Effective P2P VoD Service Distribution over HFC Networks

Jordan Nenkov[1] and Lidia Jordanova[2]

Abstract – **The paper deals with a concept for realizing a Peer-to-Peer (P2P) VoD system over HFC networks in a way to ensure possibilities to increase the number of both the subscribers and the movies supported. Different methods to transfer video information from the VoD server to subscribers are considered. Three strategies to limit the downstream video traffic are suggested.**

Keywords – **P2P VoD, multicast video, MPEG2-TS, TVoD.**

I. Introduction

Due to recent advances in broadband Internet access technology the video-on-demand (VoD) service seems to gain an increasing popularity among media streaming services. VOD networks are developed to deliver video files to distant users with minimal delay and free interactivity [1].

Traditionally, the VoD service is based on a centralized architecture. However, this architecture cannot provide the quality of service needed to a large population of users due to its limited outbound channel capacity from the server to the clients. Recently, a peer-to-peer (P2P) architecture was proposed to meet the challenge of providing live and interactive video broadcast to a large number of clients over a wide area [2]. A P2P-based architecture is appropriate enough for the design of a scalable VoD services distribution architecture as the computing and bandwidth requirements are pushed toward the network clients side. Besides, it allows optimal use of the network resources by building multisource streams from neighbouring contributing clients to a requesting client. This in turns results in minimizing the VoD request rejection rates for a very large content library [3].

The aim of this work is to compare the methods of video information transmission from VoD server to subscribers and to choose the most appropriate one in terms of price, quality and complexity of implementation.

II. VoD Service Strategies

There are three main types of service strategies: broadcast, unicast and multicast.

[1]Jordan Nenkov is with the Faculty of Telecommunications, Kliment Ohridski 8, 1756 Sofia, Bulgaria, E-mail: Jordan_n2002@yahoo.com.

[2]Lidia Jordanova is with the Faculty of Telecommunications, Kliment Ohridski 8, 1756 Sofia, Bulgaria, E-mail: jordanova@tu-sofia.bg.

A. Unicast Communication Strategy

This is a point-to-point communication channel which means that unicast establishes communication between a single sender and a single receiver over the network.

With unicast, as shown in Fig. 1, an independent flow of information is sent to each one of the clients requesting the multimedia content.

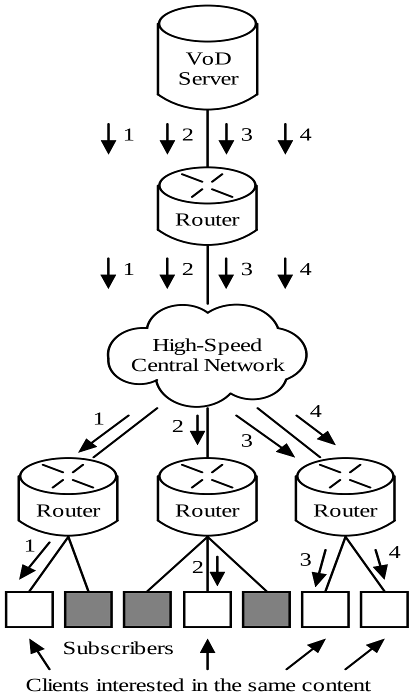

Fig. 1. Unicast communication strategy.

The main advantage of this technique is the control mechanism. Each communication channel is independent, thus the control is oriented to each client and there are no interactions between the channels. On the other hand, the bad scalability of the scheme is a great constraint of the unicast usage. Each channel consumes server and network resources in a linear fashion, which results in a rapid consumption of the resources predicted during the system's design. Unicast converts the server into a bottleneck for the VoD system.

B. Broadcast Communication Strategy

This service strategy is based on communication from a single sender to all the receivers over a network. In this scenario the video is broadcast over a dedicated channel within a predefined schedule. This approach can provide service to an unlimited number of requests for popular video content with a constant consumption of bandwidth. However, a great amount of resources can be consumed pointlessly if the popularity of the video is low. In addition, clients must wait until the scheduled time to be served. In Fig. 2 a broadcast communication scheme is shown, where the information is distributed to every end host in the system.

