Multiple Description-DASH: Pragmatic video streaming maximizing End-Users' Quality of Experience

Joachim Bruneau-Queyreix[1,2], Daniel Négru[2], Jordi Mongay Batalla[3], Eugen Borcoci[4]

[1]VIOTECH COMMUNICATIONS SARL
Versailles, France
jbruneauqueyreix@viotech.net

[2]LaBRI Lab. – University of Bordeaux
Talence, France
jbruneau@labri.fr
daniel.negru@labri.fr

[3]Warsaw University of Technology,
Warsaw, Poland
jordim@interfree.it

[4]University Politehnica of Bucarest, Romania
eugen.borcoci@elcom.pub.ro

***Abstract* — Over the past few years, adaptive bitrate streaming protocols, such as DASH, have risen to enhance the End-Users' experience towards video consumption over the Internet. Current methods perform a smoother playback (i.e., fewer re-buffering states) trading off with quality fluctuations due to the client-server link state. The proposed work goes a step further and introduces an innovative lightweight streaming solution by taking advantage of bandwidth aggregation over multiple paths using multiple content sources at the same time. This pragmatic evolving approach outperforms the QoE delivered by current DASH-based or P2P-based solutions and leverages the scope of action made available to the media delivery chain actors. The proposed solution has been implemented and is compared to a DASH-based approach used in most existing VoD use-cases through a first set of experiments. Results demonstrate the strong advantages in terms of quality delivered at the End-User's side and buffer occupancy. Discussion on the necessary trade-offs is also tackled.**

Keywords — streaming systems, Dynamic Adaptive Streaming over HTTP (DASH), Quality of Experience, Multiple Description Coding, Web Technologies, Content Delivery Networks

I. INTRODUCTION

With the emergence of streaming technologies characterized by the panel of different streaming services, user's Quality of Experience (QoE) has become a crucial factor for the success or failure of new applications. The quality of the multimedia streaming experience depends on several mechanisms. Among them, two major ones are the encoding/decoding chain coupled with the streaming protocols used to deliver content over networks.

First, concerning the encoding/decoding chain, the choice of the video codecs instantiated within the streaming systems may influence the way content is delivered. Although some video codecs standards have emerged [1][2][3] to cover most of the content consumption use-cases, the adoption of any uprising codec has been restrained, when not held back, thus refraining new ways of delivering and consuming content. Second, the streaming protocols try to exploit the best of networks and associated architectures in order to reach the highest possible QoE and quality of service.

Nowadays, CDN and Cloud architectures are extensively used for the large-scale video service delivery. Most of them are using HTTP-based adaptive streaming solutions (such as the recent MPEG-DASH [4] standard) where one server can deliver the content over multiple quality representations, allowing clients to select the most appropriate bitrates from the estimated network resource status. By avoiding buffer staling and re-buffering, MPEG-DASH clients are more resilient to temporary streaming server unreachability, network link failures and throughput fluctuations, providing better end-users' QoE [6][7][8][9]. However, the performances of such solutions are directly linked to the cost of overprovisioned network infrastructures. At the same time, there is a growing concern from the network operators to follow the massive demand in bandwidth utilization.

P2P/multi-source streaming solutions establish a feasible alternative since they offer a better tradeoff between the underlying network infrastructure usage and the delivery of over-the-top services [10]. However, if P2P systems achieve scalability while keeping service costs low, they also come with some limitations such as fluctuating peer upload capacity and peer departure. Multi-sources environments and DASH-based streaming do not seem well suited to work together because peer upload capacity is low and content can come from several peers during the same streaming session. In addition, it is presumably impossible to smart monitor an unknown number of changing paths at the same time, especially knowing that these paths may not be used in a short future. We show in this paper that, DASH-based solutions can effectively be integrated in multi-source environments in a way that leverages their strengths: multi-source improves upload capacity, and adaptive streaming enables a high visual quality while minimizing video interruptions. In most of existing P2P/multi-source streaming systems, only one server/peer is considered when it comes to the delivery of one video chunk. This "one link at a time per delivered chunk" paradigm definitively appears as a bottleneck. Some P2P-based streaming systems proposals [11][12][13][14] brought forward the use of innovative codecs to break the identified bottleneck, however their non-compliance with codec and protocol standards refrained their integration into existing solutions.

Due to the above-mentioned limitations, the panel of available streaming technologies could not keep on flourishing and many challenges remain partially unsolved or unaddressed: The DASH protocol's design and implementation in a multi-source environment including a simple client-sided server switching support-free of any third part; the possibility to aggregate available bandwidth in CDN-P2P architectures in order to maximize the visual quality of one video segment for a DASH client. Another critical issue in streaming is operating the low and unreliable bandwidth in

P2P environments [15][16]. In order to address such issues, business actors (Internet Service Providers -ISPs-, Content Providers –CPs-, etc.) would have to invest significant technical and financial efforts in modifying streaming platforms, which they are not necessarily keen to carry out.

