Efficient Multi-Codec Support for OTT Services: HEVC/H.265 and/or AV1?

Christian Timmerer[†,‡], Martin Smole[‡], and Christopher Mueller[‡]
[‡]Bitmovin Inc., [†]Alpen-Adria-Universität
San Francisco, CA, USA and Klagenfurt, Austria, EU
[‡]{*firstname.lastname*}@bitmovin.com, [†]{*firstname.lastname*}@itec.aau.at

Abstract – *The success of HTTP adaptive streaming is undisputed and technical standards begin to converge to common formats reducing market fragmentation. However, other obstacles appear in form of multiple video codecs to be supported in the future, which calls for an efficient multi-codec support for over-the-top services. In this paper, we review the state of the art of HTTP adaptive streaming formats with respect to new services and video codecs from a deployment perspective. Our findings reveal that multi-codec support is inevitable for a successful deployment of today's and future services and applications.*

INTRODUCTION

Today's over-the-top (OTT) services account for more than 70 percent of the internet traffic and this number is expected to grow even further, i.e., 82 percent of all IP traffic will be video by 2021 (up from 73 percent in 2016) [1]. A major technical breakthrough and enabler was certainly HTTP adaptive streaming (HAS) resulting in the standardization of MPEG-DASH [2] and HLS [3]. With the emergence of MPEG's common media application format (CMAF) [4], we will witness a significant reduction of the market fragmentation as DASH and HLS converged to a single segment format based on ISOBMFF (i.e., fragmented MP4). In terms of video codecs, the market was (and still is) dominated by AVC/H.264 but we are currently entering a transition phase towards a next generation of video codecs – offering the same quality at significantly reduced data rate –, which could again lead to market fragmentation, specifically within web environments. In particular, some browsers vendors support HEVC/H.265 while others use VP9 and subsequently AV1. Entering the era of UHD, virtual reality (VR)/360-degree video, and beyond, which require new video codecs to further lower the data rate while preserving quality, we argue that OTT services are in desperate need for supporting multiple codecs in an efficient way. In this paper, we will review the state of art for HTTP adaptive streaming formats (i.e., DASH, HLS, CMAF), how UHD and VR/360-degree video is supported and discuss the issue of supporting multiple video codes taking HEVC/H.265 and VP9/AV1 as an example.

BACKGROUND: DASH, HLS, CMAF

MPEG-DASH, HLS, and CMAF share the same common principles although the respective standards have a slightly different scope. In general, the media content is provided in multiple versions (e.g., different resolutions and bitrates) and each version is divided into predefined pieces of a few seconds (typically 2-10s). A client first receives a manifest describing the available content on a server, and then, the client requests pieces based on its context (e.g., observed available bandwidth, buffer status, decoding capabilities). Thus, it is able to adapt the media presentation in a dynamic, adaptive way.

The existing different formats use slightly different terminology. Adopting DASH terminology, the versions are referred to as *representations* and pieces are called *segments*, which we will use henceforth. The major differences between these formats are shown in Table 1. We note a strong differentiation in the manifest format and it is expected that both MPEG's media presentation description (MPD) and HLS's playlist (m3u8) will coexist at least for some time. However, manifest files are typically (much) smaller than media segments and online conversion between these formats is feasible. Additionally, for closed end-to-end systems the manifest format could be anything and it is not defined within CMAF. DASH and HLS support both ISOBMFF (aka fragmented MP4) and MPEG-2 Transport Stream (TS) although historically, DASH has been essentially associated with ISOBMFF (also due to DASH-IF) and HLS only supported MPEG-2 TS in the beginning. CMAF only supports ISOBMFF and a consequence that HLS included it within its specification. In principle, DASH and HLS are agnostic to the codec/media coding format and may support any format as long as some basic characteristics are obeyed (e.g., synchronization among versions to enable adaptivity). In practice, media codecs are defined by service/application format standards or guidelines such as DASH-IF but CMAF is an application format itself and, thus, also includes media coding formats. Finally, all three formats support MPEG common encryption (CENC) but, unfortunately, with different block cipher modes (i.e., CTR vs. CBC).

TABLE 1. COMPARISON OF HAS FORMATS.

