

A Private Cloud-Based Video Transcoding Architecture for Project-Based Learning Environment

Mohd Faisal Ibrahim, Rosanita Adnan, Zarina Zainol, and Nur Atiqah Sia Abdullah

Abstract—Project-based learning (PBL) is a teaching method that integrates real-world problems and technology into the curriculum. The use of mobile and video technologies in PBL has made the learning environment much more interactive. However, the gap between the richness of video content and the limited capabilities of mobile devices still remains a challenge. In order to solve this problem, video transcoding can be implemented. However, this solution may provide extreme CPU consumption and processing delay on the server's side. To overcome this issue, a private cloud-based video transcoding architecture is proposed. In this paper, the proposed solution was designed, implemented and compared with the server-based content switching method, and an experiment was conducted to evaluate the method in large-scale simulated workloads. The results show that the proposed architecture is capable of dealing with many concurrent users' access and capable of reducing the video downloading time.

Index Terms—Cloud computing, project-based learning, video transcoding.

I. INTRODUCTION

Project-based learning makes teaching and learning more effective and helps learners engage in group work activities with whatever software and technology they will be using. Through the use of multimedia-enabled mobile devices, the learning institution can create a meaningful lesson and implement projects for students using these technologies. It will allow students to use their mobile device to access learning resources, such as notes, audio and video, from a central learning repository.

In today's campus learning environment, students will access learning materials, especially videos, from different types of mobile devices, such as notebooks, tablets, and mobile phones. Due to the fact that all these devices have different CPUs, memory, displays, storage, and video processing capabilities, the video needs to be converted to resolve mismatches between the mobile device's decoding capabilities and the original video's encoding formats. This conversion process is known as video transcoding.

However, when various sets of mobile devices access video content simultaneously, it will introduce a high computational load at the server due to the considerable amount of

processing power required to transcode video content from one format to another. This will cause intolerable delays in the video transmission.

To solve this problem, a cloud-based video transcoding solution can be deployed. However, there is a significant barrier to deploying a cloud computing solution, which is the issue of lack of control [1], security [2], [3], privacy [2], [3] and trust issues [3].

To overcome the as-mentioned problems, this paper proposes a private cloud-based video transcoding architecture that performs the online-transcoding operation. This design philosophy provides the content provider (especially learning institutions) full control of their learning repository since the cloud infrastructure is entirely owned, managed, and monitored by them.

II. RELATED WORK

This section provides an overview of some of the related works in the area of cloud- and server-based video transcoding.

A. Public Cloud-Based Video Transcoding

A number of companies have started to provide public cloud-based video transcoding as a service. The Google computer engine (GCE) [4] transcoder [5] enables users to encode and transcode video into versions that can be viewed or played back on their mobile devices, such as smartphones, tablets, and PCs. It handles all the features of the transcoding process via adaptive bitrate streaming or progressive download. Amazon's elastic transcoder [6] is another video transcoder offered by Amazon's elastic computer cloud [7], a company that provides cloud-based video transcoding services. The transcoder can efficiently and seamlessly scale to match the video transcoding workload and able to cope with the request of large volumes of concurrent video files.

However, this approach offers learning institutions inadequate control because the cloud infrastructure is totally owned and controlled by the cloud service provider. Cloud customers have very limited control over the cloud infrastructure and therefore, they will have minimum security control and data protection.

B. Private Cloud-Based Video Streaming Transcoding

Most studies have focused on scheduling policies and resource allocation for video streaming transcoding in the cloud environment. Kllapi *et al.* [8] looked at how to optimize data flow scheduling by minimizing the operation cost and completion time. Ma *et al.* [9] presented a solution that can improve cloud-based online transcoding for dynamic adaptive

Manuscript received January 20, 2018; revised March 16, 2018. This work was supported in part by the ARAS Research Grant under Grant No. 600-IRMI/DANA 5/3/ARAS (0189/2016).

Mohd Faisal Ibrahim, Rosanita Adnan, Zarina Zainol and Nur Atiqah Sia Abdullah are with the Faculty of Computer and Mathematical Sciences, UiTM, Shah Alam, Selangor, Malaysia (e-mail: faisal@tmsk.uitm.edu.my, rosanita@tmsk.uitm.edu.my, zarina@tmsk.uitm.edu.my, atiqah@tmsk.uitm.edu.my).

streaming over HTTP by proposing a dynamic scheduling methodology. Jokhio *et al.* [10] have proposed a prediction-based mechanism to reduce processing delays by allocating and deallocating virtual machines (VMs) to perform the transcoding task. Jinlai [11] implemented a cloud-based interactive mobile live streaming video for learning environments. In his solution, the class lecture was captured live through a webcam and distributed to the student devices as two-second video clips.

