

Research on Video on Demand Based on P2P Network

Min Cui

School of Software Engineering, Shanghai Jiao tong University, Shanghai, China

E-mail: yin_cq@hotmail.com

Received March 10, 2009; revised October 25, 2009; accepted January 28, 2010

Abstract

A P2P approaches to extend the ability of Video on Demand systems to serve more users. In the proposed system users share with each other the media data obtained and the media server is no longer the only source to get data from, thereby, the load on the media server could be greatly alleviated and the overall system capacity increases and more users could be served. The P2P streaming system introduces efficient searching; data transfer dynamically monitoring and initial buffering to maintain a high quality of playback. Its provider selection policy helps to reduce the load of the underlying network by avoiding remote data transfer.

Keywords: P2P, Peer to Peer, Video on Demand, Streaming Media, Super-Peer, Cache Replacement Algorithm

1. Introduction

Recently, with the rapid development of large-scale storage, high performance workstation and broadband network technology, the information transferred on the Internet consists of not only text and images, but also varieties of multimedia communication, which is possible on technical and economic aspects. Multimedia communication makes video, audio, text, images as one kind, and provides richer user experience, so that it is used more and more extensive. Streaming media technology makes it possible to transfer audio, video or animation to users' PC continuously and in real-time mode instead of waiting for the whole completion.

However, the traditional client-server based network streaming media system consumes large network bandwidth and server resource, which prevents the development of the network media services. For example, a video needs 300 kbps bandwidth, and 1000 users watch it at the same time, so that we need 300 Mbps bandwidth, which is over the server limit. If there were much more users, then we could only add more servers to support the service. So, a new kind of transfer mode is needed.

In order to solve the problem above, many kinds of solutions have been brought up in research field and industry field in recent years; some important ones include Content Delivery Network CDN and IP multi-cast. But these solutions need special hardware to run, for example, several CDN servers are needed worldwide to set up CDN; IP multi-cast needs the modification of the current Internet routing method and more routers supporting the multi-cast ability. Such solution is too expensive and can

not solve our problem completely.

Then, P2P network gets in. A peer-to-peer, commonly abbreviated to P2P, is any distributed network architecture composed of participants that make a portion of their resources directly available to other network participants, without the need for central coordination instances. Peers are both suppliers and consumers of resources, in contrast to the traditional client-server model where only servers supply, and clients consume.

BT has many advantages we can learn, such as pipeline, choking algorithm, tit-for-tat theory. It uses SHA1 hash function to index file chunks uniquely, and apply random policy to the missing chunks in order to provide data resource to clients more efficient and balance each file trunks' copies in the network. For the finishing file trunk, the client can request other clients to provide the trunk so that this file trunk can be finished transferring earlier.

2. Simple VOD System Model

As the P2P media technology improves and develops, the VOD system based on P2P technology has more and more applications. Now there exists mature P2P stream software such as PPLIVE and PPSTREAM. For the P2P streaming media service has many advantages, lots of research institutions dive into the P2P transfer mode. University of Stanford in USA raised Spread system, which combines the Peers to a single grouped tree and use the Peer to forward media stream. University of Florida got the ZIGZAG, which organized the Peers into layered groups and set up the grouped tree based on

some policies to achieve large-scale broad-casting. An American researcher brought up the PROP system, which achieves extensible streaming media live using proxy. University of Massachusetts designed and implemented DirectStream, a directory based video live broadcast system. The directory maintains the video information index and all the peers' information in the grouped tree on the server. New peer will look up the index and determine whether to join the existing tree or create a new tree. Microsoft designed the SplitStream streaming media technology after CoopNet. This system aims to lower the load of each grouped tree by setting up several trees. The internal nodes in the tree are the sub nodes of another tree. The complete stream is split into some parts to be transferred in every tree. Thus, the load will be gone shares on the trees.

Relative to the video broadcast system, VOD is asynchronous. For the users who playing the same program, the startup time among each user are not the same, and the Pause, Resume etc executive by the users need more complex catch and transmission strategies to achieve the advantages of P2P. Therefore, the main idea of this article is that model for a broadcast system, and deal with some basic problem of this system model. First of all, let's see the structure of the whole system.

The VOD system model in this article will get involved with a super peer. The basic system structure will be given as follows.

P2P Streaming Media Play System is consisting with the P2P streaming media servers and other child nodes. Each network nodes of P2P network is equal, each two nodes is server and client. To ensure the system work effectively, we set P2P streaming media server as Web information management server of other child node that manages the system's users' information and program resources information.

System structure is shown in **Figure 1**. The server and client connect through the Internet.

The P2P streaming media server not only provide general streaming media server primary function, but also play a role of the contents of the index server, its functions are as follows:

1) Network management: To monitor the operational status of the child nodes, display the network status, traffic statistics and so on.

