ELectronics EXpress

A new framework for managing video-on-demand servers: quad-tier hybrid architecture

Phooi Yee Lau[1,2a)] **and Sungkwon Park**[1]

[1] *Media Communications Laboratory, Hanyang University*
17, Haengdang-dong, Seongdong-gu, Seoul 133–791, Republic of Korea
[2] *Faculty of Information and Communication Technology, Universiti Tunku Abdul Rahman*
Jalan Universiti, Bandar Barat, 31900 Kampar, Malaysia
a) *laupy@utar.edu.my*

Abstract: This paper proposes a new framework for managing large number of Video-on-Demand requests using the quad-tier hybrid architecture. This architecture provision four tiers to serve on-demand video requests and videos are mirrored to every tier based on a time-weighted popularity (TP) index derived from subscribers' log. *Tier-1 Near-Server* and *Tier-2 Quasi-Server* 'broadcast' most popular video while *Tier-3 True-Server*, located closer to subscribers, serve ad hoc on-demand video request using multicast protocol. *Tier-4 Subscriber* preloads frequently watch videos into the set-up-box for instantaneous playback. We propose suitable operational condition and evaluate the proposed TP index in the discussion.

Keywords: video delivery, VoD, hybrid architecture, user-interest
Classification: Other communication hardwares

References

[1] A. Nafaa, S. Murphy, and L. Murphy, "Analysis of a large-scale VoD architecture for broadband operators: a P2P-based solution," *IEEE Communications Mag.*, vol. 46, no. 12, pp. 47–55, Dec. 2008.

[2] T. D. C. Little and D. Venkatesh, "Popularity-based assignment of movies to storage devices in a video-on-demand system," *Multimedia Systems*, vol. 2, no. 6, pp. 280–287, Jan. 1995.

[3] J. Paris, "A stream tapping protocol with partial preloading," *Proc. 9th IEEE Int. Sym. MASCOTS*, Cincinnati, USA, pp. 423–430, 2001.

[4] J. F. Paris, S. W. Carter, and D. E. Long, "A Low Bandwidth Broadcasting Protocol for Video on Demand," *Proc. Int. Conf. Computer Communications and Networks*, Washington, USA, pp. 690–697, 1998.

[5] E. Lee and S. Park, "Internet Group Management Protocol for IPTV Services in Passive Optical Network," *IEICE Trans. Communications*, vol. E93-B, no. 2, pp. 293–296, Feb. 2010.

[6] P. Y. Lau, S. Park, and T. Kim, "Dynamic Time-Weighted Popularity Index: A Video-on-Demand Case," *Proc. 2010 IEEE IC-NIDC*, Beijing, China, pp. 809–814, 2010.

DOI: 10.1587/elex.8.1399
Received July 05, 2011
Accepted August 09, 2011
Published September 10, 2011

1 Introduction

Broadband Internet access has rapidly expanded and bandwidth-intensive applications, such as Video-on-Demand (VoD), has help fuel the demand for bandwidth. Since VoD services are expected to emerge as the main stream to access video content over IP networks, there is a need to design a framework that could efficiently manage large number of VoD requests [1]. Service providers usually support VoD on centralized architecture based on hierarchy of servers located at specific places in the network [2]. Assuming that videos are in MPEG-2 format, Paris reported that if each user requires about 5 Mbps of data per second; a server allocating a separate data stream for each VoD request would need an aggregated bandwidth of 5 Gbps to accommodate 1000 concurrent users [3]. On the other hand, service providers relentlessly aim to allocate their available bandwidth efficiently to maximize revenues [4]. So, the question is where should the servers be located? This has led us to propose a new framework for VoD delivery, i.e. quad-tier hybrid architecture (QTHA). This architecture provisions four tiers to serve on-demand video requests, i.e. 'broadcast' most popular videos from *Tier-1 Near-Server* and *Tier-2 Quasi-Server*, while popular videos are being delivered using the multicasting protocol from *Tier-3 True-Server*, as they do not consume bandwidth in the absence of a request. A time-weighted popularity (TP) index is proposed to rank videos' popularity and a recommendation engine, named *Playlist*, is proposed to display the drifting of videos instantaneously and to allow subscribers select any videos to watch.

