF video profile is used for download, one track is included in the ISO BMFF file that follows the requirements and recommendations of the media profile in section 4.2.2.1.2.

If the HEVC-based viewport-dependent OMAF video profile is used for downloading tiles, one 'hvc1' track per tile per resolution is included in the ISO BMFF file and one 'hvc2' track per potential viewing direction is included in the ISO BMFF that follow the requirements and recommendations of the media profile in section 4.2.2.1.3.

If the OMAF 3D Audio Baseline Media Profile is used for download, one track is included in the ISO BMFF file that follows the requirements and recommendations of the media profile in section 4.2.2.2.2, with the first sample of the movie as a Stream Access Point (SAP) of type 1 (i.e. sync sample).

5.4.3.3 DASH Distribution

If the HEVC-based viewport-independent OMAF video profile is used for DASH-based streaming, on Adaptation Set is included in each Period based on the requirements and recommendations of the media profile in section 4.2.2.1.2.

If the HEVC-based viewport-dependent OMAF video profile is used for tile-based DASH streaming, one Adaptation Set per tile per resolution is included in each Period based on the requirements and recommendations of the OMAF media profile in section 4.2.2.1.3. In addition, one Adaptation Set per tile per potential viewing direction is included in each period that contains a `@dependencyId` attribute or a `Preselection` as a Supplemental descriptor.

Figure 11 shows an exemplary DASH configuration where two tiles (Tile 1 and Tile 2) with two different resolutions (high and low resolution) are grouped into separate Adaptation Sets (1, 2, 3 and 4). Each of those Adaptation Sets contains three Representations where each MCTS track is encoded with a different bitrate. In addition, two Adaptation Sets (5 and 6), representing two different viewport orientations, are included in the same Period pointing to the correct Adaptation Sets containing one of the tiles in high and one of the tiles in low resolution.

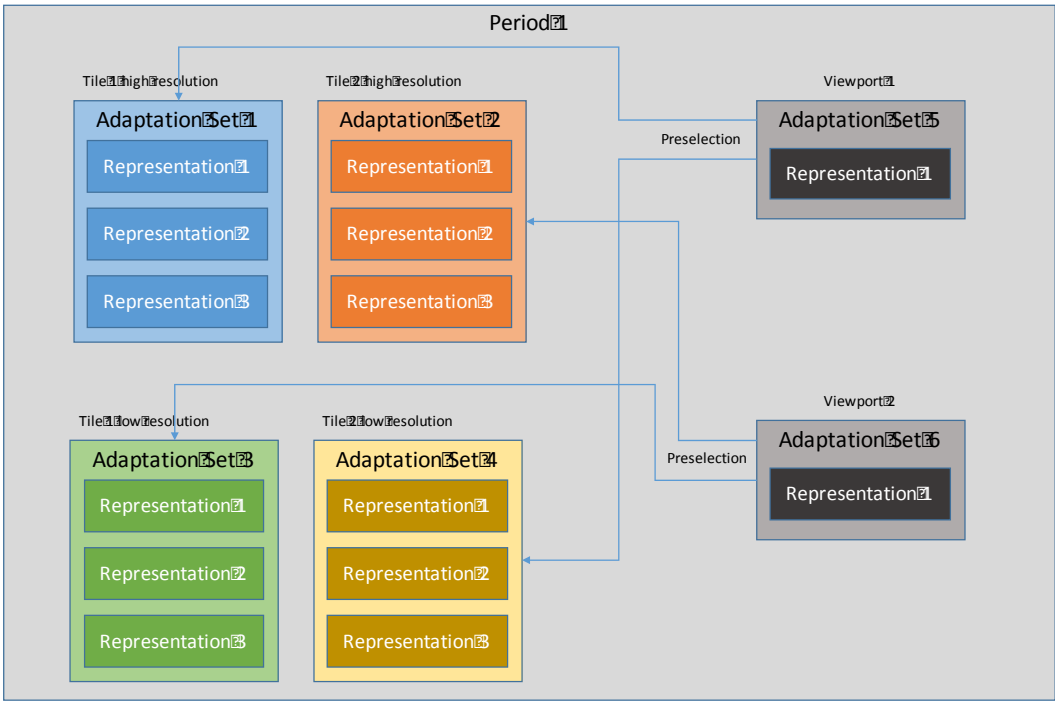

Figure 11: Exemplary DASH configuration setup

If the OMAF 3D Audio Baseline Media Profile is used for DASH-based streaming, one Adaptation Set is included in each Period based on the requirements and recommendations of the media profile in section 4.2.2.2.2, with the first sample of every fragment as a Stream Access Point (SAP) of type 1 (i.e. sync sample).

5.4.3.4 CDN Considerations

The role of a CDN is critical for enabling access to on-demand VR content with high availability and high performance. On top of the traditional adaptive bitrate (ABR) / DASH content delivery, on-demand VR content delivery requires further degree of client adaptation as it involves continuous change of the user's

viewports (e.g., as the user wearing the HMD changes his/her head orientation), which makes it important that CDNs enable means to reduce download latencies toward enabling interactive VR experiences with high degree of responsiveness. High latency in delivering the user viewport may cause poor user experience, including VR sickness. CDN may benefit from edge computing techniques deployed in proximity to the users in the service provider or operator network to allow the content served close to the clients, and thereby minimizing latency and optimizing network bandwidth efficiency.

CDN strategies toward improving OTT VR content delivery may depend on the specific OMAF media profile in use. In particular, CDN considerations could differ between viewport independent vs. viewport dependent media profiles. In case of viewport independent VR content format, the viewport agnostic nature of the VR content delivery implies that it is sufficient for the CDN to apply the existing ABR / DASH content delivery enhancement mechanisms (and regular DASH clients may be used), but also ensure that the delivered throughput performance meets the bandwidth requirements of VR content delivery (e.g., as reported in sections 7.2 and 7.3 of 3GPP TR 26.918 [TR26918] for viewport-independent and viewport-dependent media profiles, respectively), since streaming the entire 360° panorama requires significant bandwidth. In case of viewport dependent VR content format, there are further optimizations possible at the CDN level. In this case, the DASH VR content may be made available for different viewports, so the same content and associated media segments may be stored in multiple versions. For example, the CDN may cache different DASH adaptation sets corresponding to different viewport versions (e.g., tiles) of the content to provide the client the ability to interactively switch across different viewports (e.g., in response to user's change in head orientation). In case of viewport dependent delivery via use of tiled streaming, this allows that high quality / resolution tiles corresponding to the new viewport can be fetched very quickly by the client from the CDN. Thus, CDN's VR content caching ability corresponding to different viewports may help towards reducing download delays and also potentially improving the network bandwidth utilization efficiency.

A relevant latency metric to consider for tiled streaming is motion to high resolution (M2HR) latency, defined as the latency from the change in head orientation to display of the first high resolution frame based on the new field of view. The sources contributing to M2HR latency as depicted in Figure 12 and include sensor latency, network request delay, origin-to-edge delay (in case of cache miss), transmission delay (accounting for access network delay) and delays incurred in the client device due to buffering, decoding and rendering. All of these latency sources are relevant for viewport-dependent streaming while only the rendering latency is relevant for viewport-independent streaming.

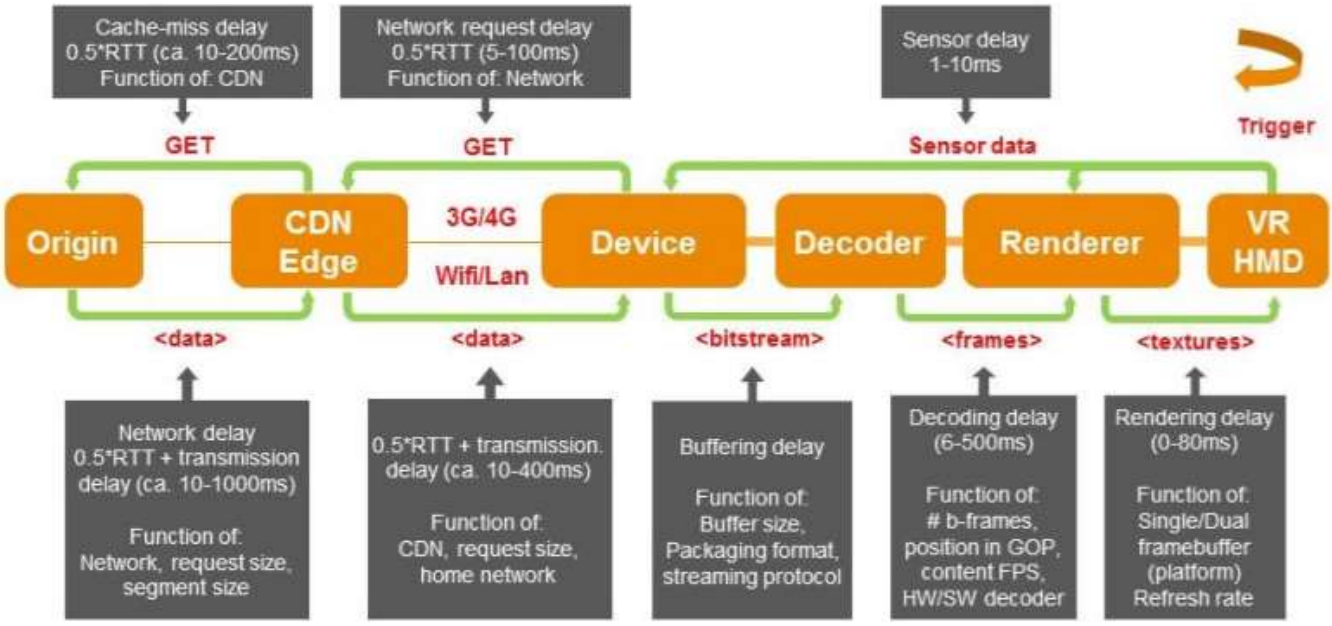

Figure 12: Latency sources contributing to M2HR latency [CDNOPT]

As reported in [CDNOPT], the M2HR latency cost of tile-based streaming can be reduced considerably by various CDN and streaming protocol optimizations including:

- Intelligent caching of tiles at the edge to avoid cache misses and consequent origin-to-edge latency
- Transport level improvements such as the use of HTTP/2.0 and QUIC

5.4.3.5 Access Network Considerations

5.4.3.5.1 3GPP/4G/5G

3GPP Packet Switched Streaming Service [TS26234] enables on-demand VR progressive download and DASH content delivery over HTTP protocols, using 3GPP-based core network architectures (e.g., evolved packet core (EPC)) and radio access air interfaces (e.g., Long Term Evolution (LTE)). 3GPP is currently developing a new 5G air interface that is expected to deliver data rates much higher than those of LTE. Feasibility of VR services in 3GPP networks has been studied in 3GPP TR 26.918 [TR26918].

5.4.3.5.2 Cable/Fiber/Copper and WiFi to Home

WiFi and cable/fiber/copper technologies provide access network connectivity to the home. WiFi is developing its next generation standards to deliver better data rates to support high-bandwidth applications such as VR, with data rates up to 2-4 Gbps (see [80211AC] and [80211AX]). Currently deployed fiber access technologies (G-PON and EPON) deliver speeds up to 3Gbps and next generation (XG-PON and 10G-PON) will deliver data rates up to 10 Gbps.

5.5 Guidelines for Service Providers

5.5.1 Suitable Production Formats

5.5.1.1 HEVC-based viewport-independent OMAF video profile

The video format is expected to conform to the video master format in clause 4.1.3.2 and video metadata is provided along with the video format.

The original video signal may be a full 360° sphere content.

The original video signal may also be restricted in coverage, i.e. only cover a subset of the full 360° sphere as indicated in the ***Coverage*** parameter.

The original video source may be monoscopic or stereoscopic as indicated by the ***StereoMode*** metadata element.