	DASH	HLS	CMAF
Manifest	MPD (xml)	Playlist (m3u8)	Not defined
Segment Format	ISOBMFF, MPEG-2 TS	MPEG-2 TS, ISOBMFF	ISOBMFF
Media Coding Format	Not defined (DASH-IF defines IOPs for common media coding formats such AVC, HEVC)	(AVC, HEVC), AAC, MP3, AC-3, Enhanced AC-3, WebVTT	AVC, HEVC, AAC, WebVTT, IMSC-1, CEA-608, CEA-708
Encryption	CENC AES-128 CTR	CENC AES-128 CBC	CENC AES-128 CTR & CBC

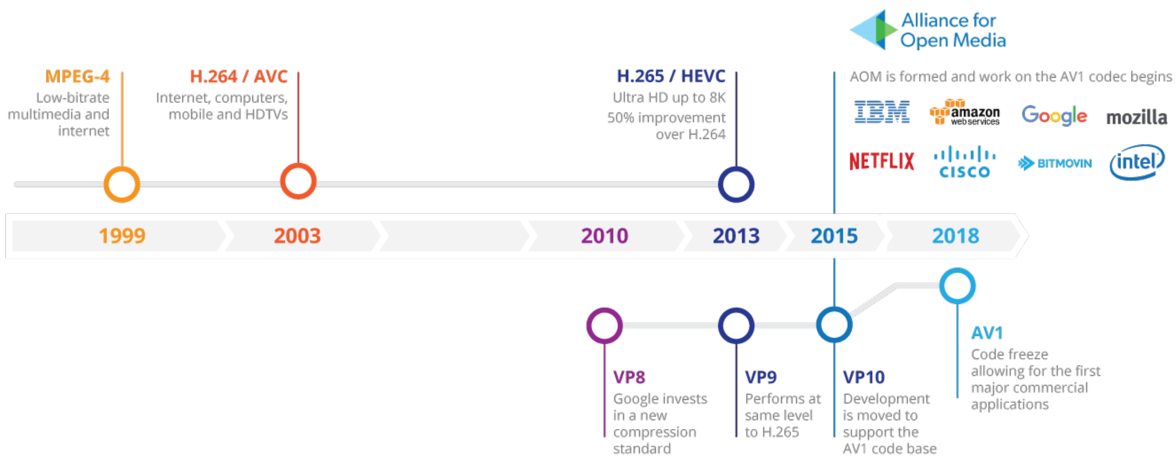

FIGURE 1. VIDEO CODEC DEVELOPMENT TIMELINE. (SOURCE: HTTPS://BITMOVIN.COM/AV1/)

MPEG is currently consolidating DASH amendments and corrigenda producing its third edition (DASH-IF is working towards v5 of its interoperability guidelines). HLS is available as informative RFC 8216 and CMAF is discussed within both DASH and HLS. Thus, we will see guidelines and/or specifications describing how to utilize CMAF within both formats in the near future.

CMAF introduces so-called *chunks* – among others – to enable low-latency HAS, which provides a finer granularity beyond segments or fragments although only the first CMAF chunk of a CMAF fragment is constrained to be an adaptive switching point (i.e., random access point). The availability of chunks, however, helps reducing latency, which has been (and maybe still is) a major issue of HAS since its infancy.

In general, however, HAS formats are deliberately codec agnostic, which allows utilizing any existing and future codecs. In principle, also other segment formats (e.g., WebM) are possible and also supported within HTML5 MSE/EME environments. State of the art and emerging video codecs are briefly reviewed in the next section.

VIDEO CODECS: AVC, HEVC, VP9/AV1, FUTURE MPEG/VCEG VIDEO CODEC

AVC/H.264 is considered to be the most widely deployed video codec and the main target was to increase coding efficiency compared to its predecessor (i.e., MPEG-2 or H.263). That is, reduce bitrate by 50 percent with the same quality or get higher quality with the same number of bits (e.g., for higher resolution like HD, adaptability to applications and networks). The target of HEVC/H.265 has been defined along the same principles (i.e., reduce bitrate by 50 percent, compare with AVC, go for UHD resolutions). Unfortunately, HEVC adoption lacks due to various reasons including but not limited to licensing issues. VP8 and VP9 has been proposed as royalty-free alternatives to AVC and HEVC respectively. Implementations of these codecs (i.e., x264, x265,

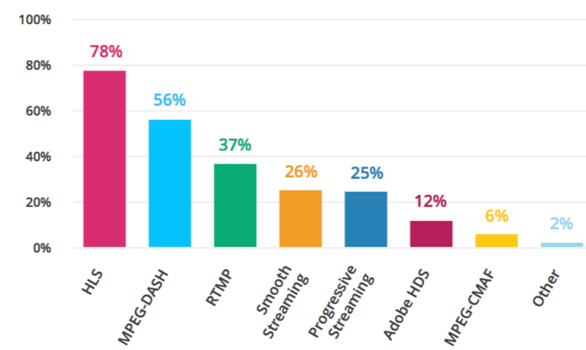

FIGURE 2. STREAMING FORMATS CURRENTLY USED BY SURVEY PARTICIPANTS (SEPTEMBER 2017) [9].