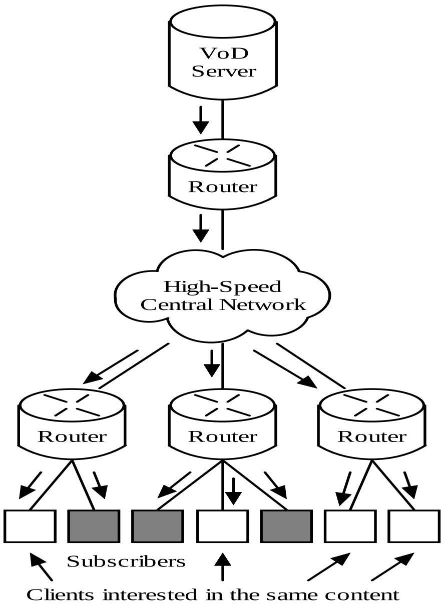

Fig. 2. Broadcast communication strategy

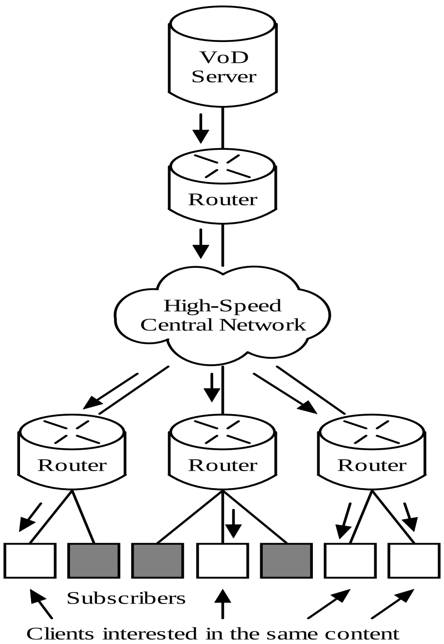

Fig. 3. Multicast communication strategy

C. Multicast Communication Strategy

The multicast technique consists of an information flow from a source to a group of receivers who have requested the same content. A number of requests for the same video are grouped and served by a single video stream. Therefore, the service scheme saves bandwidth by sharing video streams without wasting resources on non-requesting receivers. Multicast is shown in Fig. 3. It provides the best solution to scale large VoD systems.

III. THE COLLABORATIVE METHOD FOR DOWNSTREAM VIDEO TRAFFIC DECREASING

Let's analyze a collaborative method of reducing the downstream video traffic as shown in Fig. 4. After subscriber 1 has requested a video, he is not served immediately but after some time *t*, during which it is assumed that subscriber 2 will make a request for the same film. Once a subscriber has made a second request for the film and there is an awaiting subscriber (subscriber 1), the film (video stream) is fed to both subscribers simultaneously. In Fig. 4 the following symbols are used:

req. 1 and req. 2 are the requests for one and the same movie as made by subscribers 1 and 2 respectively;

reply 1 and reply 2 – request fulfilled for one and the same film, in reply to requests made by subscribers 1 and 2 respectively;

t1 – time when a request from subscriber 1 was issued;

t2 – time when a request from subscriber 2 was issued;

t3 – time to reply to both subscribers;

τ is the time between the requests for one and the same film issued by subscribers 1 and 2;

$\tau 1$ and $\tau 2$ – time to run between query and reply to subscribers 1 and 2 respectively.

The more the subscribers that have requested one and the same movie in the time interval τ, the more the ports on the VoD server that will be free for future requests. A disadvantage of this method is the delay in executing the movie requests. The value of τ must be chosen in a way to guarantee that the delay is below a given value acceptable for subscribers.