The following parameters are expected to be constant over the sequence of the content:

- Spatial resolution
- Frame rate
- Coverage

The original video format must be chroma subsampled from 4:2:2 to 4:2:0. After chroma subsampling, if the subsampled original video signal format is in the constraints of HEVC Main 10 Main Tier Level 5.1 and the scheme constraints, then the HEVC-based viewport-independent OMAF video profile may be used directly on the subsampled source signal to generate elementary streams following the media profile constraints as described in section 10.1.2.2 of [OMAFFDIS] and the encoding and content preparation in section 5.5.2.

The relevant HEVC Main 10 Main Tier Level 5.1 are:

- Max luma picture size is 8,912,896
- Max luma sample rate (samples/s) is 534,773,760
- The maximum bitrate (kbit/s) is 40,000

Monoscopic signals that can be distributed with this profile need to fulfill the following requirements:

- `ceil16`(***FullWidthPixel*** – (***Cropping.Left*** + ***Cropping.Right***)) *
  `ceil16`(***FullHeightPixel*** – (***Cropping.Top*** + ***Cropping.Bottom***)) ≤ 8,912,896
- `ceil16`(***FullWidthPixel*** – (***Cropping.Left*** + ***Cropping.Right***)) *
  `ceil16`(***FullHeightPixel*** – (***Cropping.Top*** + ***Cropping.Bottom***)) * ***FrameRate*** ≤ 534,773,760

with `ceil16(x)` the smallest integer that is greater or equal than x and a multiplier of 16.

Stereoscopic signals that can be distributed with this profile need to fulfill the following requirements:

- `ceil16`(***FullWidthPixel*** – (***Cropping.Left*** + ***Cropping.Right***)) *
  `ceil16`(***FullHeightPixel*** – (***Cropping.Top*** + ***Cropping.Bottom***)) ≤ 4,456,448
- `ceil16`(***FullWidthPixel*** – (***Cropping.Left*** + ***Cropping.Right***)) *
  `ceil16`(***FullHeightPixel*** – (***Cropping.Top*** + ***Cropping.Bottom***)) * ***FrameRate*** ≤ 267,386,880

Based on this, examples for chroma subsampled production formats that can directly be distributed are:

- Monoscopic:
  - 4096 H × 2048 V, 4:2:0, at 25, 30, 50 and 60fps with full content coverage
- Stereoscopic:
  - Each view with 4096 H × 2048 V, 4:2:0 at 25 and 30fps, if frame-packed using temporal interleaving.
  - Each view with 2048 H × 2048 V, 4:2:0, at 25, 30, 50 and 60fps, if framed packed using side-by-side.
  - Each view with 4096 H × 1024 V, 4:2:0, at 25, 30, 50 and 60fps, if framed packed using top-bottom.
  - Each view with 2944 H × 1472 V, 4:2:0, at 25, 30, 50, and 60fps, if frame packed using top-bottom.

Note: Picture sizes do not correspond necessarily to the full 360° reference if Coverage does not indicate the whole 360° sphere. They correspond to the content covered by Coverage and Cropping, i.e. sizes correspond to (***FullWidthPixel*** - (***Cropping.Left*** + ***Cropping.Right***)) * (***FullHeightPixel*** - (***Cropping.Top*** + ***Cropping.Bottom***)). Therefore, ***FullWidthPixel*** and ***FullHeightPixel*** might be greater than 4096 and 2048 respectively.

If the original video signal after chroma subsampling is beyond the constraints of HEVC Main 10 Level 5.1 constraints or the constraints dictated by the restricted scheme applied for this profile, then the HEVC-based viewport-independent OMAF video profile may be used after preprocessing of the original video content such that the constraints are fulfilled. Examples for videos that require preprocessing are

- Monoscopic:
  - 6144 H × 3072 V, 4:2:0
  - 8192 H × 4096 V, 4:2:0
- Stereoscopic:
  - For each 4096 H × 2048 V, 4:2:0, at 50 and 60fps, i.e. 8192 H × 2048 V for side-by-side and 4096 H × 4096 V for top-bottom
  - 6144 H × 3072 V, 4:2:0, i.e. 12288 H × 3072 V for side-by-side and 6144 H × 6144 V for top-bottom
  - 8192 H × 4096 V, 4:2:0, i.e. 16384 H × 4096 V for side-by-side and 8192 H × 8192 V for top-bottom

The original signal is then pre-processed, encoded and distributed following the constraints for this media profile as described in section 5.5.3.

5.5.1.2 HEVC-based viewport-dependent OMAF video profile

For the HEVC-based viewport-dependent OMAF video profile, any original video using the Equirectangular Projection (ERP) can be used as defined in section 4.1.3.2.

5.5.1.3 OMAF 3D Audio Baseline media profile

If the original audio signal is in the constraints of MPEG-H 3D Audio, LC profile, Level 3, then the OMAF 3D Audio Baseline Media Profile should be used directly on the source signal to generate elementary streams following the media profile constraints.

If the original audio signal is beyond the constraints of MPEG-H 3D Audio, LC profile, Level 3, then the OMAF 3D Audio Baseline Media Profile may be used after pre-processing of the original audio content such that the constraints are fulfilled. The pre-processed signal is then encoded and distributed the constraints for this media profile.

5.5.2 Sphere-to-Texture Mapping and SEI Message Generation

A key issue is the mapping of the spherical video to a 2D texture at the content generation and the reverse operation at the receiver. Based on the system diagram in section 4.2.2.1.2, SEI messages are added to describe the source content and the expected resulted processing from the 2D texture to a spherical video.

The mapping of the color samples of 2D texture images onto a spherical coordinate space in angular coordinates (ϕ, θ) for use in omnidirectional video applications for which the viewing perspective is from the origin looking outward toward the inside of the sphere. The spherical coordinates are defined in clause 5.1 in [OMAFFDIS].

Rotation angles yaw (α), pitch (β), and roll (γ) are also used in the specification of these semantics.

Relative to an (x, y, z) Cartesian coordinate system, yaw expresses a rotation around the z (vertical, up) axis, pitch rotates around the y (lateral, side-to-side) axis, and roll rotates around the x (back-to-front) axis. Rotations are extrinsic, i.e., around x, y, and z fixed reference axes. The angles increase clockwise when looking from the origin towards the positive end of an axis.

Assume a signal with the following parameters is provided:

- Projection is ERP
- The frame rate of the signal is provided as ***FrameRate***
- The full reference 360° video has spatial resolution ***FullWidthPixel*** times ***FullHeightPixel*** with picture aspect ratio 2:1
- The signal may follow the monoscopic or stereoscopic. If stereoscopic, the signal is provided separately per eye. The type is expressed in the ***StereoMode*** parameter
- The signal may have a restricted coverage expressed in the ***Coverage*** Parameter, if present, in the spherical domain expressed as follows:
  - ***AzimuthMin*** specifies the minimum azimuth value of the coverage sphere region in the range of −180 degrees inclusive to 180 degrees exclusive.
  - ***AzimuthMax*** specifies the maximum azimuth value of the coverage sphere region in the range of −180 degrees inclusive to 180 degrees exclusive. This value is greater than ***AzimuthMin***.
  - ***ElevationMin*** specifies the minimum elevation value of the coverage sphere region in the range of −90 to 90 degrees.
  - ***ElevationMax*** specifies the maximum elevation value of the coverage sphere region, in in the range of −90 to 90 degrees.
- The signal may have prerotation expressed in the ***Rotation*** parameter, if present, in the spherical domain expressed as follows:
  - ***RotationYaw*** specifies the value of the yaw rotation angle in the range of −180 to 180 degrees. When not present, the value is inferred to be equal to 0.
  - ***RotationPitch*** specifies the value of the pitch rotation angle in the range of −90 to 90 degrees. When not present, the value is inferred to be equal to 0.
  - ***RotationRoll*** specifies the value of the roll rotation angle in the range of −180 to 180 degrees. When not present, the value is inferred to be equal to 0.

- If the full signal is not provided but a cropped version of it is, then this is expressed by the ***Cropping*** Parameter with the four following values
  - ***Top***: the number of pixel cropped by on the top compared to the full pixel height.
  - ***Right***: the number of pixel cropped by on the right compared to the full pixel height.
  - ***Bottom***: the number of pixel cropped by on the bottom compared to the full pixel height.
  - ***Left***: the number of pixel cropped by on the left compared to the full pixel height.
- The provided image sequence therefore has a luma component with
  - Width being ***FullWidthPixel*** - (***Cropping.Left*** + ***Cropping.Right***)
  - Height being ***FullHeightPixel*** - (***Cropping.Top*** + ***Cropping.Bottom***)
  - Note that the Cropping parameter should be chosen such that all pixels that are in coverage are included in the image.

The local projected sphere coordinates (ϕ, θ) for the sample location for the center point of a sample location (i, j) is derived following clause 5.2.2 in [OMAFFDIS] for monoscopic or each of the views for ***StereoMode*** separately, invoked with ***FullWidthPixel***, ***FullHeightPixel***, ***Cropping.Left*** + i and ***Cropping.Top*** +j as inputs.

If the ***Rotation*** parameter is not present, then the global projected sphere coordinates (ϕ', θ') for the sample location for the center point of a sample location (i, j) are identical to the local sphere coordinates (ϕ, θ).

If the ***Rotation*** parameter is present with parameters ***RotationYaw*** (α), ***RotationPitch*** (β), ***RotationRoll*** (γ) - all in units of degrees - then the global projected sphere coordinates (ϕ', θ') for the sample location for the center point of a sample location (i, j) are derived based on its the local sphere coordinates (ϕ, θ).

The above content parameters may be mapped directly to the encoded signal or a preprocessing needs to be applied such that the above parameters are adjusted. Without loss of generality we assume that the above parameters are now directly mapped to the relevant SEI messages.

The equirectangular projection SEI message (as defined in sections D.2.41.1 and D.3.41.1 of [ADDSEI]) provides information to enable remapping of the color samples of the output decoded pictures onto a spherical coordinate space in angular coordinates (ϕ, θ) for use in omnidirectional video applications for which the viewing perspective is from the origin looking outward toward the inside of the sphere.

The following general rules apply for SEI message generation:

- An SEI message with payload type 150 (equirectangular projection) is generated
- The `erp_cancel_flag` is set to 0
- The `erp_persistence_flag` is set to 1

When the video provides full 360° coverage and no ***Padding*** parameter is present, then the `erp_padding_flag` is set to 0 and no region-wise packing SEI message is present.

When the video provides full 360° coverage and ***Padding*** parameter is present, then the following applies:

- region-wise packing SEI messages (as defined in sections D.2.41.4 and D.3.41.4 of [ADDSEI]) is generated in order to maximize the visible information in the encoded 2D image using the ***Padding*** information parameters as follows:
  - The `rwp_cancel_flag` is set to 0
  - The `rwp_persistence_flag` is set to 1
  - `num_packed_regions` is set to 1
  - `proj_picture_width` is set to ***FullWidthPixel***

- proj_picture_height is set to ***FullHeightPixel***
- packing_type[0] is set to 0
- proj_region_width[0] is set to ***FullWidthPixel***
- proj_region_height[0] is set to ***FullHeightPixel***
- proj_region_top[0] is set to 0
- proj_region_left[0] is set to 0
- transform_type[0] is set to 0
- packed_region_width[0] is set to ***FullWidthPixel***
- packed_region_height[0] is set to ***FullHeightPixel***
- packed_region_top[0] is set to 0
- packed_region_left[0] is set to ***Padding***

No guidance is given for parameters which are not listed above.