The work presented in this paper proposes a Multiple-Description-DASH (MD-DASH). Thanks to its codec agnosticism, DASH-compliance, and receiver-driven philosophy, this proposal is a pragmatic and evolving solution that can be applied to many streaming architecture models (P2P, CDNs, Clouds). The essence of the proposal lies with light innovations brought at the content delivery strategy, the coding/decoding scheme and the streaming protocol in order to achieve simultaneous sub-streams retrieval and aggregate bandwidth over multiple links. In our solution, each video is encoded in multiple standard-compliant different sub-streams. When a client wishes to retrieve content, multiple sub-streams are sent over the network from multiple servers, the client combines the sub-streams and then decodes and displays the reconstructed stream with the highest possible QoE.

The paper structure is the following. Section II presents the overall MD-DASH streaming solution, highlighting the innovations and designs of (A) the content delivery strategy, (B) the video encoding/decoding chain and (C) the streaming protocol. The implementation and evaluation of the proposed streaming solution is detailed in section III along with preliminary results.. Section IV concludes and presents future works.

II. MD-DASH Streaming

The proposed lightweight client-centric streaming proposal aims at defining new ways to achieve video streaming, not only beneficiating from the DASH advantages of quality adaptation but going further by exploiting multiple sources and paths towards End-Users' QoE enhancement. In addition, the proposal is also targeting a pragmatic integration and fast adoption by the industry. In this novel streaming solution, breaking the "one link at a time per delivered chunk" bottleneck allows clients to better exploit the limited available server-side bandwidth and the dynamic network link capacity that specifically characterizes P2P systems. Splitting content into multiple "lower"-bitrate streams provides the opportunity to use and aggregate different network links' capacities. The proposed MD-DASH protocol offers a twofold scope of actions to clients: to simultaneously retrieve "independently-decodable" but "quality-aggregatable" streams (for a given content) coming from multiple servers; and to select the most appropriate server to deliver the most appropriate segment quality. Additionally, full-compatibility with existing video streaming architectures is fully ensured as the MD-DASH proposal is based on existing codecs (e.g. H264 AVC in our work) and protocol standards (DASH). Indeed, it has been designed and implemented to provide backward compatibility with already deployed DASH-based systems (e.g., dash.js)

Table 1: Comparison of CDN-based, P2P-based and MD-DASH-based approaches

Systems	DASH-based	P2P-based	MD-DASH-based
Delivery Strategy	• Multi-source • One stream at a time per chunk	• Multi-source • One/several stream(s) at a time per chunk • Tree/mesh-based	• Multi-source • Multipath • Several **parallel independent streams** at a time per chunk
Enc. process	• Standard encoder [1][2][3] • Several representations per content	• Standard encoder • Several representations per content • FEC, MDC, layered coding [11[12][13][14]	• Standard encoder • Several representations per content • Several independent streams per content. Each stream yielding refinable information
Dec. process	• Standard decoder	• Standard decoder • FEC, layered coding, MDC	• Standard decoder • Stream aggregation • Light pre-decoding step
Streaming protocols	• Bitrate adaptation [4][5][6] • Server-centric streaming source handover [20]	• Bitrate adaptation with non standard codecs [14] • Client-oriented streaming source handover [16]	• Bitrate adaptation • Multiple servers at a time per video segment • Handy user-centric adaptation and handover mechanisms • User-centric best streaming servers determination
Pros	• Close to users • High available bandwidth • Scales well	• Cost-effective • Bandwidth scalability • Network path redundancy	• Better QoE from inoperable link bandwidth aggregation • Independent video streams aggregation • Leveraging the system's scope of actions made available to the media delivery chain actors • P2P/CDN-system compliant • Light, scalable and evolving solution
Cons	•Weighty maintenance & scaling costs	• Complex bitrate adaptation • Complicated design • Unreliable • Imbalance uplink-downlink capacity	• Tunable network bandwidth overhead

The principles of the proposed streaming solution are based on three main pillars composing the contributions of this work: (A) A multiple-source multipath content delivery strategy that can be easily integrated into currently deployed architectures (CDN, Clouds, P2P). The best of networks and architectures is smoothly exploited by simultaneously sending different versions of the same content over multiple links and by having clients merging such versions (called descriptions). (B) A novel encoding/decoding chain involving multiple independent description aggregation for better end-users QoE. (C) A novel and evolving streaming protocol keeping all advantages of existing adaptive streaming protocols and bringing innovations: (1) simultaneous content retrieval from several servers, (2) smart client-side network resource utilization, (3) description adaptation mechanisms, (4) client-side simple and smooth streaming source handover, and finally (5) Intra-description switching - client-side optimal and iterative best streaming server determination. Our proposal is ready for integration and can largely modify the way in which streaming business are ran nowadays. Features brought by our solution are summed up in Table 1 with a comparison to P2P-based and CDN-based streaming systems.