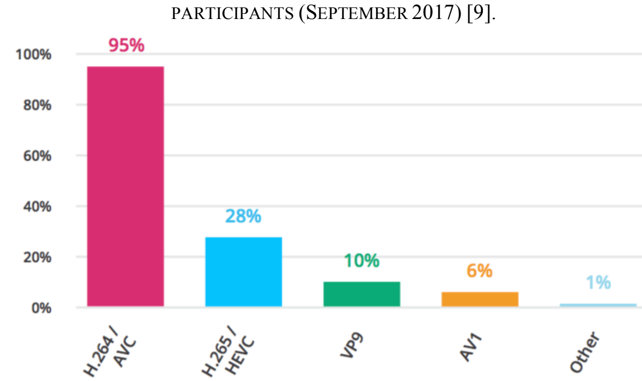

FIGURE 3. VIDEO CODING FORMATS CURRENTLY USED BY SURVEY PARTICIPANTS (SEPTEMBER 2017) [9].

libvpx) have been evaluated using a large, real-world dataset showing expectable benefits of x265 and libvpx over x264 and x265 outperforms libvpx [5].

Interestingly, both Netflix and YouTube – together responsible for more than 50 percent of the internet traffic in North America during peak hours – adopted VP9 but for different purposes. The former (Netflix) utilizes VP9 for mobile environments with low-bandwidth connections[1] whereas the latter (YouTube) uses VP9 for resolutions beyond HD [6].

[1] http://bit.ly/2DsHn3p (accessed: Jan 19, 2018)

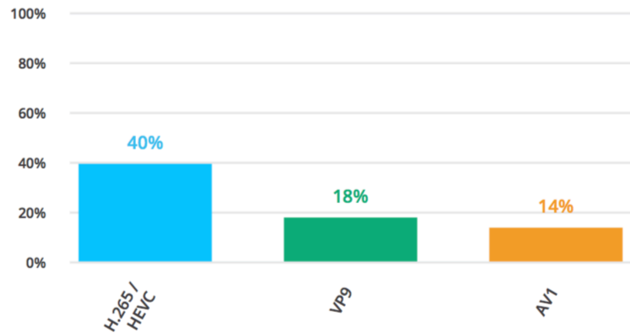

FIGURE 4. VIDEO CODING FORMATS PLANNED TO BE USED IN THE NEXT 12 MONTHS (SEPTEMBER 2017) [9].

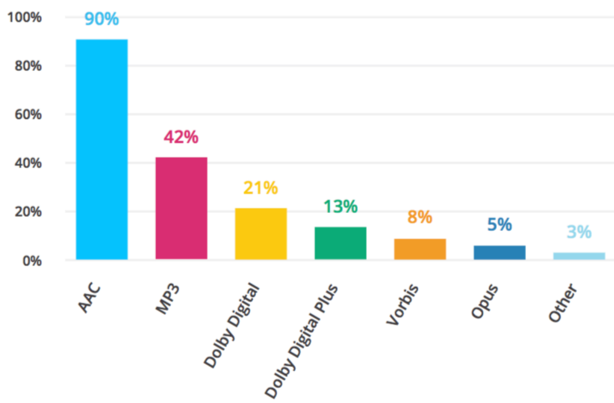

FIGURE 5. AUDIO CODING FORMATS CURRENTLY USED BY SURVEY PARTICIPANTS (SEPTEMBER 2017) [9].

Apple recently added support for HEVC to improve file efficiency, specifically targeting high resolution content. The Alliance for Open Media launched in 2015 and aims to provide an open, royalty-free video coding format referred to as AV1, which is targeting superior performance compared to HEVC. As AV1 is based on VP9 (and others like Thor and Daala), it is expected that current VP9 deployments will provide an upgrade path towards AV1 once the first official release of AV1 becomes available (at the time of writing of this paper, the first code freeze of AV1 is expected in early 2018; cf. Figure 1).

Finally, MPEG and VCEG formally established a joint collaborative team referred to as the "Joint Video Experts Team (JVET)" and issued a call for proposals (CfP) targeting 360-degree omnidirectional video, HDR/WCG, and standard dynamic range content [7]. A call for evidence has been evaluated prior to the CfP and revealed significant gains over HEVC for a considerable number of test cases with comparable subjective quality at 40–50 percent less bitrate compared to HEVC [8]. The responses to the CfP will be evaluated at the 122nd MPEG meeting in April 2018.