When the video does not provide full 360° coverage as indicated by the ***Coverage*** parameter), then

- region-wise packing SEI messages (as defined in sections D.2.41.4 and D.3.41.4 of [ADDSEI]) is generated in order to maximize the visible information in the encoded 2D image using the ***Cropping*** information parameters as follows:
  - The rwp_cancel_flag is set to 0
  - The rwp_persistence_flag is set to 1
  - num_packed_regions is set to 1
  - proj_picture_width is set to ***FullWidthPixel***
  - proj_picture_height is set to ***FullHeightPixel***
  - packing_type[0] is set to 0
  - proj_region_width[0] is set to ***FullWidthPixel*** - (***Cropping.Left*** + ***Cropping.Right***)
  - proj_region_height[0] is set to ***FullHeightPixel*** - (***Cropping.Top*** + ***Cropping.Bottom***)
  - proj_region_top[0] is set to ***Cropping.Top***
  - proj_region_left[0] is set to ***Cropping.Left***
  - transform_type[0] is set to 0
  - packed_region_width[0] is set to ***FullWidthPixel*** - (***Cropping.Left*** + ***Cropping.Right***)
  - packed_region_height[0] is set to ***FullHeightPixel*** - (***Cropping.Top*** + ***Cropping.Bottom***)
  - packed_region_top[0] is set to ***Cropping.Top***
  - packed_region_left[0] is set to ***Cropping.Left***

No guidance is given for parameters which are not listed above

When the video is stereoscopic, then the frame packing needs to be generated and an appropriate frame packing arrangement SEI message (as defined in [ADDSEI] section D.3.16) needs to be generated as follows

- An SEI message with payload type 45 is generated
- The frame_packing_arrangement_cancel_flag is set to 1
- The frame_packing_arrangement_type is set to one of the following values: 3 or 4. For more details on the choice of one of the formats, see below.
- The quincunx_sampling_flag is set to 0

Using frame-compatible plano-stereoscopic video formats means that the left-eye and right-eye images are arranged in a spatial multiplex which results in a composite image that can be treated like a conventional 2D image. Annex A of TS 101 547-2 [TS1015472] provides an informative overview of the frame compatible video formats and how a single 2D image can be generated if `frame_packing_arrangement_type` with a value of 3 or 4 is in use.

5.5.3 Encoding and Content Preparation

5.5.3.1 HEVC-based viewport-independent OMAF video profile

If the original chroma-subsampled video is in the constraints of the encoder, then it may be distributed directly.

If the original chroma-subsampled video signal is beyond the constraints of HEVC Main 10 Level 5.1, then the original video content needs to be adapted to be encoded properly with an HEVC Main 10 Level 5.1 encoder and to meet the profile level constraints. Adaptation may include temporal and/or spatial subsampling.

Specifically, for stereoscopic content at 4096 H × 2048 V, 4:2:0 at 25 and 30fps using temporal interleaving frame-packing is the most suitable format for distribution.

As examples, the signals beyond the limits of the HEVC profile level constraints documented in section 5.5.1 may be preprocessed as follows:

- 4096 H × 2048 V, 4:2:0, per eye stereoscopic at 50 and 60fps may be preprocessed to
  - 2880 H × 1440 V, 4:2:0, per eye at 50 or 60 fps, if frame-packed top-and-bottom
  - 2048 H × 2048 V, 4:2:0, per eye at 50 and 60fps, if frame-packed side-by-side
  - 4096 H × 1024 V, 4:2:0, per eye at 50 and 60fps, if frame-packed top-and-bottom
- 6144 H × 3072 V, 4:2:0, monoscopic at 50 and 60fps with full content coverage
  - To any of the formats that can be directly distributed.
- 6144 H × 3072 V, 4:2:0, stereoscopic with full content coverage
  - To any of the formats that can be directly distributed.
- 8192 H × 4096 V, 4:2:0, monoscopic or stereoscopic with full content coverage
  - To any of the formats that can be directly distributed.

The original or preprocessed video signal may also be restricted in coverage, i.e. only cover a subset of the full 360° sphere as indicated in the ***Coverage*** parameter.

In this case it is recommended that:

- The signal is properly rotated such that the covered area is centric
- The signaled is properly cropped such that a minimum of the non-covered area is included in the original signal.

Further types of adaptation may be applied for efficient encoding such as content pre-rotation. This is typically applied for increasing coding efficiency and can be achieved by moving the content specific high-motion regions into content regions (typically ERP equator) where motion is less distorted (compared to ERP poles) through global rotation. Rotation, if applied, is static for the entire stream and cannot be applied dynamically.

Either provided by the source signal or after adaptation and pre-processing, the content is expected to be in the constraints of the following parameters:

- For monoscopic:
  - 4096 H × 2048 V, 4:2:0, monoscopic at 25, 30, 50 and 60fps with full content coverage or more than 180 degree coverage
- For stereoscopic:
  - 4096 H × 2048 V, 4:2:0, per eye at 25 and 30fps, if frame-packed in top and bottom and sample aspect ratio 1:1 with full content coverage
  - 4096 H × 2048 V, 4:2:0, per eye at 50 and 60fps with up to 180 degree content coverage, if frame-packed in top and bottom, sample aspect ratio 1:1

According to the requirements of the OMAF profiles, SEI messages are added based on the original or pre-processed sequence to signal the used projection format, pre-rotation, region-wise packing and frame-packing arrangement SEI message are added as well, if the processing applies. For details on the SEI messages, see section 5.5.2.

Depending on the applications, the content provider should take into account regular random access points in the encoding, for example every 2 seconds.

If the content is prepared for adaptive bitrate streaming, then also the constraints from the HEVC CMAF Video Track as defined in [ISOCMAF] annex B.1 should be taken into account. Multiple quality representations may be generated by adapting the bitrate of the video. Note that each Representation is required to have the same OMAF metadata and SEI messages in order to ensure consistency when bitrate switching is applied.

Regular DASH or ABR recommendations for content encoding may be used (see DASH-IP IOP [DASHIFIOP] or CMAF Annex D [ISOCMAF]). The number of Representations per Adaptation Set as well as their encoding bitrates depend on different factors, such as encoder performance, content complexity, and distribution parameters. In the absence of other information, a first idea on suitable bitrates and their performance, the 3GPP TR26.918 [TR26918] provides some ideas (further details are provided in section 4.2.2.1.2.3). Generated Representations should be checked for perceptual quality and it is recommended to check the lowest bitrate Representations, if they still meet the perceptual quality expectations. If they don't, those may preferably not be offered to regular DASH clients as valid alternatives.

However, important to note that the profile prohibits spatial sub-sampling of Representations in one Adaptation Set to ensure that the rendering metadata is identical for all Representations in one Adaptation Set. A basic mapping is provided in Table 7, more details are provided in [OMAFFDIS].

Table 7: Mapping of SEI Message Information to OMAF Metadata

SEI Message	OMAF Metadata
`equirectangular_projection`	`ProjectedOmniVideoBox` `RegionWisePackingBox (Padding)`
`region_wise_packing`	`RegionWisePackingBox` `CoverageInformationBox (Optional)`
`frame_packing_arrangement`	`StereoVideoBox`

Otherwise, no specific aspects for VR content need to be taken into account.

Figure 13 provides an overview on the encoding process such that the spatially aligned and time-synchronized content can be prepared for distribution for this media profile.

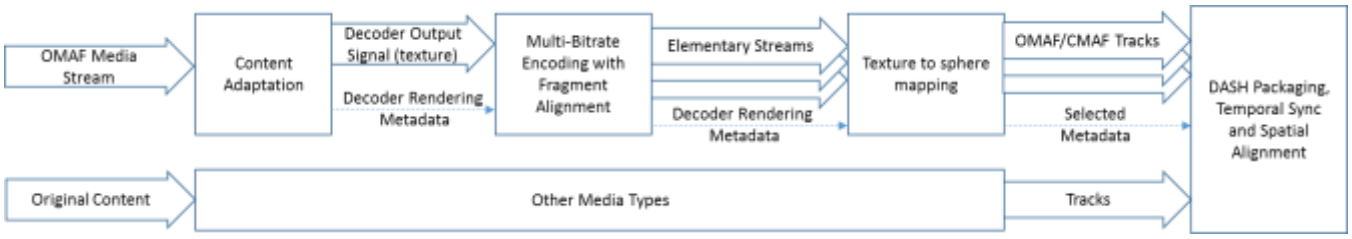

Figure 13: Content Preparation for DASH Distribution

5.5.3.2 HEVC-based viewport-dependent OMAF video profile

If the HEVC-based viewport-dependent OMAF video profile is used for distribution or download, video to be encoded may use Equirectangular Projection (ERP) or Cubemap Projection (CMP) as a projection. The ERP or CMP video may be encoded by using a Motion Constraint Tile Set (MCTS) capable HEVC Main 10 encoder.

Since the production format is only defined for ERP, if CMP is used the content needs to be converted from ERP to CMP. For guidance on the conversion between projection formats, the reader is referred to [ADDSEI] and [PROJCONV]. A related software package is available at:

`https://jvet.hhi.fraunhofer.de/svn/svn_360Lib/`

Preprocessing of the original video content may also be required to generate different versions (e.g. resolutions) of the original video that are then encoded and offered fulfilling the constraints of the HEVC-based viewport-dependent OMAF video profile as explained below.

Four main configuration parameters need to be chosen:

- Tiling granularity: number of tile columns and rows (N×M; N=Number of horizontal tiles; M=Number of vertical tiles)
- Available resolutions: number of resolutions (R=Number of resolutions) and ratios between them
- Number of Motion Constraint Tile Set for each resolution: numMCTS(r)
- Representations with a preferred viewing direction: combinations of tiles with different resolutions corresponding to a different preferred viewing direction with the combinations mixing tiles of the available resolutions (C=Number of Viewing Directions)

In order to determine the three main configuration parameters, following characteristics may be taken into account:

- Target FOV
- Resolution of ERP or CMP video before pre-processing
- Target display resolution

Table 8 summarizes the recommended tile layout patterns based on the characteristics mentioned above.

Table 8: Recommended tile layouts

Title for the tile layout scheme	Target FOV	Resolution of ERP before pre-processing	Target display resolution	Definition of scheme
6K effective ERP	Approx. 120° or less	6144 H × 3072 V or greater	2560 H × 1440 V	Clause A.6.3 of [OMAFFDIS]
6K effective cubemap	Approx. 135° or less	Greater than 6144 H × 3072 V	2560 H × 1440 V	Clause A.6.4 of [OMAFFDIS]

While the recommended schemes are defined in [OMAFFDIS], a brief summary is provided below.

For ERP, either provided by the source signal or after adaptation and pre-processing, the content is expected to be in the following versions:

- For monoscopic
  - 6144 H × 3072 V, 4:2:0, monoscopic up to 60fps
  - 3072 H × 1536 V, 4:2:0, monoscopic up to 60fps
  - 1536 H × 768 V, 4:2:0, monoscopic up to 60fps

In the 6K effective ERP scheme the content for the viewport originates from an ERP sequence of 6K resolution (6144×3072), while other parts of the content originate from either a 3K (3072×1536) version or 1.5K (1536×768) version. The polar stripes (with a value for elevation higher than 60 or lower than -60 degrees) are encoded at resolutions 1.5K and 3K, while the central part (with a value for elevation between -60 and 60 degrees)that covers an elevation range of 120° is encoded at resolutions 3K and 6K. Motion-constrained tile sets (MCTSs) are used in the encoding. The encoded MCTS sequences are combined with extractor tracks to packed pictures for 16 distinct viewing orientations, each corresponding to a selection of four spherically adjacent MCTSs from the 6K bitstream and a viewing orientation either above or below the equator. Region-wise packing metadata is included in the extractor tracks to indicate the mapping of the packed regions to the respective projected regions. Figure 14 illustrates an example for a viewing orientation above the equator. Each colored rectangle of a particular color indicates a packed region and the respective projected region. The picture size of the bitstream resolved from the extractor track is 3840×2304, which conforms to HEVC Main10 MainTier Level 5.1.