A. Multiple-source and multipath content delivery strategy

Conceiving and deploying content delivery strategies relying on already-existing architectures is a strong asset for easing adoption of a proposed solution. Accordingly, our proposal is architecture-agnostic (P2P, CDNs, clouds-based). Figure 1 illustrates our content delivery strategy. The term "Box" refers as any delivering entity belonging to any CPs.

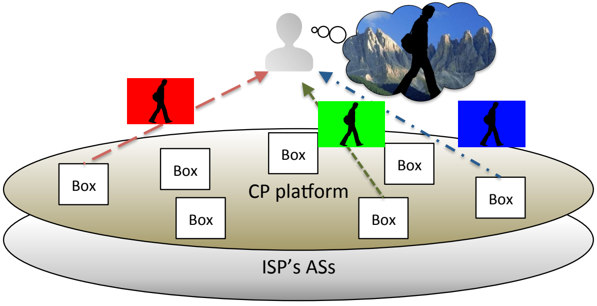


Figure 1: Content delivery strategy

Although the following proposal opens windows of collaboration between CPs and ISPs, we are not compelling these actors to modify their business. This proposal essentially keeps the current business models (i.e., minor changes in CPs and ISPs roles), however it enriches client roles with novel functionalities significantly improving the QoE. Clients achieve simultaneous concurrent stream retrievals from multiple sources over multiple paths. The clients are responsible for bitrate adaptation, seamless server-switching procedures, bandwidth aggregation and smart network resource utilization.

B. Encoding/Decoding chain of MD-DASH streaming system

The encoding/decoding scheme exploited by our streaming approach relies on the principles brought out by Multiple Description Coding (MDC) [17]. Using the MDC splitting approach is the key lever to collect and aggregate low fluctuating bandwidth. Contents are encoded in several sub-versions (namely descriptions) with lower network bandwidth requirements. Descriptions are independently decodable and yield refinable information. Each standalone decoded description provides low, yet acceptable, QoE. Part of the original video quality can be reconstructed by merging descriptions; the full video quality is obtained when all descriptions are aggregated into one single stream. Unlike layered coding, there is not any description ordering. Since each description has a lower network bandwidth requirement, it is easily sent over low capacity links. Targetting ease of adoption of this encoding/decoding chain, our contribution emphasizes the following four features:

*i. Video-standard compatibility: S*tandard compatibility allows receivers to use their standard decoder modules to decode descriptions without any change. Since there is no current MDC standard [17], this feature is of main interest for a quick adoption of this proposal. A simple pre-decoding step is required to merge descriptions, thus leaving unaltered standard compatibility.

ii. Redundancy tunability: Redundancy is referred as the common information copied into all descriptions to deliver a baseline quality. The rendered video quality is proportional to the amount of redundancy injected in all descriptions. Although an MD-based streaming protocol can be featured with major improvements (discussed in section C), a higher cumulated bitrate is required to deliver the same target decoding quality when all descriptions are used. Therefore, full-control on the amount of redundancy generated is enabled in order to match any specific use-cases.

iii. Low complexity: Due to the rapid growth of battery-powered devices, less computational complexity is still of high interest. In this paper, complexity is referred as the computational overhead compared to the standard encoding and decoding operations. It is of a major importance to define MD encoding and decoding chain minimizing such overhead.

iv. High number of different descriptions: Limiting the number of possible descriptions per content could drastically narrow the potential use-cases of our streaming systems. Even though most of the proposed MD encoders generate two descriptions only [17], having a large number of descriptions (4, 8 or even 16) provides better bitrate scalability and resilience to outages and delay variability as well as client-centric load-balancing mechanisms in multi-source environments [19].

MD Encoder: Four major MDC domains have been studied in the literature [17]: spatial, temporal, frequency, and compression. In our work, spatial and temporal domains are adopted since they allow a better codec standard-compliance. Following the work of [18][19], we propose two multiple-bitrate MD coding strategies over two scales: GOPs and frames. The MD encoding strategies support the above-described features. Our ultimate goal is to generate several descriptions at given bitrates. Although these strategies are not limited in terms of number of possible descriptions, for a matter of clearer explanations, the examples provided in this paper include three descriptions only. The MD encoding chain is presented in Figure 2.

In the first MD encoding strategy, a description is composed of a sequence of Groups Of Pictures (GOPs) whose bitrate is constantly changing over time. Although produced descriptions render periodically changing displayed quality, end-users can understand the information conveyed in the description. The second and third descriptions contain the same GOP sequence with different and complementary bitrate patterns. Such patterns should be carefully studied in order to fit best the targeted use-cases. Figure 3.a exposes the logic behind our GOP-based MD encoding strategy. The process of generating GOP-based descriptions is achieved by (1) encoding the original content over 2 (or *n*) bitrates and, (2) defining and applying an interleaving GOP-based pattern to produce the desired descriptions. This approach results in a minor additional encoding complexity compared to a regular video encoding process since the GOP-based alternating pattern is to be applied after generating two different representations of the original video. Therefore, the complexity overhead of our MD encoding strategy compared to a classic DASH approach with two quality representations can be determined by the complexity of the GOP-interleaving algorithm: $O(nd)$, where n is the number of GOP in the video sequence and d the number of generated descriptions. Such complexity is insignificant comparing to the encoding complexity of the video codec standards since the only required operations are memory copy.