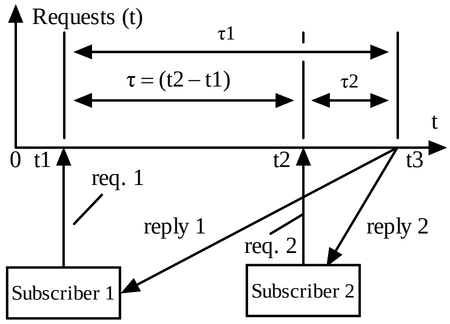

Fig. 4. Collaborative method for downstream video traffic decreasing

IV. METHOD OF DOWNSTREAM VIDEO TRAFFIC DECREASING WITH REDUCING THE CLIENT WAITING TIME

The method consists in the following. Every i^{th} video V_i is divided into 3 parts, the first W_1 minutes of each video V_i being referred to as prefix-1 $(pref\text{-}1)_i$ of V_i. If V_i is globally popular, it is replicated at all M Proxy Servers; otherwise, it is replicated across L Proxy Servers, in which the frequency of accessing the video V_i is high. The next part of W_2 minutes of video $V_{i,}$ referred to as prefix-2 $(pref\text{-}2)_i$ of $V_{i,}$ is downloaded from the Tracker and the rest of the video is referred to as suffix of the video and is stored at VoD Server, as shown in Fig.5.

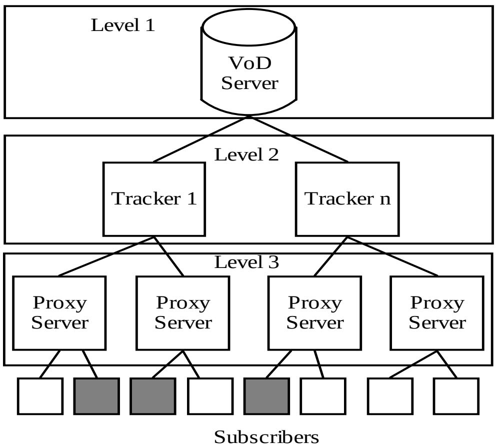

Fig. 5. Architecture realization of the method

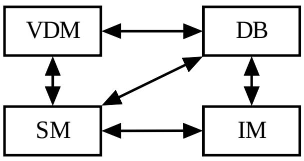

Fig. 6. A Tracker module

The proposed 3-layer architecture consists of a VoD Server, which is connected to a group of Trackers. Each tracker has various modules, as shown in the Fig. 6:

1. Interaction Module (IM) that interacts with the Proxy Server and the VoD Server.
2. Service Manager (SM) that handles the requests from the Proxy Servers.
3. Database (DB) that stores detailed information about availability and size of (pref-1) of videos at all the Proxy Servers.
4. Video Distributing Manager (VDM) which is responsible for decision-making about videos and sizes of (pref-1), (pref-2) of videos to be cached. It also handles the distribution and management of these videos to the Proxy Servers group, according to their global and local popularity.

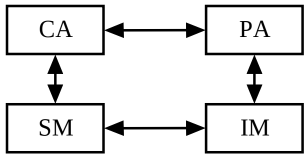

Fig. 7. A Proxy Server module

Each Tracker is in turn connected to a set of Proxy Servers. Each Proxy Server has various modules, as seen in Fig. 7:

1. Interaction Module (IM) that interacts with the user and the Tracker;
2. Service Manager (SM) that handles the requests from the user;
3. Popularity agent (PA) that observes and updates the popularity of videos at Proxy Servers and at the Tracker as well as;
4. Cache Allocator (CA) that allocates the Cache blocks using dynamic buffer allocation algorithm [4]. A large number of users are connected to each of these Proxy Servers as well. Each proxy server acts as a parent proxy server to its clients.

The Proxy Server caches the (pref-1) of videos distributed by VDM and streams this cached portion of the videos to the clients.

Here it is assumed that:

1. The Tracker is of high computational power. It has various modules to coordinate and maintain a database that contains the information about availability of videos and size of (pref-1) and (pref-2) of video in each Proxy Server and Tracker respectively.
2. Proxy Servers and their clients are closely located at a relatively low communication cost [5]. The VoD server where all the videos are stored is placed far away from Proxy Servers which involves high-cost remote communication.