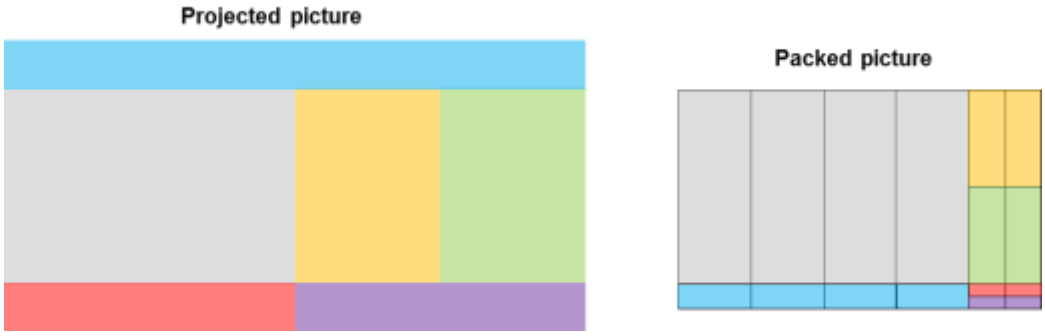

Figure 14: Example of the packed picture and the respective projected picture of one of the 16 extractor tracks, for a viewing orientation above the equator.

For CMP, the content, after adaptation and pre-processing, is expected to be in the following versions in CMP format:

- For monoscopic
  - 4608 H × 3072 V, 4:2:0, monoscopic up to 60fps
  - 2304 H × 1536 V, 4:2:0, monoscopic up to 60fps

This requires the source signal in ERP is expected to be in the following version or at higher resolution:

- For monoscopic
  - > 6144 H × 3072 V, 4:2:0, monoscopic up to 60fps

The effective 6K cubemap arrangement codes the viewport with cube faces of 1536×1536 samples, which could be considered to approximately equivalent to 6K ERP in terms of sampling density. In the arrangement, 12 tiles (encoded as MCTS) originate from the high-resolution version, and the remaining tiles (encoded as MCTS) are extracted from a cubemap having a quarter resolution compared to the high-resolution bitstream. 24 extractor tracks are created, each for different viewing orientation. The bitstreams resolved from the extractor tracks have resolution 1920×4608, which conforms to HEVC Main10 Main Tier Level 5.1.

Note: The same tiling granularity could be used if lower target display resolutions are considered with lower resolutions than the ones listed above for source and content versions.

Figure 15 illustrates the Video content preparation for DASH distribution using the HEVC-based viewport-dependent OMAF video profile.

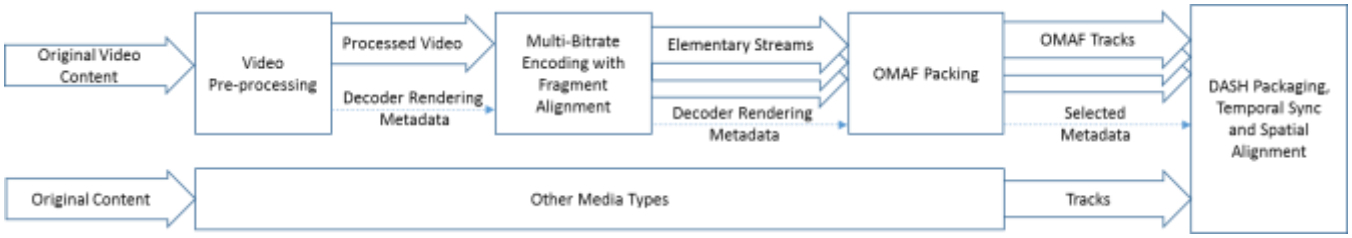

Figure 15: Video content preparation for DASH distribution with HEVC-based viewport-dependent OMAF video profile

Video pre-processing

Depending on the original video content format, various pre-processing steps may be performed before feeding the input video into the encoder:

- Projection: The input output video of the pre-processing module needs is required to be ERP or CMP projected. If the original video uses a different projection format a transformation to the targeted projection has to be applied. For further information on projection formats and conversion the reader is referred to [ADDSEI] and [PROJCONV].
- Subsampling: The input video may need to be spatially sampled to generate lower resolution variants of the input video when it is desired to offer mixed resolution video.
- Pre-rotation: For better compression efficiency, it might be beneficial to pre-rotate the input video before encoding.

The output of the video pre-processing is one video or multiple videos (of the same source) at different resolutions, which are used in the Multi-Bitrate encoding with Fragment Alignment function in order to create elementary streams with MCTS for each tile.

Multi-Bitrate and encoding with Fragment Alignment

As a result, the R pre-processed videos at different resolutions are encoded into N×M tiles (with N=number of horizontal tiles and M=number of vertical tiles), each tile being an MCTS. Each of the R pre-processed videos at different resolutions might be encoded at different bitrates, with all streams having aligned fragments, i.e. same number of samples and same Decoding Time (DT) and Composition Time (CT) for all samples.

Note: In cases where adjacent tiles are always used together the constraint of NxM tiles being encoded as MCTS can be relaxed to achieve a better coding efficiency (see for instance the polar stripes for the ERP configuration in Clause D.6.3 of [OMAFFDIS]).

SEI messages need to be added to signal the used projection format and possibly the use of frame-packing, if stereoscopic video is distributed.

The content provider should take into account that fast switching capabilities are desired for the HEVC-based viewport-dependent OMAF video profile. Therefore, it is recommended to have frequent random access points in the encoding, for example every second or less.

For DASH streaming cases, it is also possible to have an accompanying media stream with less frequent random access point (e.g. every 2 seconds), so that users that are not changing their viewport at a point in time can download a video bit stream that has a better efficiency. Thus, different elementary streams are offered with different switch point intervals. As for the different bitrate versions, elementary streams with different switch point intervals should be encoded with aligned fragments, i.e. same number of samples and same Decoding Time (DT) and Composition Time (CT) for all samples (or in other words same GOP structures). In addition, segments and subsegments of the Representations generated by these elementary should be aligned (have the same amount of the described fragments).

OMAF Packaging

After encoding, each bitstream of MCTS is encapsulated into separate ISO BMFF file that contains a single track referred to as the MCTS track. In addition, several ISO BMFF files containing the extractor tracks are generated. In the following, more details of this process are provided:

Generation of ISO BMFF files with MCTS tracks

First, each of the MCTS tiles is converted into an HEVC conforming bitstream following the MCTS sub-bitstream extraction process described in [ADDSEI]. Thereby, each MCTS track is HEVC conformant. A different `track_ID` is assigned to each MCTS track. Each MCTS is assigned a unique `mctsID` (derived as MCTS index + 1 following [ADDSEI]) assigned to it which is greater than zero and is used later for calculation of the `track_IDs`. Then, each MCTS is encapsulated into a single ISO BMFF file containing a single MCTS track with the following considerations as described in [OMAFFDIS]:

- The `track_not_intended_for_presentation_alone` flag of the `TrackHeaderBox` may be used to indicate that a track is not intended to be presented alone.
- '`povd`': indicating the projection used (e.g. ERP or CMP)
- '`covi`': indicating the coverage of the track
- '`rwpk`': indicating one region for the MCTS with its respective dimension and position within the region-wise packed frame and corresponding dimension and position in the projected frame.

These files include '`hevd`' in the compatible brands in the '`ftyp`' box. Each of the MCTS tracks covering the tiles converted into an HEVC conforming bitstream is offered with a restricted sample entry '`resv`' with an original format box '`frma`' indicating the '`hvc1`'. The `HEVCDecoderConfiguration` contains

the equirectangurlar projection SEI message or the cubemap projection SEI message that was obtained as a result of the MCTS sub-bitstream extraction process or generated at this stage. The `track_ID` of the file could be calculated using following equation:

$$trackID = tileID + r \cdot N \cdot M$$

with $r \in [0, \dots, R-1]$ indicating which resolution level they belong to , and $mctsID > 0$, so that the all `track_ID`s are different as required in the ISO BMFF specification.

Generation of ISO BMFF files with extractor tracks

Second, C tile combinations are chosen that result in a rectangular shape fulfilling the media constraints of the HEVC-based viewport-dependent OMAF video profile, each of which has a different preferred viewing direction.

Using the chosen C combinations, the files containing the extractor tracks are generated. There are C extractor track files which define different viewport configurations. Each extractor track file contains exactly one extractor track and a set of MCTS tracks TL_c, which correspond to tiles at their respective resolutions for the selected viewport combination. All tracks are added to the '`moov`' box of the extractor track file. The `track_ID` of the track containing the extractors is set to:

$$trackID = c + 1 + \sum_{i=0}^{R-1} numMCTS(i)$$

with $c \in [0, \dots, C-1]$ as a number of a specific viewport configuration. This track contains all the `track_ID`s of TL_c in the '`tref`' box (also included in the '`moov`' as mentioned).

There are as many extractors as `track_ID`s of TL_c for each sample within the extractor track; each of which is generated using an inline constructor (optional for the first extractor) and a sample constructor as defined in [NAL].

- Inline Constructor: Data carried in the extractor track (NAL and slice headers)

  The inline constructor contains the NALU and slice header of the original bitstreams with an adjusted `slice_segment_address` correctly reflecting the spatial position of the each tile within the combined video bitstream. (optional for the first extractor)

- Sample Constructor: Data referenced by the extractor track (slice payloads)

  The sample constructor references the dependent track of the corresponding MCTS and contains a `data_offset` field that allows skipping the data up to the first byte of the `slice_segment_data()`, i.e. skipping the NAL unit length and the slice header within the MCTS track. In addition, the `data_length` within the sample constructor is set to the maximum value.

These files further contain:

- '`povd`': indicating the projection used (e.g. ERP or CMP)
- '`rwpk`': indicating the regions for each of the MCTS with their respective dimensions and position within the region-wise packed frame and corresponding dimension and position in the projected frame.

As for the files containing the MCTS tracks, these files include '`hevd`' in the compatible brands in the '`ftyp`' box. The extractor tracks are offered with a restricted sample entry '`resv`' with an original format box '`frma`' indicating the '`hvc2`'. The `HEVCDecoderConfiguration` contains the equirectangurlar projection SEI message or the cubemap projection SEI message as for the whole bitstreams before the extraction

process, i.e. with full coverage. In addition, the HEVCDecoderConfiguration contains the region-wise packing SEI message. Note that the region-wise packing SEI needs to be included into the extractor tracks by the OMAF Packaging module to signal the region-wise packed picture result of combining tiles with different resolutions into a rectangular picture. The region-wise packing SEI contains the same information as the 'rwpk' box and is included to the HEVCDecoderConfiguration.

Either SphereRegionQualityRankingBox or 2DRegionQualityRankingBox should be added to the ISO BMFF file containing extractor tracks.

In case of including SphereRegionQualityRankingBox, the following applies:

- region_definition_type is equal to 1 if the projection format is ERP or equal to 0 if the projection format is equal to CMP.
- num_regions has the same value as in 'rwpk'
- quality_ranking_local_flag is set equal to 1
- quality_type is set equal to 1
- quality_ranking is set to the value r + 1, with $r \in [0, \dots, R-1]$ indicating which resolution level the region (corresponding MCTS) belongs to
- orig_width is set equal to the width of the content resolution used for encoding the region
- orig_height is set equal to the height of the content resolution used for encoding the region

In case of including 2DRegionQualityRankingBox, the following applies:

- num_regions has the same value as in 'rwpk'
- regions defined by left_offset, right_offset, top_offset and bottom_offset are aligned to the regions defined in 'rwpk'
- quality_ranking_local_flag is set equal to 1
- quality_type is set equal to 1
- quality_ranking is set to the value r + 1, with $r \in [0, \dots, R-1]$ indicating which resolution level the region (corresponding MCTS) belongs to.
- orig_width is set equal to the width of the content resolution used for encoding the region.
- orig_height is set equal to the height of the content resolution used for encoding the region.

