determine system performance. There are two generic service modes for cloud servers. In the first mode, the cloud server is primarily focused on serving the content already cached at the cloud storage system. Requests for content not in the cache is blocked until such content becomes cached. The cloud storage system updates its cache periodically to replace content without requests by content with requests a waiting. We call this the server mode .This is what is implemented in [1]. An alternative mode is the helper mode, in which the cloud server does not block any requests. For videos that are not cached, the cloud servers simply relay chunks from some peers to other peers, acting as a helper peer. The server mode when video population is large compared to cache size, and the helper mode when peer request rate is high compared to server bandwidth. We integrate these two modes into a single adaptive cloud downloading service.

2. Impact of Video Quality on User Engagement

Content popularity: There is an extensive literature on modeling content popularity and its subsequent implications for caching (e.g., [3, 4, and 5]). Most of these focus on the heavy-tailed nature of the access popularity distribution and its system-level impact. Analyzing the interplay between quality and engagement is orthogonal to this extensive literature. One question is to analyze if the impact of quality is different across different popularity segments. For example, providers may want to know if niche video content is more or less likely to be impacted by poor quality.

User behaviour: Yuet al. present a measurement study of a VoD system deployed by China Telecom [4] focusing on modeling user arrival patterns and session lengths. They also observe that many users actually have small session times, possibly because many users just “sample” a video and leave if the video is of no interest. User behaviours also have significant implications for VoD system design. For example, there are measurement studies of channel switching dynamics in IPTV systems, and understanding seek-pause-forward behaviours in streaming systems. As we mentioned in our browser minimization example for live video, understanding the impact of such behaviour is critical for putting the measurement-driven insights in context

P2P VoD: In parallel to the reduction of content delivery costs, there have also been improvements in building robust P2P VoD systems that can provide performance comparable to a server-side infrastructure at a fraction of the deployment cost (e.g., [5]). Because these systems operate in more dynamic environments (e.g., peer churn, low upload bandwidth), it is critical for them to optimize judiciously and improve the quality metrics that really matter. While our measurements are based on a server-hosted infrastructure for video delivery, the insights in understanding the most critical quality metrics can also be used to guide the design of P2P VoD systems.

Measurements of deployed video delivery systems: The net-working community has benefited immensely from measurement studies of deployed VoD and streaming systems using both “black-box” [12] inference and “white-box” [6] measurement. Our work follows in this rich tradition of providing insights from real deployments to improve our understanding of Internet video delivery. At the same time, It is believe that it have taken a significant step forward in qualitatively and quantitatively measuring the impact of the video quality on user engagement.

User perceived quality: There is prior work in the multimedia literature on metrics that can capture user perceived quality (e.g., [7) and how specific metrics affect the user experience (e.g., [8]). The first is simply an issue of timing and scale. Internet video has only recently attained widespread adoption and revisiting user engagement is ever more relevant now than before. Prior work depends on small-scale experiments with a few users, while this impact is based on real-world measurements with millions of viewers. Second, these fall short of linking the perceived quality to the actual user engagement. Finally, a key difference is with respect to methodology; user studies and opinions are no doubt useful, but difficult to objectively evaluate. Our work is an empirical study of engagement in the wild.

Engagement in other media: The goal of understanding user engagement appears in other content delivery mechanisms as well. Chen et al. study the impact of quality metrics such as bitrate, jitter, and delay on call duration in Skype [11] and propose a composite metric to quantify the combination of these factors. Given that Internet video has become main stream only recently, our study provides similar insights for the impact of video quality on engagement.

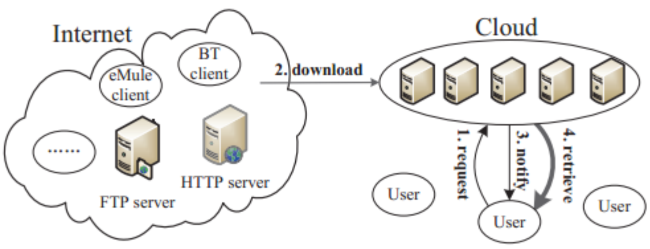

Fig. 1 - Principle of cloud download.