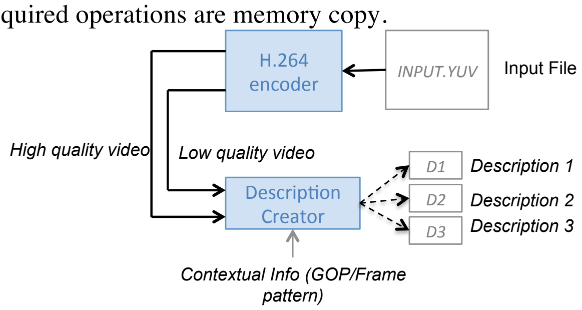


Figure 2: MD encoding chain

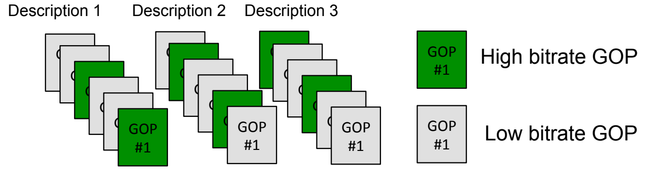


Figure 3.a: GOP-based MD scheme

The second MD encoding strategy focuses on a frame-based splitting pattern and is trying to produce a better overall rendered quality for each description, compared to the previous MD approach. Indeed, the periodically changing visual quality coming from the GOP-based splitting pattern is not present anymore and a low but more constant rendering quality is displayed from this finer-grained technique. As in the GOP-based approach, the complexity overhead of this solution can be determined by the complexity of the frame-based interleaving algorithm: $O(fd)$ where f is the number of frames in the video sequence and d is the number of generated descriptions. The additional complexity required to generate the same number of descriptions is slightly greater than in the GOP-based approach. However, it represents an insignificant overhead compared to the complexity of current video codec standards. Figure 3.b illustrates a possible frame-based MDC.

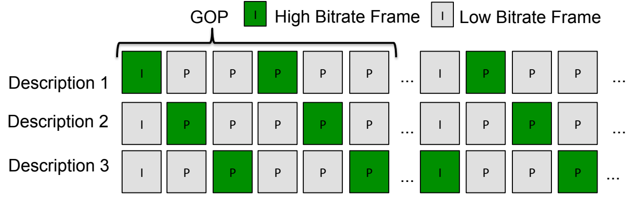


Figure 3.b: Frame-based MD scheme

Description Aggregator: According to the above-mentioned MD encoding approaches, reconstructing the best possible video sequence is achieved by selecting the GOPs/frames of most interest in each description. This operation is called the description aggregation and is performed by the Description Aggregator unit (DA). The DA is required as a pre-processing step performing a low-complexity description "merging" or selection process and generates a standard-decoder-compliant content. The DA process can be achieved over any number of descriptions in order to answer the “high number of description” feature of our MD coder. Its complexity overhead maps the one in the encoding chain since the DA’s operation is to loop on the number of frames/GOPs in the descriptions and select the most suited parts. Consequently, the complexity overhead is a linear function of the number of description multiplied by the number of frames/GOPs. Again, we are adding a small contribution on the decoding side compared to video decoding operations. The new decoding chain including the DA is outlined in Figure 4. Some contextual information (maximum number of descriptions, number of description to be merged, MD strategy -GOP/frame-, interleaving patterns, available bitrates, etc.) is provided to the DA unit, allowing the latter to be aware of the merging policy to adopt.

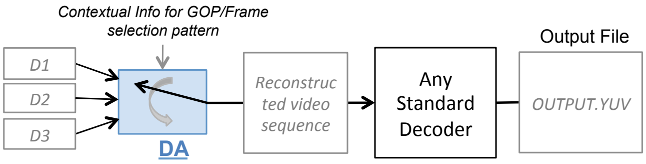

Contextual Info for GOP/Frame selection pattern
Output File
D1
D2
D3
DA
Reconstructed video sequence
Any Standard Decoder
OUTPUT.YUV

Figure 4: Decoding chain including DA

C. MD-DASH: Multiple Description - DASH

Now that the proposed video codec has given means to break the “one link at a time per delivered content” bottleneck, a relevant user-centric streaming protocol is required to cleverly utilize the available network links and available video streams. Targeting a fast adoption of our proposal and an ease of integration in currently deployed systems, we propose a DASH evolutionary streaming protocol: Multiple Description Dynamic Adaptive Streaming over HTTP (MD-DASH). This protocol strongly emphasizes client-side decisions to surpass and outperform the limitations of current streaming systems expressed in section I. A careful and conservative usage of network bandwidth is achieved by clients while extending the DASH-inherent adaptation mechanisms: (1) simultaneous description retrieval from several servers, (2) smart client-side network resource utilization, (3) description adaptation, (4) client-side simple and smooth streaming source handover procedure and, 5) client-side optimal and iterative best-streaming-server determination. Also, DASH-inherent adaptive streaming strategies and bitrate switching are maintained and performed over each simultaneously used link.