DASH Packaging

See section 5.5.4.2.

5.5.3.3 OMAF 3D Audio Baseline media profile

If the OMAF 3D Audio Baseline Media Profile is used for distribution or download, the audio elementary stream may be generated by using a regular MPEG-H 3D Audio LC Profile, Level 3 encoder.

The content provider should take into account regular random access points in the encoding, for example every 2 seconds. Otherwise, no specific aspects for VR need to be taken into account.

If the content is prepared for adaptive bitrate streaming, then also the constraints from the MPEG-H Audio Track as defined in [ISOCMAF], Annex J should be taken into account. Multiple bitrates may be generated by adapting the bitrate of the audio.

5.5.4 Distribution

5.5.4.1 Download

5.5.4.1.1 HEVC-based viewport-independent OMAF video profile

If the HEVC-based viewport-independent OMAF video profile is used for distribution for download, the generated bitstream is included in an ISO BMFF file as one track.

The entire ISO BMFF file may be offered on an HTTP server or a CDN for download. The HTTP server should expose the capabilities of the file. No specific requirements need to be taken into account for the file format encapsulation. The `Content-Type` in the HTTP header should be set as `video/mp4 profiles='hevi' codecs='resv.pov+erpv.hvc1.1.6.L93.B0'`. If the visual component is added for a download/storage application and is generated according to the viewport-independent profile, then the entire visual component resides in a single track of the ISO file format package.

Encoding should be done in a way such that the quality is sufficiently high to minimize any visual artefacts due to encoding, under the profile/level constraints of the viewport-independent profiles.

The encoding should prioritize quality rather than bitrate or size of the file. In the encoding configuration, typical offline encoding aspects may be considered, such as variable-bitrate encoding, multi-path encoding, content-dependent encoding, sync samples to enable seek and other trick modes, etc.

Such sync samples may be added in regular distance, or if more appropriate also content dependent, for example at scene-change boundaries.

5.5.4.1.2 HEVC-based viewport-dependent OMAF video profile OMAF HEVC Tile Based Viewport-Dependent Profile

If the HEVC-based viewport-dependent OMAF video profile is used for download, one '`hvc1`' track per tile per resolution is included in the ISO BMFF file and one '`hvc2`' track per potential viewing direction is included in the ISO BMFF that follow the requirements and recommendations of the media profile in section 10.1.3.3 of [OMAFFDIS].

The entire ISO BMFF file may be offered on an HTTP server or a CDN for download. The HTTP server should expose the required capabilities to process the file. The `Content-Type` in the HTTP header should be set as

`video/mp4 profiles='hevd' codecs='resv.podv+erpv.hvc1.1.6.L93.B0'`
or `'resv.podv+ercm.hvc1.1.6.L93.B0, resv.podv+ercm.hvc2.1.6.L93.B0'`

5.5.4.2 DASH Streaming

5.5.4.2.1 HEVC-based viewport-independent OMAF video profile

If the HEVC-based viewport-independent OMAF video profile is used for distribution for DASH Streaming, regular DASH distribution means can be used. For details of DASH distribution methods, please refer to DASH-IF IOP Guidelines [DASHIFIOP].

Different types of DASH client architectures may be considered:

1) A native DASH client on an existing platform is used. In this case the MPD is handed to the playback and the DASH client is expected to handle sufficiently well downloading and playback of the

content. This example follows for example the model in smart TVs for HbbTV today or playback of a DASH Media Presentation in a video element. This model is typically referred to as type 1.

2) In a variant of this, the application still uses a native client, but has additional control interfaces to influence DASH client decisions. This may control the playback of the media (rendering, stop and resume, etc.). Such a model is provided in today in browsers with interfaces on the video element to control certain features. This model is typically referred to as type 2.
3) The DASH client is part of the application and the application optimizes the operation for this profile. Such cases require that the application has a full DASH client library. Examples for such approaches are dash.js or Shaka Player for which the DASH client part of the web page. This model requires more knowledge for the app provider on DASH operations, but also provides flexibility and optimization potentials for the app provider. In this case the APIs to the service platform are typically on codec and elementary stream level. This model is typically referred to as type 3.

Prior to more knowledge on factors impacting the performance of streaming VR content, it is recommended that a conservative approach is taken with a focus on quality of the distributed video rather than on factors such as reduced latency, fast start-up and so on.

Among others, the following is recommended:

- At startup, not necessarily the lowest bitrate should be chosen, but a bitrate that is expected to provide sufficiently good quality.
- Buffer sizes should rather be kept longer, for example in the range of several seconds to 30 seconds. This reduces the necessity for rebuffering or requiring the need to down-switch to Representations with too low quality.
- If the content is short, then an approach similar to progressive download, e.g. filling the buffer with half of the content before playback, may be preferable, buffer sizes up to 30 seconds may be considered by the DASH client to ensure high-quality playback. While this may result in longer startup delays, it is expected that this is beneficial for the user experience. Other application means may be used to enable long buffering, or content may for example be generated with low complexity in the beginning to ensure that sufficiently long buffers can be built.

5.5.4.2.2 HEVC-based viewport-dependent OMAF video profile

If the HEVC-based viewport-dependent OMAF video profile is used for tile-based DASH Streaming, an MPD file is generated with $\sum_{i=0}^{R-1} numMCTS(i)$ + C Adaptation Sets: one for each tile at each resolution $\sum_{i=0}^{R-1} numMCTS(i)$ and one for each of the C extractor tracks. This implies that each Adaptation Set contains Representations with the same resolution. Each of the $\sum_{i=0}^{R-1} numMCTS(i)$Adaptation Sets (corresponding to each tile at each resolution) can contain a Content Coverage (CC) `SupplementalProperty` element as defined in [OMAFFDIS] to signal which portion on the sphere is covered by the corresponding tile. Several Representations might be available within each Adaptation Set if each tile at a given resolution is encoded at different bitrates. These, Adaptation Sets may contain a `Preselection` descriptor as an Essential Property descriptor to indicate to which `Preselection` they belong to.

All $\sum_{i=0}^{R-1} numMCTS(i)$ + C Adaptation Sets may contain a quality ranking for each region; but at least the C Adaptation Sets with Representations corresponding to the extractor tracks contain the quality ranking indication. Either the spherical region-wise quality ranking (SRQR) `SupplementalProperty` element as defined in [OMAFFDIS] is used or the 2D region-wise quality ranking (2DQR) `SupplementalProperty` element as defined in [OMAFFDIS] is used in order to signal which tiles have higher quality in respect to other tiles. This information helps the client to identify the correct Adaptation Set (with high quality inside the desired viewport) depending on the viewing orientation of the client within an omnidirectional video. The SRQR or 2DQR descriptor is set according to the `SphereRegionQualityRankingBox` or

2DRegionQualityRankingBox as described in section 5.5.3 respectively. If SphereRegionQualityRankingBox is present in the ISOBMFF file, the value of shape_type in the CC descriptor of each of the $\sum_{i=0}^{R-1} numMCTS(i)$ Adaptation Sets for each tile at each resolution is set to the same value as region_definition_type of the SphereRegionQualityRankingBox. Values of center_azimuth, center_elevation, center_tilt, hor_range, and ver_range of the CC descriptor are set to the values of center_azimuth, center_elevation, center_tilt, hor_range, and ver_range in SphereRegionStruct of the SphereRegionQualityRankingBox. If only the 2DRegionQualityRankingBox is present in the ISOBMMF file, the value of shape_type should be set to 1 if the projection format is ERP or equal to 0 if the projection format is equal to CMP. Besides, center_azimuth, center_elevation, center_tilt, hor_range, and ver_range of the CC descriptor are computed based on left_offset, right_offset, top_offset and bottom_offset values of the 2DRegionQualityRankingBox, the RegionWisePackingBox and the sample location derivation as defined in section 5.2.1 of [OMAFFDIS].

In addition, each of the C Adaptation Sets may contain a Preselection descriptor as a Supplemental Property descriptor in order to signal which of the $\sum_{i=0}^{R-1} numMCTS(i)$ Adaptation Sets are linked to the corresponding extractor track (main media component). In addition, the C Adaptation Sets contain an Essential Property descriptor indicating that the video is packed and contains region-wise packing information.

Since the generated streams have aligned segments and subsegments, all $\sum_{i=0}^{R-1} numMCTS(i)$ + C Adaptation Sets contain the same unsigned integer value for @segmentAlignment and @subsegmentAlignment.

For VoD services, it is recommended that the content is offered at the MPD using the ISO Base Media File Format On-Demand profile: urn:mpeg:dash:profile:isoff-on-demand:2011 profile.

If low latency considerations are considered and encodings are performed with various random-access points configurations, it is recommended that the content is offered at the MPD using the ISO-Base Media File Format Broadcast TV profile: urn:mpeg:dash:profile:isoff-broadcast:2015 profile.

Note: The HEVC-based viewport-dependent OMAF video profile typically requires a low delay operation and fast switching. This requires frequent stream access points (e.g., lower than 1 second interval) to be available, which can be achieved by providing different representations with different Switching@interval values or with 'sidx' boxes having different starts_with_SAP values for each of the subsegments.

5.5.5 Security

The guidelines presented in section 4.3 apply in their entirety to Service Providers in the delivery of VR360 content.

5.5.5.1 Viewport Independent Baseline Media Profile

The Viewport Independent Media Profile is fully compatible with all commonly deployed DRM functionalities and encryption work flows. Example guidelines for the usage of DRM and security in DASH are provided in the DASH-IF interoperability guidelines [DASHIFIOP], clause 7. This provides a good overview of widely deployed adaptive streaming DRM and encryption systems which are equally applicable to 360° degree video.

5.5.5.2 Viewport Dependent Baseline Media Profile

When the DASH Access engine in the VR Service Platform performs DASH sub-segment concatenation, it will construct a single ISOBMFF file.

This file will contain encrypted data from individual DASH streams for each tile that will make up the frame, concatenated into the single ISOBMFF file.

Each sample in the frame will contain encrypted data received from the DASH stream for each tile included in that the sample.

ISOBMFF supports the definition of encryption metadata at the granularity of a sample, but not different encryption metadata within a single sample. For this reason it is necessary that the key-id and the initialization vector for each part of the sample is the same. The decryption function will decode the byte ranges indicated by the subsample information stored in the Sample Auxiliary Information within the single ISOBMFF file.

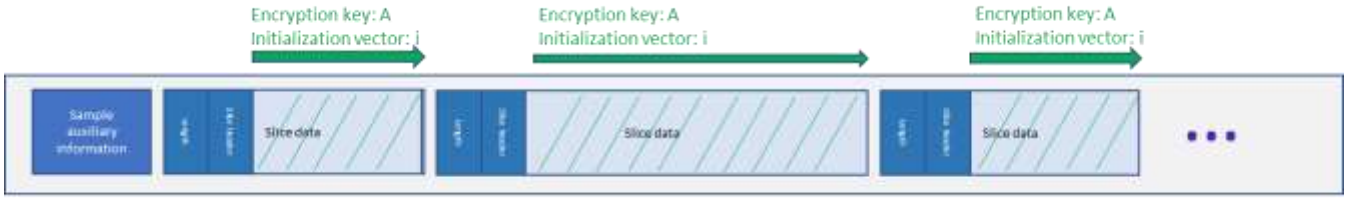

Figure 16: Logical Receiver Model

This restricts the AES encryption mode that can be used – `ctr` and `cbc1` cannot be used. For this reason the recommended encryption mode for viewport dependent media profile VR at this time is `cbcs`.

Note: This section of the guidelines remains a work in progress and further investigation is required to verify the operation of VR Players and to analyse the performance implications of this approach.
There exists proposals to support the definition of different encryption metadata for different parts of a single sample within an ISOBMFF file. This may allow the support of AES `ctr` and `cbc1` modes also in the future, again with the caveat that player support and performance implications would need to be understood.