2) Resource management: To storage media files,

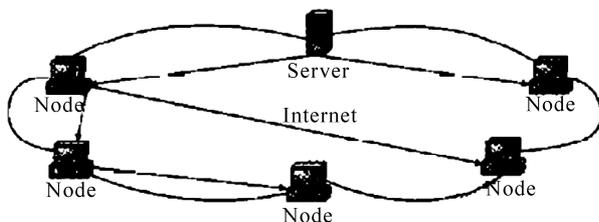


Figure 1. P2P broadcast system structure.

media resources search (locate), maintain the system node resource information, exchange resources and information with other master node. For each block of streaming media, maintain a online streaming media client list of all downloaded and cached this block, provide the download address of all requests for streaming media blocks for new customers.

3) User Management: To register node user information, implement user authorization management, control user system permissions to verify the legitimacy of the visitor.

4) Media server: To provide the child nodes of the system with the original streaming media content.

5) When necessary, the server would do block, encoding, packaging and transport and so on on multimedia content.

The functions of child nodes are as follows:

1) Streaming media services: as the streaming media server of other child nodes to provide streaming media services.

2) Local media resources: management media files have been downloaded, maintain node resource information.

3) Media play: re-sorted the received streaming media data to achieve broadcast with media player controls.

Figure 2 shows the specific process on broadcast.

3. Cache Replacement Policy

The classic P2P based VOD system can be parted into two types, one is P2VOD represented tree model, and other is net model. Some are based on GOSSIP protocol, and some are based on GNUSTREAM, which is an organized P2P network.

A good replacement algorithm should take both the popularity of the media file and the cache status of the file in other peers into account. We believe that, if we keep that the number of copies of all the streaming media blocks which in system cache are directly proportional to

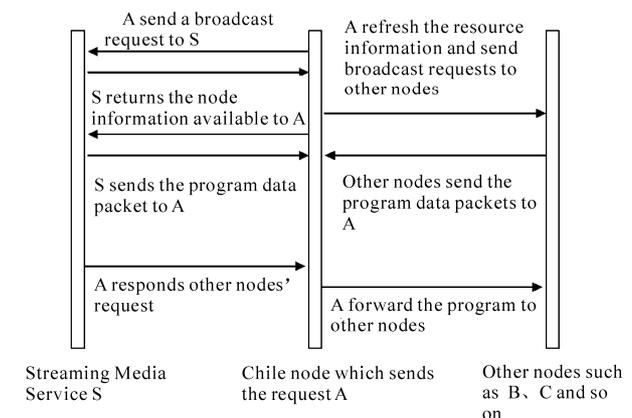


Figure 2. Broadcast process.

its popularity, it will most likely reduce the number that nodes downloaded from the server. Therefore, as the following formula defined, the replacement algorithm calculates the ranking value R for streaming media.

R = the popularity of streaming media block (D)/the copy number of media blocks system has been cached (N)

We consider a new replacement algorithm according this; this algorithm requires that the server statistics all the media block access frequency, so that they used to estimate the prevalence of streaming media. Specific algorithm is as follows.

When a child node requests for streaming media block A and get the respond, follows are the implementation of the algorithm's pseudo-code:

```

if (Empty_Buffer_Size  $\geq$  size of ( $A$ ))
  cache  $A$ ;
else
  {
  Download  $F_a$  and  $N_a$  from the server;
  For all  $B_i \in \Phi$ , Download  $F_{b_i}$  and  $N_{b_i}$  from the
server;
  Calculate  $R_{b_i} = F_{b_i}/(N_{b_i} - 1)$ 
   $R_a = F_a/(N_a + 1)$ ;
  Calculate sum of all  $R_{b_i} < R_a$  size of ( $B_i$ ), as  $C$ ;
  if ( $C >$  size of ( $A$ ))
  {
  While (Empty_Buffer_Size  $<$  size of ( $A$ ))
  Delete  $B_i$ , and  $R_{b_i} = \min\{R_{b_i}\}$ ;
  cache  $A$ ;
  }
  else
  not cache  $A$ ;
  }

```

One symbol meanings are as follows:

A : stream media block to be downloaded;

N_a : the number of child nodes that cache A ;

B_i : the i cache streaming media block that cached by self node;

N_{b_i} : the number of child nodes that cache B_i ;

F_{b_i} : the recent frequency of B_i 's request according the server statistics, which is popularity of B_i ;

Φ : set of all streaming media block cached by self node;

Empty_Buffer_Size: the free space of self node cache.

4. Overall Optimizing

It is not enough to ensure a smooth playing to depend on dynamic monitoring only.

If the use starts to play the first chunk after it has been downloaded and begins downloading the second chunk, and assuming that the user will never start downloading the next chunk until the current one has been finished (we use t to present the length of the first chunk's play-

ing), then every chunk must be finished within the period t except the first chunk, otherwise the gap will exist and cause the temporarily stopping of the video. Although dynamic monitoring can ensure that the downloading rate will be larger than the video playing rate R , it is possible that a single chunk downloading rate is smaller than R for the unexpected service node adjusting. As a result, the chunk can not be finished downloading in time.

In view of this question, a method has been proposed that a length of $N + 1$ chunks buffering space is set at the client.