In recent years, the world wide deployment of cloud utilities provides us with a novel perspective to consider the above problem. This paper, propose and implement the cloud download scheme, which achieves high quality video content distribution by using cloud utilities to guarantee the data health and enhance the data transfer rate. The principle of cloud download is depicted in Figure 1. Firstly, a user sends his video request to the cloud (see Arrow 1 in Figure 1). The video request contains a file link which can be an HTTP/FTP link, a Bit Torrent /eMule link, and so on. Subsequently, the cloud downloads the requested video from the file link and stores it in the cloud cache (see Arrow 2 in Figure 1) to guarantee the data health of the video $\geqslant$1.0 (each video also has a duplicate in the cloud cache for redundancy). Then the cloud notifies the user (see Arrow 3 in Figure 1) and the user can usually retrieve his requested video (whether popular or unpopular) from the cloud with high data rate (see Arrow 4 in Figure 1) in any place at any time, via the intra-cloud data transfer acceleration. In practice, the cloud does not need to notify the user when his requested video is available. Instead, the user actively checks the down load progress by him and takes corresponding actions. That is to say, Arrow 3 in Figure 1 can also be rep laced by “3. Check”.

3. Design Challenges and Methodologies

Assumption

To keep the problem tractable, we make some simplifying assumptions for the models we study in this paper.

• The set of videos remains unchanged.

• Peer population is much larger than video population.

• All videos have the same size. The cloud storage can only store a small subset of the videos. In simulation, we do use heterogeneous videos length obtained from practical system.

• Peers have the same upload capacity. In most networks, peer’s download capacity is much more than upload capacity. We do not consider the constraint of download capacity in this work.

• Each peer issues one downloading request at any time. A multiple concurrent request model will be considered in future study.

• Peers are fully connected, forming a full mesh topology.

• The cloud server is able to replace any cached video with a new one instantly. In other words, assume that the time for video replacement can be ignored.

• The video size is large enough to be divided into an infinite number of small chunks.

Notations are given below

- N is the peer population. Since one peer only issues one request, the peer population is equal to the request population. The peer population for a video j is denoted by N_j .
- $\mathcal{M}$ is the set of all videos. The total number of videos is M.
- $\mathcal{K}$ is the subset of videos cached by cloud storage. The number of cached videos is K. $\mathcal{M}$ -- $\mathcal{K}$ is the set of videos not cached by cloud storage.
- Each video has a relative popularity η_j , i.e., the probability that the video is the target of any request. By definition, $\sum_{j=1}^{M} \eta_j = 1$.
- Peer upload capacity is U. The upload capacity of the server (that sources a video content) is U_0 .
- The (total) upload capacity of the cloud server(s) is H.
- File size for each video is denoted by F

The cloud server is able to serve multiple requests simultaneously. We assume a simple fair sharing scheduling strategy. In other words, the cloud server allocates its upload bandwidth evenly among all video requests being served.

Adaptive algorithm

The server mode and helper mode are having the strength and drawback. The cloud server needs to keep downloading new content to help peers so the helper mode wastes some P2P resources; however the server mode wastes the bandwidth resource of blocked peers. Design an adaptive algorithm to determine the service mode for each movie. The cloud server adjusts its strategy occasionally, by the use of following algorithm to determine the mode for each movie. The movies in helper mode have higher priority to be included into cloud storage. Then, consider the other movies in the order of decreasing peer population.

Algorithm1 Mode Switching Algorithm

for Each movie j not in $\mathcal{K}$. do
  if The active movie is less than K then
    Update cloud storage to add movie j by replacing any movie without request.
    $N' = N' + N_j$
  else
    if $\left(\frac{H}{N'} + a_j N_j\right) < N_j U$ then
      Use helper mode for movie j.
      $N' = N' + a_j N_j$
    else
      Keep blocking peers requesting for movie
      end if
  end if
end for

User requests a movie which they want to download. If the requested movie is not in the cache the request will be held in server mode. Requested movie is present in the cache means the helper peer acts as a server and used to download the requested video. Otherwise the peer request will be blocked.

The Management Center (MC) consists of a set of servers that have a number of responsibilities: (1) maintaining metadata in all media files, including the MD5 hash value, length, and popularity of each file; (2) authenticating users; and (3) maintaining the current status of participating peers in the P2P storage cloud, including the status of their local storage space. Routine tasks such as NAT traversal are also handled.

The Media Servers (MS) cooperate with the P2P storage cloud to feed a rich repository of videos to a large number of users. The objective of the media servers is to utilize limited server bandwidth in the best ways possible, and to maximize content availability and service quality in the P2P storage cloud.