2 Quad-Tier Hybrid Architecture

The QTHA exploits the underlying hierarchical network and rely heavily on the TP index. A TP index ranks each video at each tier and is derived

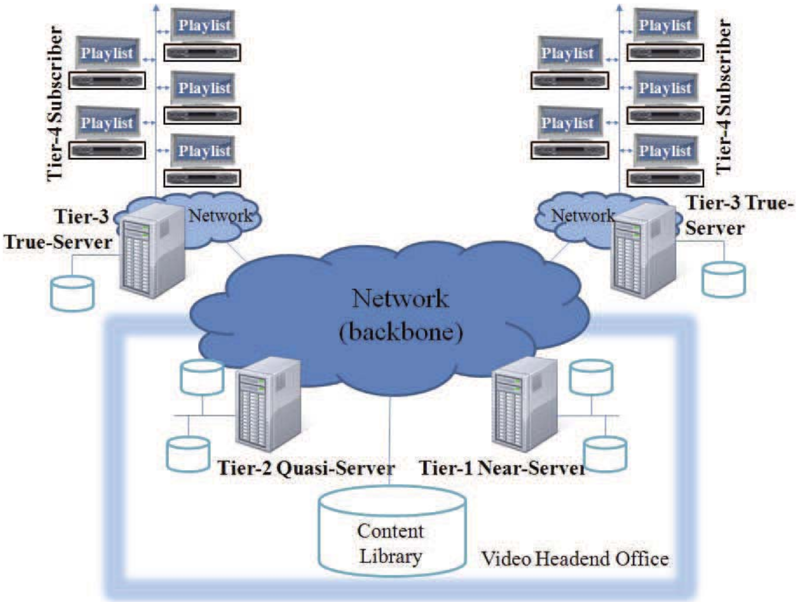


Fig. 1. Quad-tier hybrid architecture

DOI: 10.1587/elex.8.1399
Received July 05, 2011
Accepted August 09, 2011
Published September 10, 2011

from instantaneous video popularity. The QTHA integrates four VoD services, namely: 1) *Tier-1 Near-Server* ('broadcast'/remote), 2) *Tier-2 Quasi-Server* ('broadcast'/remote), 3) *Tier-3 True-Server* (multicast/proxy), and 4) *Tier-4 Subscriber* - see Fig. 1. This architecture allows 1) *Tier-1 Near-Server* and *Tier-2 Quasi-Server* to 'broadcast' most popular videos, 2) *Tier-3 True-Server* to multicast popular videos, using the earlier proposed method presented in [5], and 3) *Tier-4 Subscriber* to store videos, thus, allowing subscriber access videos effortlessly.

2.1 Playlist

A *Playlist* is an application that lists all available videos and their mirrored locations [6]. This *Playlist* allows user to freely select a video to be play instantaneously or wait for a video scheduled to be 'broadcast' later - see Fig. 2.

- *Playlist 4* - no startup delay; videos are mirrored to the STBs
- *Playlist 3* - minimum startup delay; videos are mirrored to *Tier-3 True-Server*
- *Playlist 2* - start-up delay depends on the time the video is scheduled to be multicast next; videos are mirrored to *Tier-2 Quasi-Server*
- *Playlist 1* - start-up delay depends on the time the video is scheduled to be multicast next; videos are mirrored to *Tier-1 Near-Server*

2.2 Assignment of videos

A TP index ranks the videos popularity based on the time all subscribers spent watching a video out of the total hours the subscriber logged [6]. It is calculated for each video and it influences where the video will be mirrored and streamed, from which server, and using what delivery scheme, as shown in Eq. (1).

$$TP_{x(k)} = \sum_{i=1}^{n} \frac{t_w(i)}{t_d} \qquad i \in \{1, 2, 3, \ldots, n\} \tag{1}$$

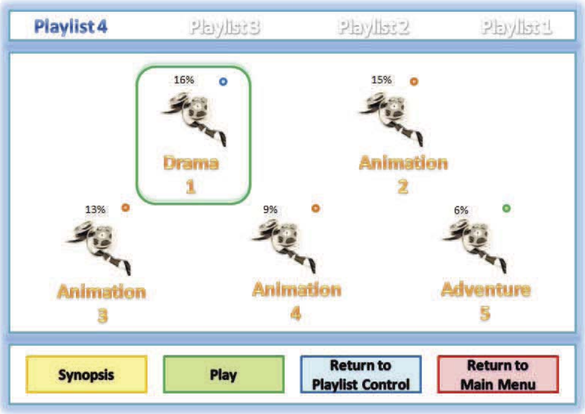


Fig. 2. Conceptual Graphical User Interface of *Playlist*

DOI: 10.1587/elex.8.1399
Received July 05, 2011
Accepted August 09, 2011
Published September 10, 2011

where $x \in$ {Tier1,Tier2,Tier3,Tier4}, k is the video code, n is the number of times a subscriber(s) logged in to watch a video, i is a counter, t_w represents the duration for which a subscriber(s) watches a video and t_d represents the total hours logged by a subscriber(s).