Users start play when fragments of the first N downloaded, while the $N + 1$ fragments start the download, so as the follows. So only when each fragment from the first $N + 1$ fragment complete download in $N \times t$ time, it will not cause playback pauses. This has undoubtedly increased the user's startup delays; we can reduce this delay through giving the user maximum download speed, rather than video playback rate R to transmit the former N -fragment method. In the given simulation we can see, there has relationship between fragment size and download speed to ensure smooth playback of the N 's minimum value, if not use the method that download N fragment accelerate, this started delay can be tolerated.

We use GT-ITM network topology generator to generate a 500-node Internet-like topology, with the ns-2 network simulator as a simulation of the two aspects.

Simulation environment parameters are set to: Media file's length is 30 minutes, and is divided into a series of 10 second clips. Video recording speed is 256 Kb/s. Speed of dial users is 1 Mb/s, the bandwidth of LAN is 10 Mb/s. Backbone network bandwidth is 155 Mb/s, streaming media server bandwidth is 10 Mb/s. Adding of system users obeys Poisson distribution.

Figure 3 shows the contrast result that the client resources for different cache settings and system service capacity. User cache 0% means that all users directly get resources from the streaming media server, equivalent to the traditional C/S mode of video broadcast system. Figure 50% of the user cache and user cache of 70%, the service capacity of the system are equal because that the speed limit because the system backbone network service capabilities.

The initial buffer time

For the initial buffering time testing, in view of the media file processing by 2 seconds, 5 seconds, and 10 seconds. Which creates a play process, the principle is invoked according to the scheduling and request fragments. When you call the play process, if the corresponding fragments is downloaded, then repeat it again for next fragment after a time of calling process, if download is not complete, it requests again after 0.5 seconds pause the clip.

Figure 4 tells us the results of this test, we can see that after the 15 seconds initial buffer, it can be ensured that the broadcast stay basic stability. According to the size of

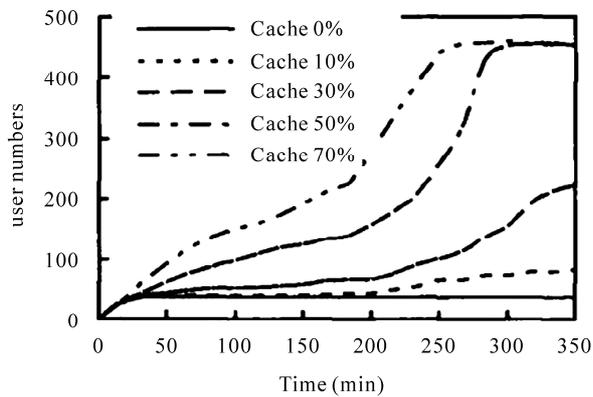


Figure 3. System server capacity under different cache.

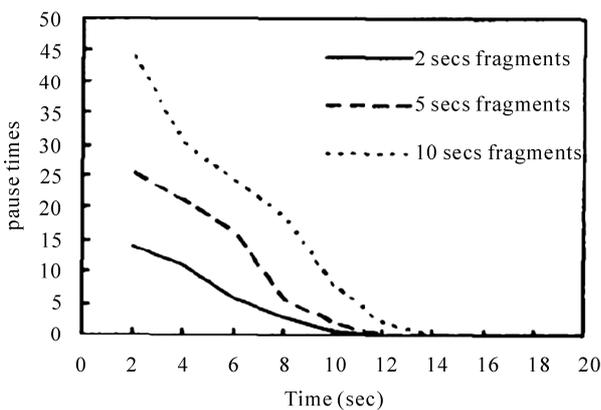


Figure 4. The relations between different initial buffer time and pause times.

this fragment can determine the number of the first download, fragment N , to meet the smooth playback and minimal start-up delay as far as possible.

5. Conclusions

This article mainly talks about some basic problems of P2P VOD, including the process of VOD, service node choice, network increasing, media file chunk choice, recovering of failed service node, data monitoring and QoS control. We set up a simple VOD system model and combine it with load of the servers and the topology of application layer grouped tree and complete distributed peer to peer network. Besides, this article also discusses more about the quality of the video playing, quoting the method of setting multi-level length of the streaming media for the client buffer space and lists the test cases about system service capacity and initial buffering time. The overall system optimizing is also involved.

6. References

- [1] Napster. <http://www.napster.com>
- [2] W. L. Zhou and X. F. Wu, "Survey of P2P Technologies," *Computer Engineering and Design*, Vol. 27, No. 1, 2006, pp. 76-79.
- [3] L. D. Cheng and S. J. Li, "On the Other Network Research and Application," *Computer and Information Technology*, Vol. 14, No. 4, 2006, pp. 1005-1228.
- [4] Y. B. T and F. J. Chen, "The Reliability of P2P-Based Live Video Service Model and Analysis," *Computer Engineering and Science*, Vol. 1, 2006.
- [5] M. Liu, Z. C. Li, X. B. Guo and H. Deng, "An End-to-End Available Bandwidth Estimation Methodology," *Journal of Software*, Vol. 17, No. 11, 2006, pp. 108-116.