As a consequence, in the future we *may* have to deal with many video codecs, namely *(i)* AVC (legacy), *(ii)* HEVC, *(iii)* VP9 (legacy), *(iv)* AV1, and *(v)* future MPEG/VCEG video codec, henceforth referred to as JVET.

TABLE 2. BROWSER MARKET SHARE AND VIDEO CODEC SUPPORT. (SOURCE: NETMARKETSHARE, JANUARY 2018)

Browser	**Market share in US**	**Video codecs supported**
Google Chrome	57.50%	AVC, VP9
Mozilla Firefox	7.37	AVC, VP9
Safari	16.41	AVC, HEVC*
Microsoft Edge	2.12	AVC, HEVC, VP9
Internet Explorer	6.99	AVC
Others (~20 diff. browsers)	9.61	AVC

* only available in Safari for iOS and macOS High Sierra.

MULTI-CODEC STREAMING

As seen in the previous section, we need to support multiple codecs in an efficient way. In this context, the Bitmovin's 2017 video developer report [9] surveyed 380 video developers from over 50 countries to share their view on video delivery over the internet. The results show that HLS is still dominant, but DASH is catching up and CMAF is also taken into account (Figure 2). Additionally, video developers are more looking into DASH (31%) than HLS (24%) and CMAF (19%) within the next 12 months (figure not shown here, see here [9] for further details). AVC is still the main video codec used (Figure 3) but many video developers are looking into HEVC and some are also considering VP9 and AV1 in the near future (Figure 4). Audio coding formats are depicted in Figure 5, which shows that AAC is dominating the market with 90 percent.

In terms of browser market share and support for video codecs, we note that more than 83 percent of the devices can be reached with HEVC and VP9 and all remaining devices would fall back to AVC as shown in Table 2. When assuming a 50 percent bandwidth reduction by using HEVC or VP9, the total saving potential would be 42 percent[2].

AV1 is based on Google's VP9/VP10 codec with additional tools incorporated from Thor, Daala, and others. The reference implementation is freely available as open source but, like all reference implementations, its focus is on coding efficiency rather than runtime performance. Thus, it is currently not suitable as a standalone tool unless being optimized – which requires a lot of effort – or massively distributed within a server cluster and/or the cloud. In particular, the latter seems to be feasible and we have seen livestream demos at NAB 2017[3]. This demo shows encoding of an AV1 livestream of 1080p video sequence at 1.5 Mbps, which could potentially reduce storage and networking (CDN) costs by up to a factor of 10.

A practical quality comparison of AVC, HEVC, VP9, and AV1 (integrated within Bitmovin's encoding service) in the context of HAS reveals promising results and is highlighted in the following. Therefore, we encoded the open source movie Sintel from Blender Foundation with a fixed group of pictures (GOP) size of four seconds (corresponds

[2] https://bitmovin.com/higher-quality-lower-bandwidth-multi-codec-streaming/ (accessed: Jan 19, 2018)

[3] https://bitmovin.com/bitmovin-supports-av1-encoding-vod-live-joins-alliance-open-media/ (accessed: Jan 19, 2018)

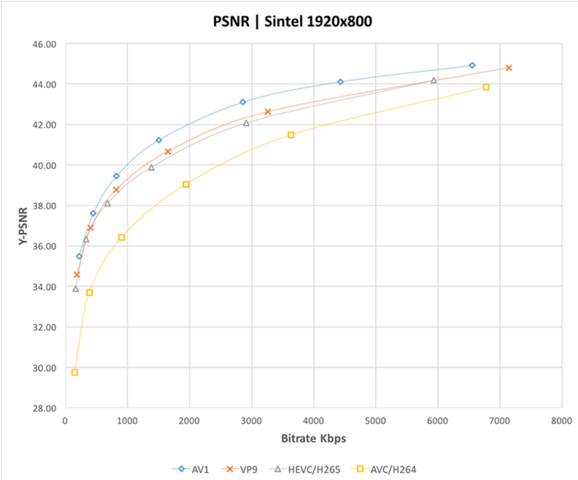

FIGURE 6. PSNR COMPARISON FOR SINTEL USING AV1, VP9, HEVC, AND AVC.