5.6 Guidelines for Service Platform Developers

5.6.1 Overview

Figure 17 shows the basic receiver for one media component in OMAF. In case of DASH streaming, it is considered that the DASH client is part of the application and a conforming OMAF media stream is handed to the file format parser, potentially as a result from an adaptive streaming process and from a concatenation of DASH Segments/Subsegments or CMAF Fragments.

The OMAF conforming media stream is processed by the file format parser. Rendering metadata that is present in the media stream as defined for this profile is extracted and forwarded to the texture to sphere mapping. The function generates the described 3D/spherical signal. The sensor viewpoint information is then used to generate the actually rendered view. OMAF primarily describes the metadata to translate from the decoder output texture information to a 3D/spherical video.

The elementary media stream is decoded by the media decoder. The elementary stream contains the equivalent rendering metadata and may be used instead of the file format metadata as the information is available on both layers.

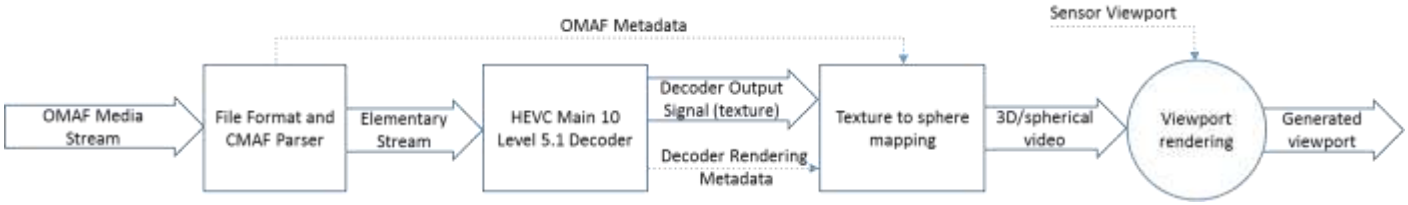

Figure 17: Logical Receiver Model

OMAF media profiles define requirements for the possible presence or absence of such rendering information in the file format and/or the in the elementary streams as well as the required receiver capabilities to process the elementary stream.

A more detailed flowchart of the logical steps of rendering chain is provided in Figure 18. The available metadata in the OMAF stream or the SEI messages is summarized as follows:

- Region-wise packing information as constraint in 7.3.1.2 in OMAF [OMAFFDIS]
- Frame-packing information
- Projection format parameters with content rotation

Based on this information a spherical content can be recovered for each eye with the information of the viewport the viewport can be generated dynamically. Additional information regarding rendering the spherical image using SEI messages can be found in both [ADDSEI] and [OMAFFDIS]

Note that basically all processing is independent of the viewport. At the same time, implementations may take into account the viewport for optimized performance to reduce processing load and power consumption.

Furthermore, implementation may combine several of the logical steps to reduce memory consumption and processing time compared to a naïve implementation of the rendering chain. For instance, it is viable to generate a polygon mesh according to the rotation of the pre-rotated content and set texture coordinates for each polygon according to the region-wise and frame-packed content at once.

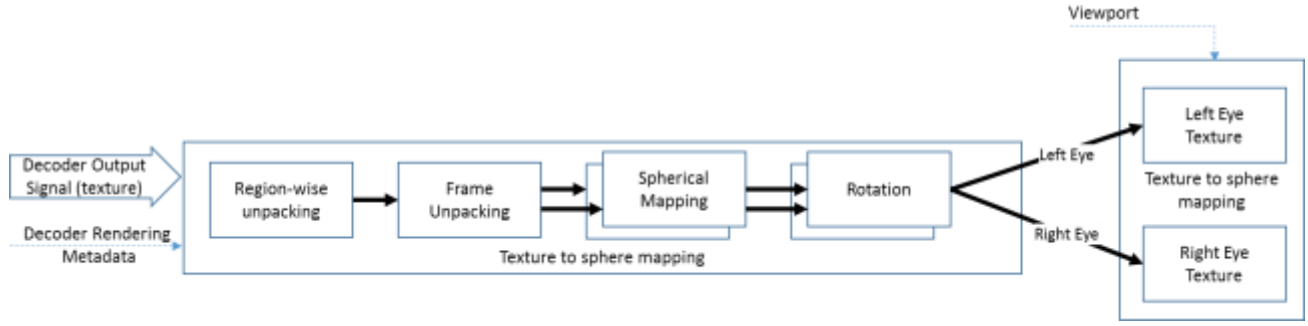

Figure 18: Rendering and viewport generation

5.6.2 Rendering Process based on SEI messages

Based on the service provider guidelines in section 5.5, the elementary stream may include SEI message that permit the 2D texture mapping to the spherical coordinate - spatially aligned – for left and right eye. In [ADDSEI] section D.3.41.6, the sample location remapping process is documented.

For the restricted scheme based for the viewport-independent media profile, to remap colour sample locations of a region-wise packed picture to a unit sphere, the following ordered (also indicated in Figure 18) steps are applied:

- If a region-wise packing SEI message is present a region-wise packed picture is obtained as the cropped output picture by decoding a coded picture. For purposes of interpretation of chroma samples, the input to the indicated remapping process is the set of decoded sample values after applying an (unspecified) upsampling conversion process to the 4:4:4 color sampling format as necessary when `chroma_format_idc` is equal to 1 (4:2:0 chroma format) or 2 (4:2:2 chroma format). This (unspecified) upsampling process should account for the relative positioning relationship between the luma and chroma samples as indicated by `chroma_sample_loc_type_top_field` and `chroma_sample_loc_type_bottom_field`, when present.
- Furthermore, the sample locations of the region-wise packed picture are mapped to sample locations of the respective projected picture as specified in [ADDSEI] section D.3.41.6.4. Note that this is a 1:1 mapping for the viewport-independent profile
- If frame packing is indicated, the sample locations of the projected picture are converted to sample locations of the respective constituent picture of the projected picture, as specified in [ADDSEI] section D.3.41.6.6. Otherwise, the constituent picture of the projected picture is identical to the projected picture.
- The sample locations of a constituent picture the projected picture are converted to sphere coordinates relative to the local coordinate axes, as specified in [ADDSEI] section D.3.41.6.2.

If rotation is indicated, the sphere coordinates relative to the local coordinate axes are converted to sphere coordinates relative to the global coordinate axes, as specified in [ADDSEI] section D.3.41.6.3. Otherwise, the global coordinate axes are identical to the local coordinate axes.

5.6.3 Distribution and Delivery

5.6.3.1 CDN Considerations

If the OMAF HEVC Tile Based Viewport-Dependent Profile is used, the amount of resources that are access via HTTP transactions is increased. Therefore, it is recommended to use HTTP/2.0 and ISO/IEC 23009-6 [DASHPUSH]. If both are supported at CDNs and servers a more efficient network performance can be achieved.

5.6.4 Security

It should be understood that for encrypted media there is no access to any decoded 360° video pixels outside of the secure media pipeline and the video bit stream should contain all the information required to recover a 360° video. Additionally, the graphics subsystem must be capable of receiving any external inputs required to produce the final display, for example orientation sensor inputs.

5.7 Guidelines for App Developers

App developers that attempt to playback content provided according to this profile on service platform are expected to have an OMAF metadata functionality included as well as APIs available to service platform. The OMAF metadata and the scheme restrictions may either be handled by the application by parsing and processing the OMAF metadata, or the application instructs the service platform to use the included SEI messages for proper rendering.

The app developer has two options:

- It checks if the rendering platform supports the usage of the SEI message. If the case, the rendering may be deferred to the rendering platform.
- If the rendering platform does not support the functionalities, the app developer may interpret the OMAF metadata to map the 2D texture output to the sphere.

Typically, the app developer needs the following functions, either from the service platform with proper APIs, or integrated into the app.

- A DASH client, unless the DASH client is part of the application, (type 3), possibly with configuration APIs to supported optimized playback and rendering.
- OMAF metadata functionality to parse and extract the relevant information or at the minimum to instruct playback in the media pipeline.
- File format parsing functionality for video playback
- Decryption module, if the content is encrypted
- HEVC video decoder to decode the video content
- Rendering and GPU functionalities to generate viewports
- Sensors for viewport tracking

Generally, it is preferable to use HW supported functionalities to optimize speed, latency, power consumptions and overall performance. Each of those above functions may be accessed with APIs. Specific APIs, possibly supported on SDKs and media frameworks are currently under development for example in Khronos, CTA WAVE or W3C.

5.7.1 Distribution and delivery

5.7.2 Decoding and Rendering

If the HEVC-based viewport-independent OMAF video profile is used, an HEVC Main 10 profile, Main tier, Level 5.1 capable decoder is needed. In addition, specific Metadata needs to be present to perform the inverse projection/rendering function on the receiver side. This metadata is either carried as OMAF Metadata in Fileformat signaling or as Decoder Rendering Metadata as SEI messages within the elementary stream.

If the HEVC-based viewport-dependent OMAF video profile is used, an HEVC Main 10 profile, Main tier, Level 5.1 capable decoder is needed. In addition, specific Metadata needs to be present to perform the inverse projection/rendering function on the receiver side. This metadata is carried as OMAF Metadata in Fileformat signaling and as Decoder Rendering Metadata as SEI messages within the elementary stream as detailed below in 5.7.3. Depending on the implementation of the renderer either one or the other metadata can be used.

5.7.3 APIs

Figure 19 shows an OMAF-DASH Client model for illustration.

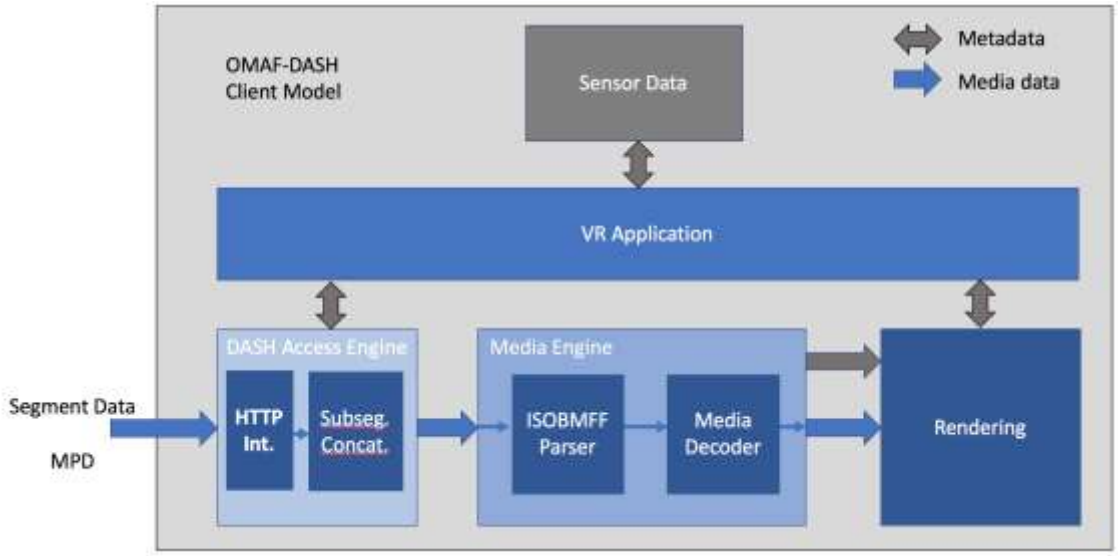

Figure 19: OMAF-DASH Client model with interfaces

In the following each sub module and the associated interfaces are described:

DASH Access Engine

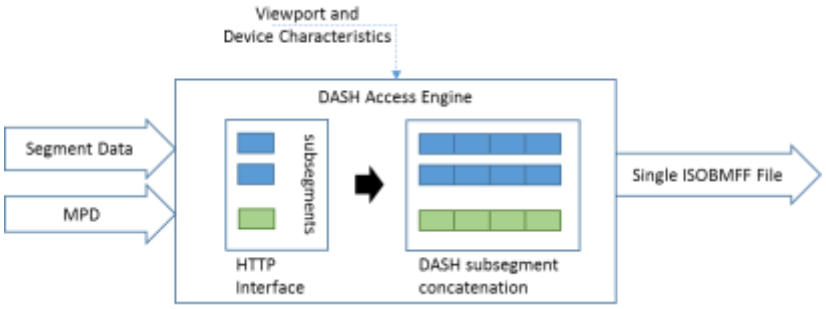

Figure 20: DASH Access Engine for HEVC-based viewport-dependent OMAF video profile

The DASH Access Engine is responsible for downloading of all OMAF media streams as well as for the fetching of the DASH manifest file. All network related aspects with respect to HTTP streaming (HTTP version, ISO/IEC 23009-6 [DASHPUSH] aspects, rate adaptation, buffering, etc.) are considered inside this module.

