



A Hybrid P2P/Multi-Server Quality-Adaptive Live-Streaming Solution Enhancing End-User's QoE

Joachim Bruneau-Queyreix, Mathias Lacaud, Daniel Négru

► To cite this version:

Joachim Bruneau-Queyreix, Mathias Lacaud, Daniel Négru. A Hybrid P2P/Multi-Server Quality-Adaptive Live-Streaming Solution Enhancing End-User's QoE. ACM MultiMedia, 2017, Mountain View, United States. hal-01584493

HAL Id: hal-01584493

<https://hal.science/hal-01584493>

Submitted on 8 Sep 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Hybrid P2P/Multi-Server Quality-Adaptive Live-Streaming Solution Enhancing End-User's QoE

Joachim Bruneau-Queyreix
Univ. Bordeaux, LaBRI, UMR 5800,
F-33400 Talence
Viotech Communications, France
joachim.bruneau-queyreix@labri.fr

Mathias Lacaud
Univ. Bordeaux, LaBRI, UMR 5800,
F-33400 Talence
Joada, Bordeaux, France
mlacaud@joada.net

Daniel Négru
Univ. Bordeaux, LaBRI, UMR 5800,
F-33400 Talence, France
daniel.negru@labri.fr

ABSTRACT

HTTP Adaptive Streaming have become the de-facto solutions to deliver video over the Internet due to their ability to enhance consumers' Quality of Experience (QoE). Nevertheless, they do not have the possibility to improve the actual delivered video quality, limited by the available client-server throughput. In comparison, multiple-server and P2P streaming offer the opportunity to obtain enhanced QoE by benefiting from expanded bandwidth, link diversity and reliability in distributed streaming infrastructures. We present a prototype for a hybrid P2P/multi-server quality-adaptive streaming solution, simultaneously using several servers and peers, and trading off the server infrastructure capacities and QoE gains.

CCS CONCEPTS

• **Information systems** → **Multimedia streaming**; • **Networks** → **Network protocols**; **Peer-to-peer protocols**;

1 INTRODUCTION

Users' Quality of Experience (QoE) has become a crucial factor for the success of video streaming services. According to Cisco [6], video traffic represents 75% of today's Internet traffic and will face a near two-fold increase by 2020. Such increases are usually not followed by the necessary upgrade of core networks capacity due to the important costs it incurs. Thus, QoE and scalability costs are major issues in the design of current and future delivery solutions.

Content Delivery Networks (CDNs) are extensively used for the large-scale delivery of live content over the Internet. Most of the time, HTTP Adaptive Streaming (HAS) techniques -such as the widely adopted DASH standard [8]- are employed to increase QoE by adjusting the desired bitrate to the observed network conditions. Hence, HAS tends to avoid video freezing events that are mainly caused by the lack of throughput at client or server side. Although CDNs solutions can handle large request volumes, they are still constrained by the scale of their infrastructures. In view of the video traffic growth forecasts, CAPEX and OPEX covered by content delivery providers will rise and eventually make these services highly pricey for content providers and consumers seeking high QoE. As an alternative to CDNs, live P2P systems solve the scalability cost issue by leveraging on consuming clients' resources. However, they also come with limitations (churn, heterogeneous capabilities, etc.) that can rapidly result in re-buffering and degraded QoE.

In order to address the issues of low QoE in P2P systems, lack of server-side throughput, and high scalability costs, we exploit the advantages of quality adaptation for hybrid P2P/multi-server live streaming. Deriving from the work [1–5], we demonstrate our