All of these studies have focused on video streaming. In this paper, we focus on online video transcoding for HTTP web servers using a progressive download approach.

III. PRIVATE, PUBLIC, AND HYBRID CLOUDS

Cloud computing is the delivery of on-demand network services that share a pool of physical and virtual resources over the Internet [12]. Cloud computing solutions can be implemented as public, private, and hybrid clouds. In a private cloud, its infrastructure is operated exclusively by a single business or organization and managed internally by the organization themselves or by a third party [12]. A public cloud is an infrastructure that is used by the public, but it is owned, managed and operated by a third-party cloud service provider or organization [12]. A hybrid cloud is an infrastructure that combines public and private clouds [12].

IV. PROJECT-BASED LEARNING

Project-based learning (PBL) is an instructional approach that offers learning around projects and focuses on student centered inquiry of problems in a real-world setting [13]. In PBL, students will explore real-world problems and challenges, working together in a small group toward achieving a common goal with the lecturer will act as a facilitator by setting project goals and providing guidelines and resources. PBL provides an effective way for students to use digital technology. By using technology in PBL, students can access a vast array of ideas or information and collaborate more effectively with team members.

V. TYPES OF VIDEO DELIVERY METHODS

A. Progressive Download

In this method, the user requests a video from the HTTP web server using the HTTP GET request method and the server will send the whole file over HTTP connection. The user can only play the video when the downloading process of the entire video file has been completed. The downloaded video is stored locally in the user's device.

B. Streaming

Streaming video on the other hand provides a video delivery mechanism using specialized streaming servers over the real-time messaging protocol (RTMP) or real-time streaming protocol (RTSP). The video is sent via a streaming server without the file ever being downloaded to the user's device. The user can start playing the video file before the

entire file has been transmitted.

VI. VIDEO TRANSCODING

Although the use of mobile devices in video downloading has become increasingly popular among students, it still has many challenging issues. One of the main issues for heterogeneous mobile devices is their video processing capabilities, which is different for each mobile device manufacturer. A video format or feature that can be supported by a certain mobile device is unlikely to be supported by another device model. Therefore, the unsupported video format has to be transformed into one of the formats, in which the device can support it before being viewed or displayed. The process of converting a video from one format to another format is known as video transcoding [14]. The goal of video transcoding is to change the format of the original video without changing the content.

A. Location of Video Transcoding

One of the main design issues is to determine whether the transcoding process should be conducted on the server or proxy side. Each method has its advantages and disadvantages in terms of the design choice.

1) Server-based transcoding

In this approach, the transcoding process will be done at the server side. The drawback of this approach is that an additional computing power is required on the server side for transcoding process. However, the distribution of the computation load with this approach can be resolved by implementing content switching technology.

2) Proxy-based transcoding

In this architecture, an intermediary computational entity performs the adaptation tasks on the fly, transparently to the servers and clients. The proxy will send a request (to the server) and intercepts any reply (from the server), decide whether to perform the transcoding process and submits the adapted content to the client [15]. This proxy-based approach is known as a technology that provides caching capabilities. If the data is already in the cache memory, the proxy will immediately send the data. This will reduce the processing load on the origin server. However, this approach has several disadvantages, such as a lack of control, the server might use outdated object stored in a cache and a slower processing load distribution between the origin server and the proxy.

3) Cloud-based transcoding

In this approach, the transcoding will be executed on the cloud servers. The advantage of cloud-based transcoding is that the transcoding server can efficiently assign video resources in time-varying capacities and execute several transcodings simultaneously to improve performance. The transcoding server can be implemented in a private or public cloud infrastructure. The public cloud has the disadvantages of lack of control, privacy, and security issues, while the private cloud has several drawbacks, such as cost, scalability, and on-site maintenance.

B. Timing of Video Transcoding

The execution of the transcoding process can be carried out

either online or offline. These techniques can be classified as follows:

1) *Offline video transcoding*

In this approach, the video is transcoded before the delivery process begins and stored in a different format, quality and processing requirement [16]. Only when the client requests for data, the most suitable version will be selected depending on the user’s device or capabilities. However, the limitation of this approach is that it requires a massive storage space to maintain all the various adapted video formats on the server.

2) *Online video transcoding*

In this approach, the video is processed and delivered on the fly, also known as dynamic [16]. This approach, however, requires a massive amount of computing resources for video transcoding and produces extreme processing delays [17].

C. *The Issues of Video Transcoding*

In today’s technology-rich learning environment, students regularly need to download learning materials, such as video, from an e-learning server via different types of devices, such as personal computers, notebooks, tablets, and mobile phones. As all these devices have diverse capabilities for video processing and the mobile devices have limited computing resources, these videos need to be transcoded into different formats so that they can be adapted to different types of user devices. However, transcoding is a computer-intensive operation. When a large number of students access the video simultaneously from their mobile devices, the server’s CPU utilization will drastically increase. When the CPU utilization is high, the downloading time will also increase. In order to provide time-efficient transcoding services, the content providers need to use a number of high-end transcoding servers to handle the transcoding process under different load conditions.