2.2.1 'Broadcast' video from Tier-1 Near-Server

Tier-1 Near-Server 'broadcast' most popular videos. It exploits the characteristics of Near-VoD server using the $TP_{Tier1(k)}$ index - see Eq. (1). x is *Tier1*, n is the total number of *Tier-4 Subscriber* watching the video. Assuming 20 subscribers watched video *54321*, $t_w = \{20, 37, 93, 43, 62, 49, 50, 11, 26, 4, 73, 16, 38, 27, 7, 9, 28, 18, 38, 62\}$, t_d = 500,000 hours, and the $TP_{Tier1(54321)}$ will be 2.38×10^{-5}.

2.2.2 'Broadcast' video from Tier-2 Quasi-Server

Tier-2 Quasi-Server 'broadcast' most popular videos for different quasi-genre group presented in [6]. This server periodically 'broadcast' most popular videos, excluding* those already streamed from *Tier-1 Near-Server*, using the $TP_{Tier2(k)}$ index, calculated for all video from different quasi-genre group - see Eq. (1). x is *Tier2*, n is the total number of *Tier-4 Subscriber* watching the video in particular quasi-genre group. For example, assume 10 subscribers watched video *98765* from Quasi-Group 'Animation', $t_w = \{50, 37, 93, 43, 62, 49, 50, 41, 26, 62\}$ and t_d = 40,000 hours, the $TP_{Tier2(98765)}$ will be 2.14×10^{-4}. *Note:* *There are no duplications.

2.2.3 Multicast video from Tier-3 True-Server

Tier-3 True-Server is placed closer to the subscriber in order to maximize the streaming efficiency - see Fig. 1. This VoD server only mirrors frequently watched videos based on $TP_{Tier3(k)}$ index in a service area, as it is not feasible to mirror all on-demand videos - see Eq. (1). x is *Tier3*, n is the total number of *Tier-4 Subscriber* for *Tier-3 True-Server*. For example, assume 5 subscribers watched video *07654* logged at service area *A301*, $t_w = \{50, 43, 49, 55, 27\}$ and t_d = 2,000 hours, the $TP_{Tier3-A301(07654)}$ will be 1.87×10^{-3}.

2.2.4 Mirroring video at Tier-4 Subscriber

Tier-4 Subscriber exploits the unused capacity in the STBs to pre-load selected video based-on $TP_{Tier4(k)}$. Initially, a recent log is calculated, using Eq. (2).

$$TP_{Tier4(L)(k)} = \sum_{v=1}^{n} \frac{t_w(v)}{t_d} \qquad v \in \{1, 2, 3, \ldots, n\} \tag{2}$$

where L is the counter for registering the number of log for the video, n is the total number time *Tier-4 Subscriber* watch video k, i is a counter, t_w represents the duration subscriber(s) watch the video and t_d represents the total hours logged by the subscriber. For example, subscriber *T400001* watches video *12345* twice, 20 minute for each log-in, and logged a total

DOI: 10.1587/elex.8.1399
Received July 05, 2011
Accepted August 09, 2011
Published September 10, 2011

of 3 hours of watching time. The $TP_{T400001(1)(12345)}$ is 0.22. Then, this $TP_{T400001(1)(12345)}$ is weighted with older-log, if any, using a time element as shown in Eq. (3) to obtain $TP_{Tier4(k)}$. Notice that, a subscriber who frequently watches the same video has higher $TP_{Tier4(k)}$.

$$TP_{Tier4(k)} = \sum_{L=1}^{m} \frac{TP_{Tier4(L)(k)}}{L} \quad L \in \{1, 2, 3, \ldots, m\} \tag{3}$$

3 Discussions

3.1 Operational consideration

A moderate size server can serve up to 15,000 subscribers concurrently and a proxy server may handle subscribers in the order of thousands at most. In a region, there may be a small number of subscribers or an extreme number of subscribers and it all depends on the popularity of their services. *Note:* We determined the effective number of popular movies using a normal Zipf distribution.

- *Tier-1 Near-Server* and *Tier-2 Quasi-Server* - Zipf distribution assumes 90% of subscribers watch top 50 videos. As the system has a total of ten 'broadcast' group: one from *Tier-1 Near-Server* and nine from *Tier-2 Quasi-Server* [6], we recommend to mirror top 5 videos for each group. Moreover, assuming a server scheduled a multicast stream every 5 minutes, a 2 hours video encoded in MPEG-2 format would need an aggregated bandwidth of 6 Gbps. If we increase to 8 videos per group, aggregated bandwidth needed to stream the videos would exceed 10 Gbps, i.e. exceeding the bandwidth available for a fiber core network.

- *Tier-3 True-Server* - An hour of HD video requires at most 7 GByte of storage capacity. Assuming a 2 hour video, 100 HD movies would require about 1.5 Terabytes of storage, which is the size of a normal VoD server using hard disk as storage. On the other hand, if 90% of the video request will be served by the scheduled multicast stream, then, only 10% video request will be served by the *Tier-3 True-Server* using the QTHA. Assuming *Tier-3 True-Server* may handle customers in the order of thousands at most; 10% of the subscribers (in hundreds) may have enough selection to choose a video to watch. Therefore, we recommend mirroring top 100 videos to serve video request for popular videos in this VoD server.