1. Simultaneous description retrieval from several servers: Several low-bandwidth requirement streams (descriptions) are simultaneously retrieved by MD-DASH clients. Should any of the sub-streams not reach client in time, the video quality will not be severely impaired before a replacement stream is delivered. This feature allows clients to be more resilient to low network link capacities or server bandwidth fluctuation as well as failures and outages. By retrieving several descriptions from several network links at a time, the probability to receive at least one stream is high and permits clients to constantly display content (thus increases the QoE), despite the low video quality. When all the descriptions forming the subset of available descriptions per content are retrieved, the highest video quality can be reconstructed and displayed.

2. Description adaptation and description combination: As in the DASH adaptive streaming strategy, MD-DASH protocol offers bitrate adaptation mechanisms. Although several different descriptions can be generated; each description (namely MD encoding pattern) can be supplied over a variety of bitrates with different interleaving pattern or with different bitrates at the GOP or slice level, thus allowing clients to adapt the bitrate of requested descriptions to the estimated bandwidth availability. Description adaptation possibilities can be structured and visualized as a matrix of possible description combinations. Table 2 exposes a set of three descriptions, each of which being provisioned over three different bitrates. Again, MD-DASH is an evolutionary DASH-based protocol and clearly offers room for opportunities such as determining the most effective number of descriptions and representations for targeted use-cases. For example in a P2P-based system, aggregating more bandwidth is achieved by merging a large number of descriptions sent over several paths. The more the descriptions and representations sent, the greater the bandwidth overhead due to redundancy. Hence, the optimum set of descriptions and representations can be determined with regard to the streaming system capabilities in order to maximize the QoE and minimize the bandwidth overhead.

Table 2: Description adaptation matrix

Representation number	Bitrate (a>b>c)	QoE (x>y>z)	Description 1	Description 2	Description 3
1	a	x	▥	▨	░
2	b	y	▥	▨	░
3	c	z	▥	▨	░

3. Smart client-side network resource utilization for better QoE: MD-DASH clients adopt a careful usage of network resource. No more resources than needed are consumed for an acceptable QoE. Based on the simultaneous description retrieval from II.C.1, MD-DASH clients use DASH bandwidth estimation to evaluate upcoming request durations, can decide to drop requests or to stop downloading descriptions from a server which bandwidth seems too low or unusable. Should a MD-DASH client receives part of all the descriptions for a video sequence, the MD-DASH client will not necessarily wait for the last descriptions to be delivered. Waiting too long could cause buffer staling and major QoE decrease. MD-DASH prioritizes a continuous video playback against the highest possible displayed video quality.

Dropping some description requests is made possible by relying on the smooth delivery of other description. Such approach permits the offload of over-loaded server and/or inoperable network links while keeping a high delivered QoE. Therefore, the MD-DASH protocol proffers elegant and pragmatic network resource utilization without any heavy third-part and complex management service support.

4. Client-side simple and smooth streaming source handover procedure: By providing a list of available servers for each description (via MPD files), clients perform a straightforward, simple and smooth handover procedure that, coupled with a "most available" server determination algorithm, could significantly enhance the QoE. While relying on the delivery of descriptions from other servers (section II.C.1), clients could achieve server switching without any temporary impact on content playback. Hence, MD-DASH protocol is a new client-side oriented protocol giving means for easy seamless handovers.

5. Intra-description switching: Client-side optimal and iterative "most available" streaming server determination: Based on II.C.1, II.C.2, II.C.3 and, II.C.4, we define a client-side optimal and iterative best streaming server determination algorithm available in the technical report of this paper [21]. This iterative intra-description switching algorithm targets a seamless and sequential lookup process to probe server's throughput and determine if the tested server can handle the delivery of a given bitrate.

III. Implementation and Evaluation

A. Implementation

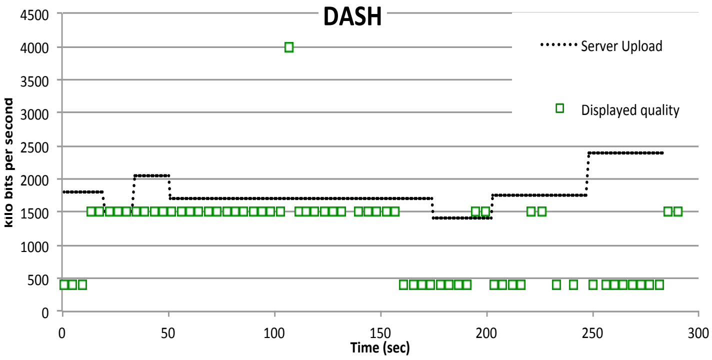
Encoder: To our knowledge, there is no existing implementation of any MD coder that can compete with existing standard codecs in terms of performance. Our work was based on the under-development C/C++ H.264 Cisco codec.