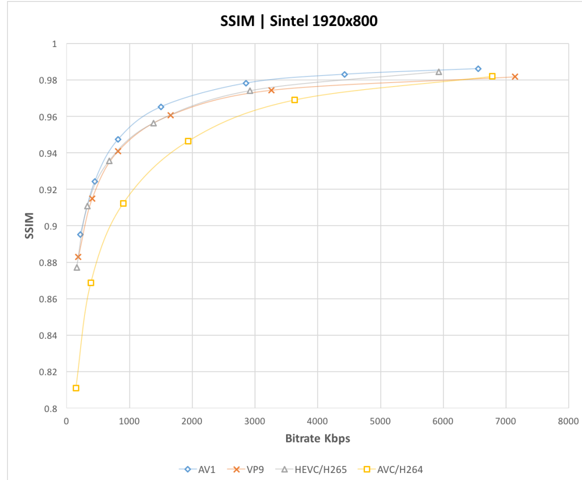

FIGURE 7. SSIM COMPARISON FOR SINTEL USING AV1, VP9, HEVC, AND AVC.

also to the segment length) and also used variable bitrate (VBR) encodings with a target bitrate as follws: 100 Kbps, 250 Kbps, 500 Kbps, 1 Mbps, 2 Mbps, 4 Mbps. We calculated PSNR and SSIM for the bitrate that has actually been achieved by the individual codec as typically codecs in VBR mode do not hit the target bitrate exactly. The following encoding settings for the different codecs were used in the Bitmovin encoding service:

- **AVC/H.264**: GOP Size: 96 frames (4 seconds), Me_range: 16, Cabac: true, B-Adapt: 2, Me: UMH, Rc-Lookahead: 50, Subme: 8, Trellis: 1, Partitions: All, BFrames: 3, ReferenceFrames: 5, Profile: High, Direct-Pred: Auto
- **HEVC/H.265**: GOP Size: 96 frames (4 seconds), Sao: 1, B-Adapt: 2, CTU: 64, Profile: Main, BFrames: 4, Rc-Lookahead: 25, WeightP: 1, MeRange: 57, Ref: 4, Subme: 3, Tu-Inter-Depth: 1, Me: 3, No-WeightB: 1, Tu-Intra-Depth: 1
- **VP9**: GOP Size: 96 frames (4 seconds), Cpu-used: 1, Tile-columns: 4, Arnr-Type: Centered, Threads: 4, Arnr-maxframes: 0, Quality: Good, Frame-Parallel: 0, AQ-Mode: none, Arnr-Strength: 3, Tile-Rows: 0
- **AV1**: Build f3477635d3d44a2448b5298255ee054fa71d7ad9, Enabled experiments by default: adapt_scan, ref_mv, filter_7bit, reference_buffer, delte_q, tile_groups, rect_tx, cdef, Passes: 1, Quality: Good, Threads: 1, Cpu-used: 1, KeyFrame-Mode: Auto, Lag-In-Frames: 25, End-Usage: VBR

Figure 6 and Figure 7 shows the results of PSNR and SSIM respectively. Please note that when we refer to codecs here, we always refer to the respective codec implementation and integration as used within the Bitmovin encoding service. All codecs used in this paper are ready to use (i.e., allow for reproducibility) except AV1, which is still under development. However, the AV1 implementation used here is clearly identified and can be retrieved from AOMedia's code repository. The results clearly show that AV1, VP9, and HEVC outperform AVC as expected. Additionally, it shows advantages for AV1 compared to HEVC and VP9; HEVC and VP9 provide comparable results. Interestingly, differences are smaller for lower bitrates but increase with an increasing bitrate starting from approximately 1 Mbps.

DISCUSSION

These first results look promising and encouraging although further tests are needed with more comprehensive datasets and subjective quality assessments in order to gain more detailed insights regarding efficiency and performance of different codecs. In practice, however, one needs to differentiate general purpose evaluations from application-specific evaluations like this one specifically targeting HAS.

AV1 targets to be 30 percent better than HEVC and JVET targets to be again 50 percent better than HEVC. The gauntlet has been thrown and the challenge will be taken but at the end of the day one may be confronted with multiple codecs offering (slightly) different coding efficiency under different conditions potentially leading to similar results with minor, insignificant differences. Consequently, deployment decisions may be based on other criteria such as licensing costs, encoder scalability, hardware support, decoder complexity, etc. which go beyond the scope of this paper.