- *Tier-4 Subscriber* - The system enable frequently watched videos to be mirrored in the STB. For example, for a subscriber who favours watching animation video, the system automatically registers his/her preferences and mirrors frequently watched videos in the STB. In Korea, most of the hard disks in the STB are larger than 100 GBytes. At the moment, a STB with 600 GBytes of hard disk has already been commercialized but its price remains high. Consider the capacity and

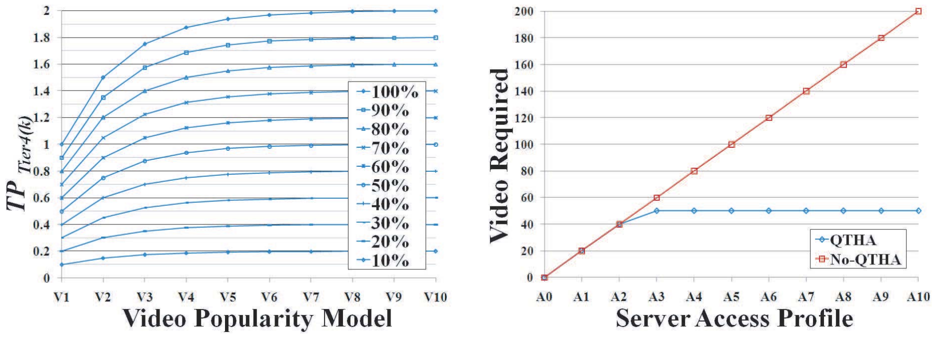
DOI: 10.1587/elex.8.1399
Received July 05, 2011
Accepted August 09, 2011
Published September 10, 2011

cost of hard disk (assuming a disk space of 5-7 Gbyte would be able to store about an hour of ATSC HDTV videos), we recommend to mirror at most 10 videos.

3.2 Evaluating the TP index

- Analyse different viewing interest:- Assume ten different viewing interest, V1 through V10, and each depicts the subscriber watching the same video for y% of the total subscriber's log, e.g. V1 depicts a subscriber spending 10% of the time watching a video. Analysis results are shown in Fig. 3 (a). Results show that when a subscriber depicts a single viewing trend, i.e. more than 6 times, the $TP_{Tier4(k)}$ index increases up to twice the initial index, i.e. initial index for V1 is 0.10 while V6 is 0.19. The longer a subscriber spend watching a video, the higher the TP index would be, and vice versa. This index varies depends on the viewing interest of all subscribers.

- Analyse different access profile:- Consider two servers receiving 200 requests at one time. There are 11 access profiles, with m probability requesting for videos mirrored in remote servers (*Tier-1* and *Tier-2*) and n probability requesting for videos mirrored in the network edge (*Tier-3* and and *Tier-4*). For example A0 Profile describe all subscribers request for a video from network edge, and A10 Profile describe all subscribers request for a video from remote servers, 10% increase/decrease for each profile, respectively. When all requests made are for popular videos, i.e. A0 profile, the total number of request for videos in other VoD server is null, see Fig. 3 (b). For A10 profile, only 50 videos are streamed using the QTHA. Service providers which uses a single centralized server, in the worst-case scenario, could only serve limited incoming video request, even if they fully utilize all their network resources.



(a) Popularity index

(b) Total videos required

Fig. 3. (a) Different viewing profiles (b) Different access profiles

DOI: 10.1587/elex.8.1399
Received July 05, 2011
Accepted August 09, 2011
Published September 10, 2011

4 Conclusion

This paper discusses how large numbers of on-demand video request can be efficiently managed using the QTHA. The QTHA depends heavily on the TP index and *Playlist*. It provisions different tier to serve different on-demand video request. Our idea is to enable a subscriber: (1) watch a video instantaneously available in *Tier-4 Subscriber* ; (2) watch a highly popular video from *Tier-1 Near-Server* or *Tier-2 Quasi-Server*; and (3) watch other videos from *Tier-3 True-Server*. We consider the following problems as dominant when designing the architecture, namely the ability: 1) to scale up to potential subscriber (quad-tier hybrid architecture); 2) to facilitate speedier access to content (popularity drifting policy); 3) to allow subscribers' full control over content (*Playlist*); and 4) to support large content libraries (TP index).

Acknowledgments

This work was supported in part by research fund from the Brain Korea 21 (BK21) Project in 2011 and the Seoul R&BD Program (No. PA090720).

DOI: 10.1587/elex.8.1399
Received July 05, 2011
Accepted August 09, 2011
Published September 10, 2011