DASH Access Engine is connected to a VR Application module, which constantly provides the information on the selected viewport-dependent adaptation sets for the next requests.

When `@dependencyId` is used, the initialization segment of the Representation corresponding to '`hvc2`' and subsegments of the dependent Representations and complementary Representations in the order as indicated by `@dependencyId` and increasing presentation order are concatenated (i.e. as specified in section 5.3.5.1 of ISO/IEC 23009-1 [DASH]). When `Preselection` is used, the initialization segment of the Representation corresponding to '`hvc2`' can be concatenated with subsegments of the component of the `Preselection` in any order. The results leads to an ISOMBFF file conforming to the constraints defined in section 10.1.3.3 of OMAF [OMAFFDIS] which correspond to the HEVC-based viewport-dependent OMAF video profile.

Since each of the frames consist of multiple tiles that together depict the whole covered scene (with a quality/resolution emphasis on a selectable viewport), obviously, all tiles need to be received before the frame can be decoded. Therefore, the subsegments of all tiles and subsegments corresponding of an extractor track are concatenated. It is also important to mention that the bitrate of each of the tiles is much lower than that of the entire video stream, so downloading a segment for each of the tiles is comparable to downloading the equivalent sub portion of a segment when using the independent profile.

Media Engine

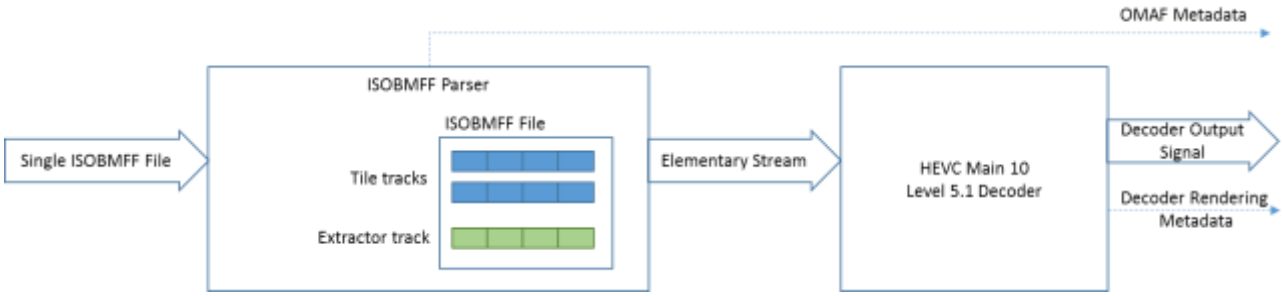

Figure 21: DASH Media Engine for HEVC-based viewport-dependent OMAF video profile

The input to the Media Engine is a single ISO BMFF file, which is the output of the DASH access engine (see Figure 20). By usage of HEVC tiles and ISO BMFF Extractor functionality, the ISO BMFF Parser generates a single HEVC Main 10 Level 5.1 elementary stream compliant to video decoders available in the market. The ISO BMFF Extractor is specified in [NAL] and contains all aggregation rules for reconstruction of a single HEVC Main 10 Level 5.1 elementary stream from multiple MCTS tracks. A detailed overview of the Extractor functionality specified in [NAL] and how they are resolved is given in Annex B.

The output signal, i.e. the decoded picture, is then rendered using either the metadata contained in the File format container or using the SEI messages contained in the video elementary streams.

Renderer

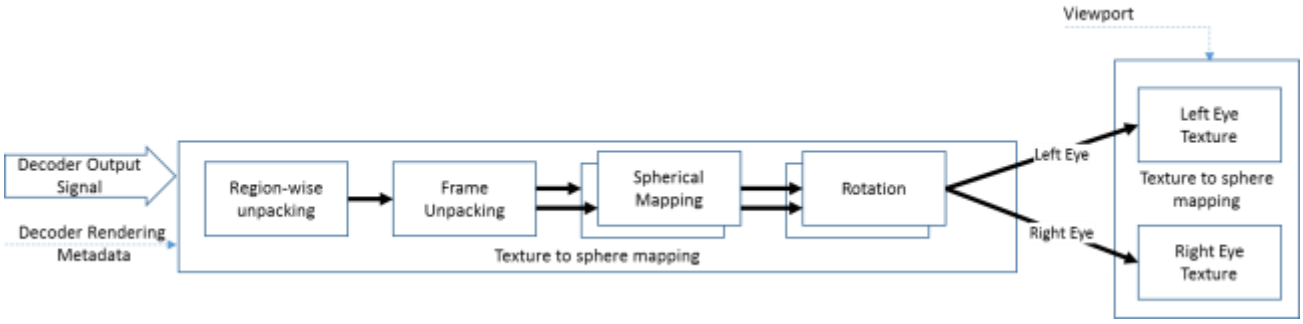

Figure 22: Inverse Projection/Renderer

A more detailed flowchart of the logical steps of rendering chain is provided in Figure 22. The available metadata in the OMAF stream or the SEI messages is summarized as follows:

- Region-wise packing ("Indicates the packing format of the content"):
  - Decoder Rendering Metadata:
    - Region-wise packing SEI
  - OMAF Metadata:
    - `RegionWisePackingBox`

- Padding (“Indicates whether there is padding in the packed frame”)
  - Decoder Rendering Metadata:
    - Equirectangular projection SEI (only for ERP and simple padding) or Region-wise packing SEI
  - OMAF Metadata:
    - `RegionWisePackingBox`
- Stereoscopic frame packing (“Indicates the packing format for stereoscopic content”)
  - Decoder Rendering Metadata:
    - Frame packing arrangement SEI
  - OMAF Metadata:
    - `StereoVideoBox`
- Projection mapping (“Indicates the projection format”)
  - Decoder Rendering Metadata:
    - Equirectangular projection SEI
    - Cubemap projection SEI
  - OMAF Metadata:
    - `ProjectionFormatBox`
- Content pre-rotation (“Indicates if content is pre-rotated”)
  - Decoder Rendering Metadata:
    - Sphere rotation SEI
  - OMAF Metadata:
    - Rotation Box
- Coverage restriction (“Indicates content coverage”)
  - Decoder Rendering Metadata:
    - Region-wise packing SEI
  - OMAF Metadata:
    - Coverage Information Box and `RegionWisePackingBox`

Based on this information a spherical content can be generated dynamically for each eye with the information about the desired viewport.

Note that basically all processing at the renderer function as defined in Figure 22, is independent of the viewport. At the same time, implementations may take into account the viewport for optimized performance to reduce processing load and power consumption. In this case, the sensor data or viewport information needs to be available in the Renderer.

Furthermore, implementation may combine several of the logical steps to reduce memory consumption and processing time compared to a naïve implementation of the rendering chain. For instance, it is viable to generate a polygon mesh according to the rotation of the pre-rotated content and set texture coordinates for each polygon according to the region-wise and frame-packed content at once.

Annex A Video Master Format Metadata

A.1 Video Metadata

Table 9: Master Format metadata

Name	Description	Data Type	Parameter Space	Defaults
		Spherical Video		
Spherical	Flag indicating if the video is a spherical video	Boolean	yes	
ProjectionType	Projection type used in the video frames	Enum	equirectangular	
StereoMode	Description of stereoscopic 3D layout	Enum	mono, stereo-left, stereo-right	mono
Coverage	Coverage parameters of the video	CoverageType See section 5.5.2		Full spherical
Rotation	Rotation parameters	RotationType See section 5.5.2		none
InitialView	Initial viewing point of the projected view	InitialViewType		If omitted, elevation=0, azimuth=0, tilt=0 is assumed
		2D Video Parameters		
FullWidthPixels	Width of the reference video frame in pixels.	Integer	See section 5.5.2	
FullHeightPixels	Height of the full reference video frame in pixels.	Integer	See section 5.5.2	

Name	Description	Data Type	Parameter Space	Defaults
FrameRate	Frame Rate in frames per second	Integer	25, 30, 50, 60, 75, 90, 100, 120	
Cropping	Cropping information expressed as top, right, bottom, left	Integer, Integer, Integer, Integer	See section 5.5.2	none
Padding	Number of horizontal pixels expressing overlapping region on the left side of the frame. Used when the frame overlaps when wrapping around the equator.	Integer	See section 5.5.2	none
PictureAspectRatio	Picture aspect ratio	Integer : Integer	2:1	
ChromaFormat	Chroma format	Enum	YCbCr	
ColourSampling	Colour sampling format	Enum	4:2:2	
SampleAspectRatio	Sample aspect ratio	Integer : Integer	1:1	
BitDepth	Bit depth	Integer	10	
ColourPrimaries	Colour primaries	Enum	ITU-R BT.709: colour_primaries=1, matrix_coefficients=1	
TransferFunction	Transfer function	Integer	1: BT.709 14: SDR BT.2020	

Name	Description	Data Type	Parameter Space	Defaults
		Production Metadata		
Duration	Duration of the content	xs:duration		
TimeStamp	Epoch Time stamp when first frame was recorded	xs:dateTime		
DirectorCut	Directors cut data	DirectorCutType	Named sequence of time stamped center and tilt angles points	
Annotation				
Contact	Name, phone, email of person or organization to contact	string		
StitchingSoftware	Name and version of stitching software	string		
Copyright	Copyright information	string		
License	License information	string		