SUPPORT FOR VIRTUAL REALITY AND 360-DEGREE VIDEO

Finally, we would like to briefly highlight current support for virtual reality (VR) applications and specifically 360-degree omnidirectional video. An overview of ongoing standardization activity is provided in [10] and adaptive streaming of such services is discussed in [11]. MPEG has recently ratified the omnidirectional media format (OMAF) [12], which is expected to be published in 2018. OMAF primarily adopts HEVC and ISOBMFF including support for region-wise packing that can be used for tiled streaming as suggested in

[11]. Current deployments still utilize AVC with equirectangular projection and HAS is basically used in the same way as for traditional video. OMAF can certainly help to increase the streaming efficiency but also increases complexity at both encoder and decoder as well as streaming and adaptation logic. Additionally, tiling or region-wise packing is not yet widely supported but this is expected to change in the near future as the VR Industry Forum[4] has recently published draft guidelines adopting OMAF. DASH and DASH-IF are exploring possible amendments of their specifications and guidelines to support OMAF, but this is still in its infancy.

CONCLUSIONS

In this paper we have reviewed the current state of the art of HTTP adaptive streaming with respect to current, emerging, and future video coding formats. We are currently entering a transition phase from AVC to a new era of video coding formats, where it is not certain that we will have one dominating video coding format as in the past. At the time of writing of this paper, we see several short term (e.g., HEVC, VP9) and long term (e.g., AV1, JVET) options. In practice, we have to deal with multiple codecs, which calls for efficient HAS solutions to provide high quality, cost-effective OTT services.

ACKNOWLEDGMENT

This work was supported in part by the Austrian Research Promotion Agency (FFG) under the Next Generation Video Streaming project "PROMETHEUS".

REFERENCES

[1] Cisco Systems, Inc., "Cisco Visual Networking Index: Forecast and Methodology, 2016–202," *White Paper*, Sep. 15, 2017. [Online]. Available: http://bit.ly/2wmdZJb

[2] I. Sodagar, "The MPEG-DASH Standard for Multimedia Streaming Over the Internet," *IEEE MultiMedia*, vol. 18, no. 4, pp. 62–67, Oct. 2011. [Online]. Available: http://dx.doi.org/10.1109/MMUL.2011.71

[3] R. Pantos and W. May, "HTTP Live Streaming," IETF RFC 8216, Aug. 2017. [Online]. Available: https://www.ietf.org/rfc/rfc8216.txt

[4] K. Hughes, D. Singer, (editors) "Information technology – Multimedia application format (MPEG-A) – Part 19: Common media application format (CMAF) for segmented media," *ISO/IEC 23000-19*, 2017.

[5] J. De Cock, A. Mavlankar, A. Moorthy, A. Aaron, "A large-scale video codec comparison of x264, x265 and libvpx for practical VOD applications," *Proc. SPIE 9971, Applications of Digital Image Processing XXXIX*, Sep. 27, 2016. doi: 10.1117/12.2238495

[6] A. Trattnig, "An Investigation of YouTube's Video Streaming Service," *Master Thesis Alpen-Adria-Universität Klagenfurt*, Dec. 2017.

[7] C. Timmerer, "MPEG Column: 120th MPEG Meeting in Macau, China," *ACM SIGMultimedia Records*, vol. 9, no. 3, Jan. 2018. doi: 10.1145/3178422.3178426

[8] C. Timmerer, "MPEG column: 119th MPEG meeting in Turin, Italy," *ACM SIGMultimedia Records*, vol. 9, no. 2, 2017. doi: 10.1145/3173058.3173061

[9] Bitmovin, Inc., "2017 Video Developer Report," September 2017. [Online]. Available: https://bitmovin.com/2017-video-developer-report/

[10] C. Timmerer, "Immersive Media Delivery: Overview of Ongoing Standardization Activities," in *IEEE Communications Standards Magazine*, vol. 1, no. 4, pp. 71-74, Dec. 2017. doi: 10.1109/MCOMSTD.2017.1700038

[11] C. Timmerer, M. Graf, C. Mueller, "Adaptive Streaming of VR/360-degree Immersive Media Services with high QoE," *in Proceedings of 2017 NAB Broadcast Engineering and IT Conference (BEITC)*, Las Vegas, NV, USA, April 2017.

[12] B. Choi, Y.-K. Wang, M. Hannuksela, Y. Lim, A. Murtaza, (editors) "Information Technology -- Coded Representation of Immersive Media (MPEG-I) -- Part 2: Omnidirectional Media Format," *ISO/IEC 23090-2 Final Draft International Standard (FDIS)*, Dec. 2017.

[4] http://www.vr-if.org/ (accessed: Jan 19, 2018)