Description Aggregator unit: The DA has been implemented from scratch, and currently supports GOP-based and frame-based multiple description aggregation.

MD-DASH Protocol: The MD-DASH protocol implementation was carried out under the dash.js JavaScript framework from DASH-Industry Forum. By the time of paper submission, simultaneous description retrieval from several servers, description adaptation and combination, alongside with client-side network resource management have been fully implemented and tested. The MD-DASH protocol defines many new features that are all coming from the DASH philosophy. An extension of the existing MPD file was proposed to insert multiple description information and to grant MD-DASH clients the possibility to behave as in section II.b and II.c. The proposed MD-MPD file is available in the technical report [21]. The DA unit has been integrated in Google Chrome web browser with Google NaCL technology and is summoned by the MD-dash.js player to merge descriptions. MD-dash.js then uses the media source extension API to append merged video buffers to the browser decoder.

B. Evalutation

MD-DASH: The main advantages brought by MD-DASH streaming systems are presented in this section. Firstly, a basis of comparison is presented in Figure 8 with a DASH-based system. One HTTP server, hosting three representations of the BigBuckBunny video, was made available with an available upload bandwidth of 1800kbps and with fluctuations as in Figure 8.a. The streaming session lasted 5 minutes (x-axis) and the three representations were generated in H.264 at given bitrate following YouTube's recommendations for live streaming [23], namely: 400kbps, 1500kbps, 4000kbps. The video player dash.js was used to perform the adaptation algorithm of the DASH protocol. In order to evaluate the MD-DASH system proposal, three descriptions were generated from the video BigBuckBunny. Descriptions were separately hosted on three different servers. Each of these three servers had a mean upload bandwidth of 1800kbps in order to map the DASH evaluation methodology. The GOP-based MDC (Figure 3) was used in this experiment.



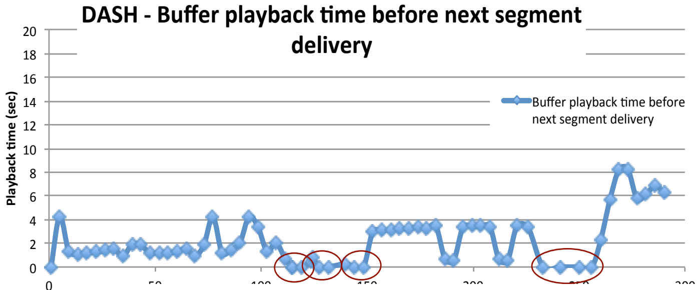
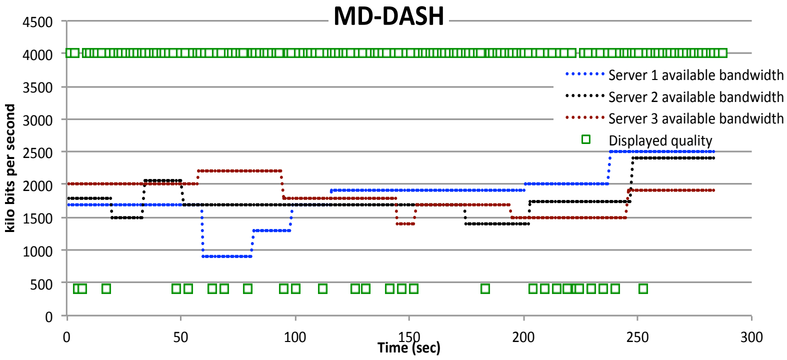


Figure 8.a: Displayed quality for DASH system
Figure 8.b: buffer playback time evolution in DASH system



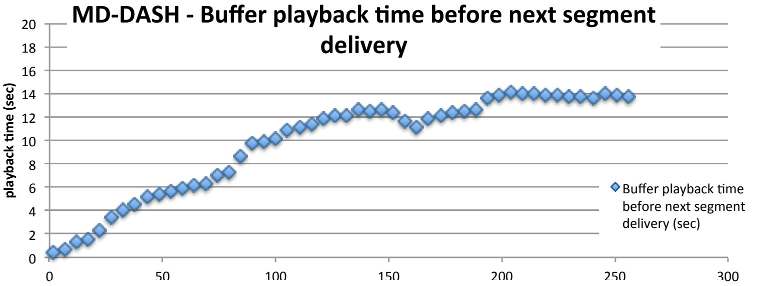


Figure. 9.a: Displayed quality for MD-DASH system
Figure. 9.b: Buffer playback time evolution in MD-DASH system

Each description has one 4000kbps-encoded GOP out of three, while the two other ones are encoded at 400kbps in H.264. Thus, the final bitrate of each description is $b = \frac{4000+400+400}{3} = 1600$. The required bandwidth to send all three description over networks is: $1600 + 1600 + 1600 = 4800kbps$. By merging all three descriptions together, a content coded in H.264 with a given bitrate of 4000kbps can be displayed. Thus, an overhead of $4800 - 4000 = 800kbps$ is required to display the top-quality content. As a matter of fact, only the most interesting GOPs are considered during the merging process, and the two other lower-quality GOPs are not even displayed. On the other hand, should 1 or 2 descriptions be missing during the description aggregation operation, then part of the displayed quality is equivalent to a H.264 4000kbps-encoded content, while the other part is equivalent to a H.264 400kbps-encoded content. In both use-cases (DASH and MD-DASH), and according to the DASH standard, representations and descriptions have been split into 4-second video segments.