A.2 XML Schema for VR Video Master Format

XML documents containing VR Video format properties should be instantiated against the following schema. Descriptions for the use of the elements can be found above and in the main sections of these guidelines.

```
<?xml version="1.0" encoding="utf-8"?>
<schema targetNamespace="http://vr-if.org/VRVideoMetadata/1"
    elementFormDefault="qualified"
    xmlns:this="http://vr-if.org/VRVideoMetadata/1"
    xmlns="http://www.w3.org/2001/XMLSchema">

  <element name="VRVideoMetadata" type="this:VRVideoMetadata_type"/>
  <complexType name="VRVideoMetadata_type">
    <sequence>
```



```
    <!-- Spherical Video parameters-->
    <element name="Spherical" type="boolean" default="true" minOccurs="0"/>
    <element name="ProjectionType" type="this:AllowedProjections_type"
        minOccurs="0" default="equirectangular"/>
    <element name="StereoMode" type="this:StereoMode_type" minOccurs="0" default="mono"/>
    <element name="Coverage" type="this:Coverage_type" minOccurs="0"/>
    <element name="Rotation" type="this:Rotation_type" minOccurs="0"/>
    <element name="InitialView" type="this:InitialView_type"/>
    <!-- 2D Video parameters -->
    <element name="FullWidthPixels" type="unsignedInt"/>
    <element name="FullHeightPixels" type="unsignedInt"/>
    <element name="FrameRate" type="this:FrameRate_type"/>
    <element name="Cropping" type="this:Cropping_type" minOccurs="0"/>
    <element name="Padding" type="unsignedInt" minOccurs="0" default="0"/>
    <element name="PictureAspectRatio" type="this:PictureAspectRatio_type"
        minOccurs="0"/>
    <element name="ChromaFormat" type="this:ChromaFormat_type" default="YCbCr"
        minOccurs="0"/>
    <element name="ColourSampling" type="this:ColourSampling_type" default="4:2:2"
        minOccurs="0"/>
    <element name="SampleAspectRatio" type="this:SampleAspectRatio_type" minOccurs="0"/>
    <element name="BitDepth" type="this:BitDepth_type" minOccurs="0" default="10"/>
    <element name="ColourPrimaries" type="this:ColourPrimaries_type"/>
    <element name="TransferFunction" type="this:TransferFunction_type"/>
    <!-- Production metadata -->
    <element name="Duration" type="duration"/>
    <element name="TimeStamp" type="dateTime"/>
    <element name="DirectorCut" type="this:DirectorCut_type" minOccurs="0"/>
    <!-- Annotation -->
    <element name="Contact" type="this:Contact_type" minOccurs="0" maxOccurs="5"/>
    <element name="StitchingSoftware" type="string" minOccurs="0"/>
    <element name="Copyright" type="string" minOccurs="0"/>
    <element name="License" type="string"/>
  </sequence>
</complexType>

<simpleType name="AllowedProjections_type">
  <restriction base="string">
    <enumeration value="equirectangular"/>
  </restriction>
</simpleType>
<simpleType name="BitDepth_type">
  <restriction base="unsignedInt">
    <enumeration value="10"/>
  </restriction>
</simpleType>
<simpleType name="ChromaFormat_type">
  <restriction base="string">
    <enumeration value="YCbCr"/>
  </restriction>
</simpleType>
<simpleType name="ColourPrimaries_type">
  <restriction base="string">
    <enumeration value="ITU-R BT.709"/>
```



```
    </restriction>
  </simpleType>
  <simpleType name="ColourSampling_type">
    <restriction base="string">
      <enumeration value="4:2:2"/>
    </restriction>
  </simpleType>
  <complexType name="Contact_type">
    <sequence>
      <element name="Name" type="string" minOccurs="0"/>
      <element name="Email" type="string" minOccurs="0"/>
      <element name="Phone" type="string" minOccurs="0"/>
    </sequence>
  </complexType>
  <complexType name="Coverage_type">
    <sequence>
      <element name="AzimuthMin" type="this:OMAFMinusPlus180_type" minOccurs="0" />
      <element name="AzimuthMax" type="this:OMAFMinusPlus180_type" minOccurs="0" />
      <element name="ElevationMin" type="this:OMAFMinusPlus90_type" minOccurs="0" />
      <element name="ElevationMax" type="this:OMAFMinusPlus90_type" minOccurs="0" />
    </sequence>
  </complexType>
  <complexType name="Cropping_type">
    <sequence>
      <element name="Top" type="unsignedInt" minOccurs="0"/>
      <element name="Right" type="unsignedInt" minOccurs="0"/>
      <element name="Bottom" type="unsignedInt" minOccurs="0"/>
      <element name="Left" type="unsignedInt" minOccurs="0"/>
    </sequence>
  </complexType>
  <complexType name="DirectorCut_type">
    <sequence>
      <element name="description" type="string" minOccurs="0" />
      <element name="gp" type="this:DirectorCut_entry" minOccurs="0"
          maxOccurs="unbounded" />
    </sequence>
  </complexType>
  <complexType name="DirectorCut_entry">
    <attribute name="ts" type="duration" use="required"/>
    <attribute name="azi" type="this:OMAFMinusPlus180_type"/>
    <attribute name="ele" type="this:OMAFMinusPlus90_type"/>
    <attribute name="tilt" type="this:OMAFMinusPlus180_type"/>
</complexType>
  <simpleType name="FrameRate_type">
    <restriction base="unsignedInt">
      <enumeration value="25"/>
      <enumeration value="30"/>
      <enumeration value="50"/>
      <enumeration value="60"/>
      <enumeration value="75"/>
      <enumeration value="90"/>
      <enumeration value="100"/>
      <enumeration value="120"/>
    </restriction>
```



```
</simpleType>
<complexType name="InitialView_type">
  <sequence>
    <element name="InitialAzimuth" type="this:OMAFMinusPlus180_type" default="0"
        minOccurs="0" />
    <element name="InitialElevation" type="this:OMAFMinusPlus90_type" default="0"
        minOccurs="0" />
    <element name="InitialTilt" type="this:OMAFMinusPlus180_type" default="0"
        minOccurs="0" />
  </sequence>
</complexType>
<complexType name="PictureAspectRatio_type">
  <sequence>
    <element name="Width" default="2" minOccurs="0">
      <simpleType>
        <restriction base="unsignedInt">
          <enumeration value="2"/>
        </restriction>
      </simpleType>
    </element>
    <element name="Height" default="1" minOccurs="0">
      <simpleType>
        <restriction base="unsignedInt">
          <enumeration value="1"/>
        </restriction>
      </simpleType>
    </element>
 </sequence>
</complexType>
<complexType name="Rotation_type">
  <sequence>
    <element name="RotationYaw" type="this:OMAFMinusPlus180_type" default="0"
        minOccurs="0"/>
    <element name="RotationPitch" type="this:OMAFMinusPlus90_type" default="0"
        minOccurs="0"/>
    <element name="RotationRoll" type="this:OMAFMinusPlus180_type" default="0"
        minOccurs="0"/>
  </sequence>
</complexType>
<complexType name="SampleAspectRatio_type">
  <sequence>
    <element name="Width" default="1" minOccurs="0">
      <simpleType>
        <restriction base="unsignedInt">
          <enumeration value="1"/>
        </restriction>
      </simpleType>
    </element>
    <element name="Height" default="1" minOccurs="0">
      <simpleType>
        <restriction base="unsignedInt">
          <enumeration value="1"/>
        </restriction>
      </simpleType>
```



```
      </element>
    </sequence>
  </complexType>
  <simpleType name="StereoMode_type">
    <restriction base="string">
      <enumeration value="mono"/>
      <enumeration value="stereo-left"/>
      <enumeration value="stereo-right"/>
    </restriction>
  </simpleType>
  <simpleType name="TransferFunction_type">
    <restriction base="unsignedInt">
      <enumeration value="1">
        <annotation>
          <documentation>BT.709</documentation>
        </annotation>
      </enumeration>
      <enumeration value="14">
      <annotation>
        <documentation>SDR BT.2020</documentation>
      </annotation>
      </enumeration>
    </restriction>
  </simpleType>

  <simpleType name="OMAFMinusPlus180_type">
    <restriction base="float">
      <minInclusive value="-180"/>
      <maxExclusive value="180"/>
    </restriction>
  </simpleType>
  <simpleType name="OMAFMinusPlus90_type">
    <restriction base="float">
      <minInclusive value="-90"/>
      <maxInclusive value="90"/>
    </restriction>
  </simpleType>
</schema>
```


Annex B ISO BMFF Extractors (informative)

The single ISO BMFF file contains several tracks (one per tile) with an original format equal to 'hvc1' and a track with original format equal to 'hvc2', which corresponds to the tracks with extractors.

The ISO BMFF parser has to play the track with original format equal to 'hvc2'. When parsing the 'moov' box, the parser finds a non-empty 'tref' box within the 'trak' box with original format 'hvc2'. The 'tref' box contains a list of track_IDs that indicate the tracks which the 'hvc2' track depends on.

Then, the ISO BMFF parser parses (single track) fragments until it finds the ones corresponding to the 'hvc2' track. Then, it gets the samples of the 'hvc2' track by accessing the bytes indicated in the 'trun' box within the fragments. A sample of the 'hvc2' track contains extractors, i.e. NAL units that have nal_unit_type equal to 49.

When the ISO BMFF parser finds such a NAL unit it needs to resolve the extractor, i.e. it parses the body of the extractor and replaces it with the corresponding data. The data that is replaced can be either data encapsulated within the extractor, i.e. within a construct called inline constructor, or data from other tracks, i.e. data that is referenced at a construct called sample constructor.

When parsing an extractor, a parser might find zero or more inline constructor and zero or more sample constructors. An inline constructor consists of a length field and a data field (of size indicated by the length field). The parser, simply takes the data in the data field and extracts it.

A sample constructor consists of an index pointing to one of the track_IDs in the 'tref' box, a sample offset, data offset and a data length field. The sample offset indicates the difference to the decoding time of the sample in the tracks containing the extractors and the sample in the referenced track. This value is zero since in the HEVC-based viewport-dependent OMAF video profile extractors are used for tile aggregation and all have the same decoding time. The data offset indicated how many bytes of the sample of the referenced track are skipped from the start of the sample and data length indicates the number of bytes that are copied after skipping data offset bytes. Note that if the indicated data length + data offset is bigger than the length derived from the 'trun' box of the referenced track, data length is clipped to the end of the sample as indicated in the 'trun' box.

In summary, when a sample constructor is found, the 'trun' box of the track with track_ID corresponding to the track reference index is parsed and the sample with same decoding time is searched, then the bytes from the sample following data offset up to data offset + data length (clipping if bigger than the sample size) are extracted to the extracted data.

After resolving all extractors, a valid sample is obtained.

Figure 23 shows how ISO BMFF Extractors are used for a sample. As can be seen in the figure there are N Extractors in an extractor track. The first NALU in the aggregated stream is not modified and therefore the first extractor only contains a sample constructor referencing the whole NALU with the preceding length field in the referenced track. The rest of extractors contain an inline constructor responsible of prepending an appropriate length field and slice header (for the aggregated stream) and a sample constructor that fetches the slice payload from the referenced data.

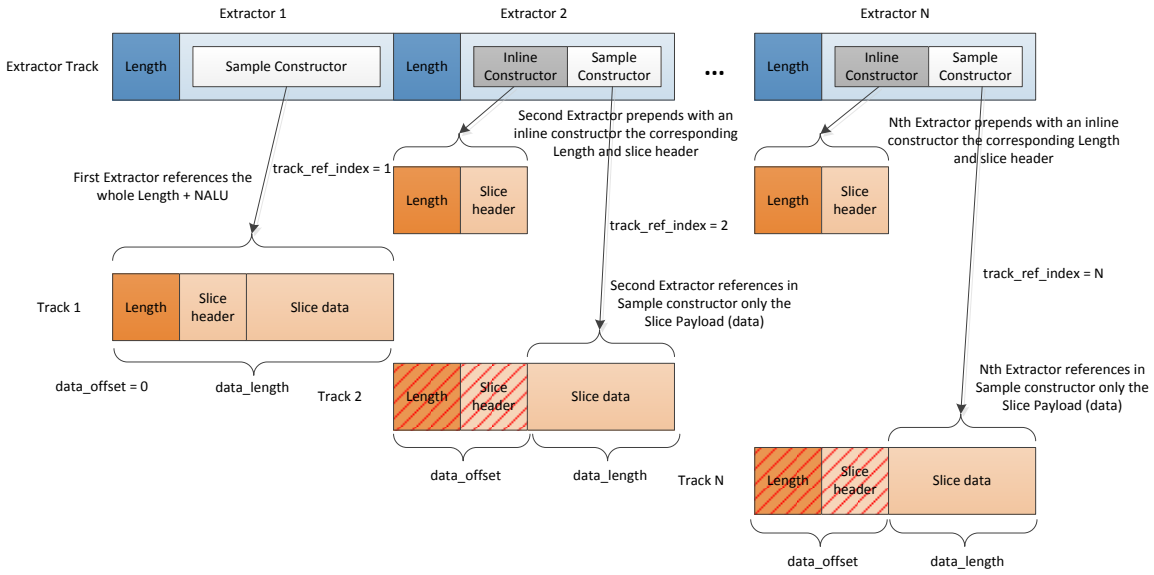

Figure 23: Single ISO BMFF File with one extractor track, N extractors and N MCTS tracks after subsegment concatenation