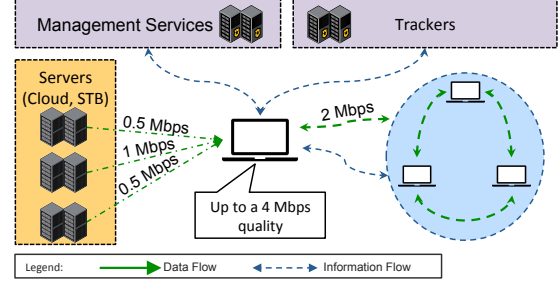


Figure 1: High level solution overview

solution for cost- and quality- adaptive hybrid P2P/multi-server live-streaming trading the system scalability and its QoE. We consider the use-case of a streaming provider delivering a live video at Q_{max} qualities to $N(t)$ clients. Clients simultaneously use multiple servers with heterogeneous upload capacities and obtain higher QoE by exploiting link diversity in this distributed streaming infrastructure. Additionally, clients leverage on the upload capacities of neighbors to engage in P2P data transfer to achieve higher scalability. Each client adjusts the requested quality with regards to P2P and client-server communications' conditions. Finally, clients locally minimize their utilization of the server infrastructure's limited throughput capacity, increasing scalability without sacrificing on QoE.

2 A HYBRID P2P/MULTI-SERVER LIVE-STREAMING SOLUTION

As shown in Fig.1, our system is composed of three major components: (1) streaming portals including management services delivering MPD files; and trackers responsible for serving peers with lists of K neighbors, and periodically computing indicators on the current health of the P2P overlay based on peer-reported metrics; (2) Servers provisioned with live video flows at multiple bitrates; (3) An application-layer mesh-pull overlay composed of $N(t)$ peers consuming the content, exchanging their buffer maps and their estimated upload rates. We also provide the functional software architectures of servers and peers, depicted in Fig. 2 (HAS functional modules are colored white). While HTTP is employed for signaling and streaming from servers, WebRTC is used for the P2P communications, and enables the contribution of peers behind NAT.

For every video segment, a client simultaneously requests several servers and peers to deliver independent codec-compliant and aggregatable sub-segments generated by the *sub-segment composer* from the *DASH storage*, which contains several bitrate representations of the same content split in segments of a few seconds.

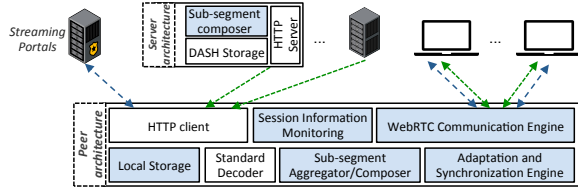


Figure 2: Functional software architectures



Figure 3: Composition example for two sub-segments

Sub-segments are generated by interleaving the Group Of Pictures (i.e. sequence of correlated frames -GoPs-) available at two different bitrates: one critically low bitrate referred as redundant bitrate, and one target bitrate. The redundant bitrate is set to such low values (e.g., 150 Kbps for a 720p resolution in H.264) in order to provide playback at the lowest possible network transfer cost. Before sending sub-segment requests, the client specifies the composition of each sub-segment in terms of number of GoPs at the target and redundant bitrate. Sub-segments can also be composed of GoPs at a null bitrate, as shown in Fig.3. The number of sub-segments (i.e., the number of used servers/peers) and their compositions are evolving parameters determined by the client according to the heterogeneous characteristics of the considered sources, and to the targeted bitrate. This sub-segment generation scheme has features of interests, easing its adoption by streaming actors: video-codec standard compatibility, tunable redundancy (i.e., the redundant and low bitrate GoPs), low additional complexity, and high flexibility in the number of possible sub-segments for a given segment, that translates into flexibility in the number of used servers/peers.

When retrieved, sub-segments are used to reconstruct the original requested content quality (in the *Sub-segment Aggregator*) by selecting the GoPs of higher size from the set of available sub-segments. In order to be resilient to network heterogeneity and to ensure uninterrupted video experience, the *synchronization engine* at client side can cancel late requests containing redundant GoPs, and can handover the delivery of late/missing GoPs to other servers to benefit from available sources. Hence, in the event of sub-segment late delivery, content is still playable (playback continuity is not affected) with a temporary lower quality on a GoP-duration basis.