The P2P storage cloud is derived from traditional P2P designs: all participating peers are organized into a mesh topology, and exchange information about the availability of video segments periodically.

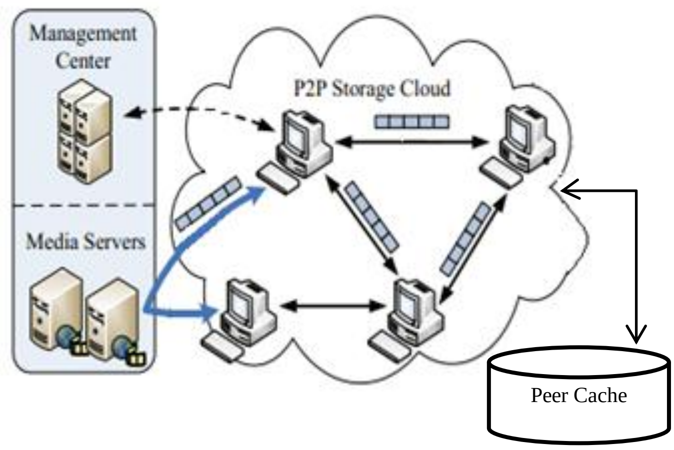

Figure 2 Architecture of cloud downloading service

The architecture of cloud downloading service is illustrated in Figure 1 the dotted arrow between the Management Center and P2P storage cloud represents the interaction between peers and the Management Center to manage the P2P storage cloud. The dark black arrow represents the gossip communication and video data distribution among peers i.e., peers are connected as full mesh topology. By concurrently downloading coded video blocks from multiple serving peers, a viewing peer can progressively decode and playback the video. Each peer is able to cache both original and coded segments of multiple videos. The dark blue arrows, the media server adaptively and proactively under-supplied videos to selected peers.

4. Conclusions

This project builds a theoretical model to analyse different strategies for a cloud downloading system. In particular, helper mode and server mode are used as abstraction of two different design philosophies using the cloud as peer or as server. The project analysis reveals that each strategy can be advantages, for certain operating scenarios. Helper mode wastes some server band-width, but is best at leveraging P2P capacity when request load is high. On the other hand, server mode is most efficient for dealing with large video population relative to the cache size. The design of mode switching algorithm is to choose the suitable service mode for different scenarios. This helps a cloud downloading system to optimize its design.

5. Feature Enhancement

The problem is with limited size cache. The drawback is about cache size because of its limited size. Future work is focus on the cache replacement and storage strategies. It is possible by replacing the unpopular video with coded segment.

REFERENCES

[1] Y. Huang, Z. Li, G. Liu, and Y. Dai, “Cloud download: Using cloud utilities to achieve high-quality content distribution for unpopular videos,” in Proc . ACM Multimedia, 2011, pp. 213–222.
[2] Y. Liu, Y. Guo, and C. Liang, “A survey on peer-to-peer video streaming systems,” Peer-to-Peer Netw. Appl., vol. 1, no. 1, pp. 18–28, 2008.
[3] H. Y. et al. Inside the Bird’s Nest: Measurements of Large-Scale Live VoD from the 2008 Olympics. In Proc. IMC, 2009.
[4] H. Yu, D. Zheng, B. Y. Zhao, and W. Zheng. Understanding User Behavior in Large-Scale Video-on-Demand Systems. InProc. Eurosys, 2006.
[5] X. Hei, C. Liang, J. Liang, Y. Liu, and K. W. Ross. measurement study of a large-scale P2P IPTV system.IEEE Transactions on Multimedia, 2007.
[6] Y. Huang, D.-M. C. Tom Z. J. Fu, J. C. S. Lui, and C. Huang. Challenges, Design and Analysis of a Large-scale P2P-VoD System. InProc. SIGCOMM, 2008.
[7] M. Claypool and J. Tanner. The effects of jitter on the peceptual quality ofvideo. InProc. ACM Multimedia, 1999.
[8] S. R. Gulliver and G. Ghinea. Defining user perception of distributed multimedia quality.ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP), 2(4), Nov. 2006.
[9] [21] D. Galletta, R. Henry, S. McCoy, and P. Polak. Web Site Delays: How Tolerant are Users?Journal of the Association for Information Systems, (1), 2004.
[10] Keynote systems.http://www.keynote.com.
[11] K. Chen, C. Huang, P. Huang, C. Lei. Quantifying Skype User Satisfaction. In Proc. SIGCOMM, 2006.