On figure 8.a and 9.a, the y-axis represents bitrates of the displayed content (i.e. displayed quality) along with the available server-side bandwidths. From Figure 8.a, the average bandwidth availability over the streaming session is 1800kbps and therefore allows the streaming of representation n°1 and n°2 (1500kbps) at least. Throughout the DASH experiment (figure 8), the 1500kbps content quality was displayed for 68% of the session, the 400kbps video quality was used for 41.8% of the time, while the maximum quality (4000kbps) was streamed for 0.2% only. These quality variations are coming from the available bandwidth fluctuations that the dash.js is trying to overcome in order to avoid buffer staling events. It is important to notice representation n°3 (4000kpbs) never could be streamed due to the too low available bandwidth.

In the MD-DASH streaming system test-bed, and according to figure 9.a, the same server-side mean bandwidth is available on each server. For most of the streaming session duration, all three descriptions could be retrieved and the end-user was served with the maximum quality of 4000kbps for 81.69% of the time. The rest 18.31% of the streaming session were carried out with a 400kbps video quality. Even though, in 81.69% of the sessions, an overhead of 800kbps was used, the top quality could be displayed and could surpass the QoE delivered in the DASH evaluation. It also drastically decreases the amount of time the 400kbps quality was displayed.

Figure 8.b shows the buffer playback time prior next segment delivery. It provides a reasonable metric to evaluate buffer occupancy and re-buffering events. Several re-buffering events (framed by ellipses on Figure 8.b) are taking place in the DASH streaming system and are caused by bandwidth fluctuations coupled with the decision to increase the requested video quality. In DASH use-case, re-buffering happened 12 times with a peak of re-buffering duration of 6 seconds. In the MD-DASH use-case (Figure 9.b), even though descriptions were not all delivered in time, the MD-DASH player displayed the available merged descriptions, thus avoiding any buffer dryness and re-buffering period. The client prioritized a continuous video playback against the highest possible displayed video quality. Figure 9.b also outlines a buffer occupancy in constant increase. Indeed, the MD-DASH client always tried to play video sequences and therefore decided to drop requests which duration exceeded 95% of segment playback duration (II.C.3). An evaluation on description adaptation and combination (II.C.2) has been performed and is reported in [21]

MD Encoder: The three descriptions were generated by (1) encoding two representations of the original video (280 seconds of video) in two different bitrates (4000kbps and 400kbps) and (2) shuffling their GOPs according to the GOP-based strategy in Figure 3. Step (2) represents the overhead brought by the MD encoding chain. This step took 2.317 seconds on a Ubuntu 12.04 mono core intel i3 1.8GHz machine. The additional encoding duration is therefore low comparing to an encoding time for step (1) of 152 seconds.

Description Aggregator: The DA unit was tested on the same machine and merging three descriptions together took 2.092 seconds. Once again, the additional encoding duration is low.

IV. CONCLUSIONS

The approach presented in this paper tackles real video streaming system challenges and proposes a client-centric pragmatic and lightweight solution enhancing end-users' QoE: MD-DASH. The “one link at a time per delivered content” bottleneck is alleviated thanks to the proposal of a multi-source over multi-path content delivery strategy and the associated encoding/decoding scheme for descriptions creation. The associated novel and evolving receiver-driven DASH-based protocol (MD-DASH), along with its consumption algorithm, were designed to give means to the set-up of the elaborated strategy. A complete MD-DASH-based solution has been implemented and evaluated. The results show that a MD-DASH system outperforms a DASH system in terms of End-Users QoE, regarding the retrieval of displayed quality and buffer occupancy. However, the MD-DASH streaming system comes with the cost of a bandwidth overhead (although tunable) resulting from the use of multiple descriptions sent over multiple paths.

From a deployment point of view, both the coding/decoding chain and the MD-DASH protocol are ready for integration in any streaming architectures (P2Ps, CDNs, Cloud-based) and in real life use-cases (VoD, Live, User-Generated-Content), since they have been implemented in a pragmatic vision relying on existing software guidelines and conforming to the H.264 and MPEG-DASH standards. An on-line demonstration is available at [22] with the technical report of this work [21].

Further works will include a study on feasibility and scalability of our MD-DASH system for VoD use-cases as well as OTT-TV and UGC broadcasting events. An extensive QoE evaluation is to be carried out to determine the trade-off in using GOP vs. frame Multiple Description encoding patterns.