Each peer embeds an *Adaptation Engine* that controls the transitions from one quality to another. By relying upon local and global indicators respectively reflecting the local streaming session status and the global delivery of each quality, peers strive to enhance QoE while preserving the efficient delivery of each quality. Finally, for each segment n the client decides on the percentage X_n of GoPs that should be delivered by the servers. The value of X_n is adjusted for every segment so as to locally minimize the server-infrastructure utilization and provide scalability without ceding on the QoE gains.

3 RESULTS

Our hybrid P2P/multi-server live streaming protocol has been implemented in a pragmatic vision conforming to the H.264 and DASH

standards. The video player was implemented on top of the DASH-IF player (<http://dashif.org>). We conducted large scale evaluations for more than 400 hours of streaming with 50 consumers spread around Europe, equipped with various devices (hand-devices, laptops, multiple OS). Twelve servers and set-top-boxes, with various capacities (in terms upload capabilities), were deployed over 7 cities in 4 European countries. The Internet connectivity was provided by 9 different ISPs. Each client could simultaneously use from 1 to 10 servers. 15 different 4 to 7-minute videos were made available at 7 bitrates: 0.2, 1, 1.5, 2, 3, 4 and 6 Mbps.

We derived a set of criteria, each considered essential for the QoE of streaming services, and compared MS-STREAM vs a DASH client (the dash.js player). Our implementation obtained in average 5.47Mbps mean bitrate whereas the DASH player reached 3.3Mbps only. Our player experienced very few rebuffering (0.4 per 5 minutes) compared to the DASH player (4 per 5 minutes) and also had a smaller start-up delay (1.1 sec vs. 2.03 secs). Thanks to the utilization of the P2P data exchange, these QoE gains came along with 67% of bandwidth savings at the server infrastructure side. Finally, as the resiliency mechanism of our solution includes cancelling some late sub-segment requests, hence having unused GoPs transferred to the consuming clients, a 6.5% bandwidth overhead was observed.

4 DEMONSTRATION SCENARIO

We provide an online platform [7] to test our P2P/multi-server streaming solution. This platform implements personalizable and interactive demonstrations of our solution for multiple users at the same time and in heterogeneous network environments. In this demo, end-users are given the possibility to add/discard servers contributing to their streaming session, and can decide whether to act as contributing peers. Additionally, live feeds are displayed to end-users regarding the displayed bitrate, video freezing events, used servers/peers, data from servers and peers as well as the uploaded data. End-users can also simulate several peers.

REFERENCES

- [1] J. Bruneau-Queyreix, M. Lacaud, and D. Negru. 2016. Multiple Description-DASH: Pragmatic video streaming maximizing End-Users' Quality of Experience. In *IEEE International Conf. on Communications (ICC)*.
- [2] J. Bruneau-Queyreix, M. Lacaud, and D. Negru. 2017. A multiple-source adaptive streaming solution enhancing consumer's perceived quality. In *IEEE Consumer Communications and Networking Conf. (CCNC), demonstration track*.
- [3] J. Bruneau-Queyreix, M. Lacaud, and D. Negru. 2017. MS-Stream: A multiple-source adaptive streaming solution enhancing consumer's perceived quality. In *IEEE Consumer Communications and Networking Conf. (CCNC)*.
- [4] J. Bruneau-Queyreix, M. Lacaud, D. Negru, J. Batalla, and E. Borcoci. 2017. QoE Enhancement Through Cost-Effective Adaptation Process for Multiple-Server Streaming over HTTP. In *IEEE International Conf. on Multimedia and Expo (ICME)*.
- [5] J. Bruneau-Queyreix, M. Lacaud, D. Negru, J. Mongay Batalla, and E. Borcoci. 2017. Adding a New Dimension to HTTP Adaptive Streaming through Multiple-Source Capabilities. *IEEE MultiMedia* (2017).
- [6] Cisco. 2016. Cisco Visual Networking Index. (2016). <http://www.cisco.com/>
- [7] MS-Stream. 2017. MS-Stream Demonstration. <http://msstream.net>. (2017).
- [8] I. Sodagar. 2011. The MPEG-DASH Standard for Multimedia Streaming Over the Internet. *IEEE Multimedia* 18, 4 (2011), 62–67.