V. METHOD FOR DOWNSTREAM VIDEO TRAFFIC DECREASING WITH VIDEO POPULARITY USING

The distribution of the VoD-movie requests generally follows a Zipf / Pareto-like distribution [6], where the VoD movies are of two classes: popular and unpopular. The relative probability of a request for *i* (the most popular movie) is proportional to $1/i^{\alpha}$, with $0 < \alpha < 1$, and typically takes a value that is less than unity [7]. The assumption here is that all blocks of a popular movie belonging to the popular class are stored in the mesh network of the client, and if needed by a given client, they can be downloaded from its mesh network partners. For Zipf-like distributions, the cumulative probability that one of the *k* popular movies class is accessed (i.e. the probability of a movie request from the client mesh network) is given asymptotically by:

$$\psi(k) = \sum \delta / i^{\alpha} = \delta k^{1-\alpha} / (1-\alpha), \quad \delta = (1-\alpha) / V^{1-\alpha} \qquad (1)$$

where *V* is the total number of movies in the system [7]. Next, we estimate the probability (Pu) of a request for a movie that belongs to the unpopular class (i.e. does not exist in the mesh network setup boxes) and therefore should be obtained from the central head-end video server. For a VoD system with *V* total movies and *k* popular ones at the mesh network the probability of a request for an unpopular movie stored in the head-end server is as follows:

$$P_u = 1-(k/V)^{1-\alpha}. \quad (2)$$

This method is applicable in networks with large numbers of subscribers. The more the subscribers and fewer the movie titles in the system, the better the results obtained with the above method.

VI. CONCLUSION

The paper deals with a comparative analysis of three types of communication strategies for transmission of video information from the VoD server to subscribers - unicast, broadcast and multicast. Three methods to reduce the downstream traffic from the VoD server to subscribers are considered. The following conclusions can be drawn:

1. The unicast communication strategy is appropriate when high security of information transmitted is required, but is inefficient in terms of price realization.
2. The broadcast communication strategy is a low-cost selling, but much of the resources are used unnecessarily when the popularity of films is low.
3. The multicast communication strategy saves bandwidth by sharing the video streams without loss of resources to unsolicited recipients. It provides the best solution for growing large VoD systems.
4. The collaborative method for downstream video traffic decreasing is easy to implement, but it results in movie requests execution latency which is a disadvantage of the method.
5. The method for downstream video traffic decreasing with reducing the client waiting time eliminates the shortcomings of the collective method but turns out to be complex and relatively expensive for implementation.
6. The method for downstream video traffic decreasing with video popularity using is applicable in networks with large numbers of subscribers. It gives much better results in systems with more subscribers and fewer movie titles.

REFERENCES

[1] F. Thouin and M. Coates, "Video-on-Demand Networks: Design Approaches and Future Challenges", IEEE Network Magazine, March/April 2007.

[2] Y. He, I. Lee, and L. Guan, “Distributed Throughput Maximization in P2P VoD Applications “ , IEEE Transaction on Multimedia , Vol. 11, No. 3, April 2009.

[3] A. Nafaa, S. Murphy, and L. Murphy, “Analysis of a Large-Scale VOD Architecture for Broadband Operators: A P2P-Based Solution” , IEEE Communications Magazine December 2008.

[4] H.S. Guruprasad, M. Dakshayini, “Dynamic Buffer Allocation for VoD System Based on Popularity” proceedings of NCIICT 2006, PSG College of Technology, Coibatore, 13- 17, www.psgtech. edu/NCIICT/files/NCIICT06.

[5] B. Wang, S. Sen, M. Adler and D. Towsley, “Optimal Proxy Cache Alloction for Efficient Streaming Media Distribution” IEEE Transaction on Multimedia, vol. 6, No. 2, April 2004.

[6] L. Jordanova, J. Nenkov, Distribution of Video-on-Demand Service over Cable Television Networks. Radioengineering, Vol. 2, No 09-021, 2009.

[7] L. Breslau, "Web Caching and Zipf-like Distributions: Evidence and Implication", Proc. IEEE INFOCOM 99, New York, March 1999, pp.126-34.