V. ACKNOWLEDGMENT

This work is part of the DISEDAN project within the European CHIST-ERA program, which is supported by the European Union’s Future & Emerging Technologies scheme (FET). We want to thank the other project partners for their support and contribution to the idea presented here.

REFERENCES

[1] High efficiency video coding, ITU-T Rec. H.265 and ISO/IEC 23008-2, ITU-T, April 2015.

[2] *Advanced Video Coding for Generic Audiovisual Services*, ITU-T Rec. H.264 and ISO/IEC 14496-10 (MPEG-4 AVC), ITU-T, February 2014.

[3] J. Bankoski, J. Koleszar, L. Quillio, J. Salonen, P. Wilkins, and Y. Xu, "VP8 Data Format and Decoding Guide", RFC 6386, November 2011. Available: https://www.rfc-editor.org/rfc/rfc6386.txt

[4] Information technology-Dynamic adaptive streaming over HTTP (DASH) - Part 1,Media presentation description and segment formats, ISO/IEC DIS 23009-2, May 2015.

[5] IETF, HTTP Live Streaming - Internet Dra , 2010, Available: http://tools.ietforglhtmllldra -pantos-h p-Iive-streaming-05

[6] Li, Z. Wang, J. Liu and W. Zhu, "Two decades of internet video streaming", *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 9, no. 1, pp. 1-20, 2013.

[7] S. Akhshabi, S. Narayanaswamy, A. Begen and C. Dovrolis, "An experimental evaluation of rate-adaptive video players over HTTP", *Signal Processing: Image Communication*, vol. 27, no. 4, pp. 271-287, 2012.

[8] M. Seufert, S. Egger, M. Slanina, T. Zinner, T. Hobfeld and P. Tran-Gia, "A Survey on Quality of Experience of HTTP Adaptive Streaming", *IEEE Communications Surveys & Tutorials*, vol. 17, no. 1, pp. 469-492, 2015.

[9] M. Diallo, H. Moustafa, H. Afifi and N. Marechal, "Adaptation of audiovisual contents and their delivery means", *Communications of the ACM*, vol. 56, no. 11, pp. 86-93, 2013.

[10] Y. Huang, T. Fu, D. Chiu, J. Lui and C. Huang, "Challenges, design and analysis of a large-scale p2p-vod system", *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 4, p. 375, 2008.

[11] M. Castro, P. Druschel, A. Kermarrec, A. Nandi, A. Rowstron and A. Singh, "SplitStream", *ACM SIGOPS Operating Systems Review*, vol. 37, no. 5, p. 298, 2003.

[12] S. Xie, B. Li, G. Keung and X. Zhang, "Coolstreaming: Design, Theory, and Practice", *IEEE Trans. Multimedia*, vol. 9, no. 8, pp. 1661-1671, 2007.

[13] J. Li, Y. Cui and B. Chang, "PeerStreaming: design and implementation of an on-demand distributed streaming system with digital rights management capabilities", *Multimedia Systems*, vol. 13, no. 3, pp. 173-190, 2007.

[14] Z. Liu, Y. Shen, K. Ross, S. Panwar and Y. Wang, "LayerP2P: Using Layered Video Chunks in P2P Live Streaming", *IEEE Trans. Multimedia*, vol. 11, no. 7, pp. 1340-1352, 2009.

[15] D. Jurca, J. Chakareski, J. Wagner and P. Frossard, "Enabling adaptive video streaming in P2P systems", *IEEE Commun. Mag.*, vol. 45, no. 6, pp. 108-114, 2007.

[16] J. Bruneau-Queyreix, D. Negru and J. Batalla, "Home-Box based collaborative caching strategy: An asset for Content Delivery Networks", *2014 International Conference on Telecommunications and Multimedia (TEMU)*, 2014.

[17] M. Kazemi, S. Shirmohammadi and K. Sadeghi, "A review of multiple description coding techniques for error-resilient video delivery", *Multimedia Systems*, vol. 20, no. 3, pp. 283-309, 2013.

[18] M. Lu, J. Wu, K. Peng, P. Huang, J. Yao and H. Chen, "Design and Evaluation of a P2P IPTV System for Heterogeneous Networks", *IEEE Trans. Multimedia*, vol. 9, no. 8, pp. 1568-1579, 2007.

[19] T. Tillo, E. Baccaglini and G. Olmo, "Multiple Descriptions Based on Multirate Coding for JPEG 2000 and H.264/AVC", *IEEE Transactions on Image Processing*, vol. 19, no. 7, pp. 1756-1767, 2010

[20] E. Nygren, R. Sitaraman and J. Sun, "The Akamai network", *ACM SIGOPS Operating Systems Review*, vol. 44, no. 3, p. 2, 2010.

[21] www.labri.fr/~jbruneau/technical_report_13.pdf

[22] www.labri.fr/~jbruneau/mddash/index.html

[23] https://support.google.com/youtube/answer/2853702?hl=en