VII. CLOUD-BASED VIDEO TRANSCODING ARCHITECTURE

Fig. 1 shows the proposed cloud-based video transcoding architecture. It consists of a web server, transcoding server, a load balancer, and content switching policies.

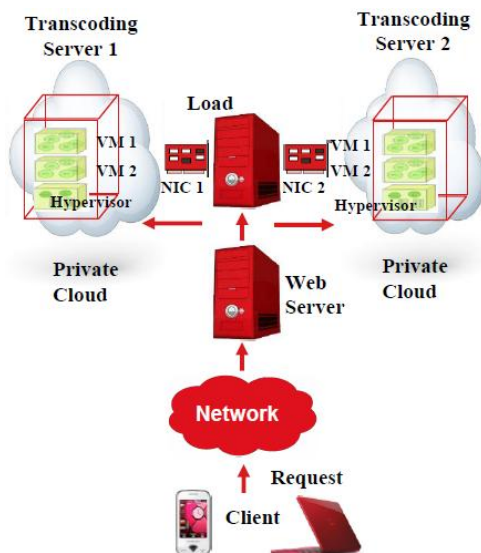


Fig. 1. The cloud-based video transcoding architecture.

When the user requests a video, the request will be sent to the web server. The web server will forward the request to the load balancer. When the request reaches the load balancer, it employs the associated content switching policies to that particular request. The load balancer provides a content switching mechanism and it will decide whether to forward the video request to Transcoding Server 1 or Transcoding Server 2. This decision will be based on a rotational basis because in the proposed solution, a round robin algorithm was implemented. The first video request grants access to the first available video transcoding server and the second video request will be forwarded to the second video transcoding server. If there is a third video request, this request will be routed back to the first transcoding server and the loop will continue every time there is a new video request.

When the video transcoding sever receives the video request from the web server, it will proceed by converting the data based on the frame rate, bit-rate, or screen resolution. After the transcoding process is completed, the transcoded video will then be forwarded to the web server and the web server will forward it to the requested client.

The web server was implemented using Microsoft SQL Server 2008. The web server was installed on Intel Pentium 4 Hyper-Threading Technology 3 GHz with 4 GB RAM running on Microsoft Windows Server 2016. Two physical servers for the transcoding server were installed on Intel Quad-Core Xeon 3.73 GHz with 8 GB RAM running on VMware vCenter Server 5.5. The video transcoding server was implemented using an open source library/tool called FFmpeg [18], which provides a program for decoding and encoding any video format. HAProxy was used as the load balancer and installed on a 2.80 GHz Intel Pentium4 desktop PC with 8 GB RAM running on FreeBSD.

VIII. EXPERIMENTAL EVALUATION

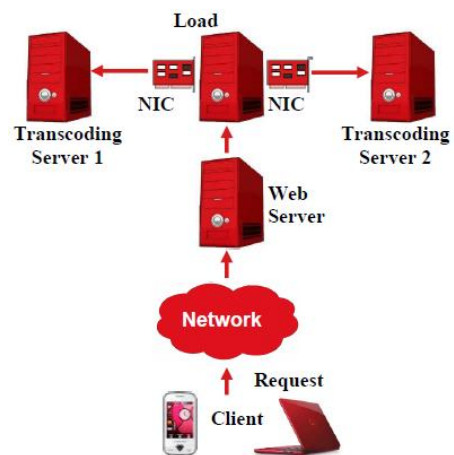


Fig. 2. Server-based video transcoding using a content switching method.

An experiment was conducted to evaluate the impact of the cloud-based technique in terms of the downloading time. The results of the proposed approach was compared with the results obtained from server-based content switching, which is currently being implemented in our faculty, as shown in Fig. 2. The hardware and software used for the server-based content switching components were the same as what we have

used in the cloud settings except that the two transcoding servers were configured on Windows 7 instead of VMware vCenter Server 5.5. The function of the load balancer was to provide a content switching mechanism so that it can balance the video request between the two transcoding servers in order to spread the load.

A. Experimental Design and Setup

A program called ThreadUploader was developed to simulate hundreds of concurrent virtual users requesting a video file in order to evaluate the proposed solution performance under a massive load. This program will allow multiple video files to be downloaded simultaneously from the web server by making use of two or more “threads” of execution. Fig. 3 shows how to simulate concurrent users using ThreadUploader.



Fig. 3. ThreadUploader.

The following equation is used to calculate the computing time involved in the downloading operations:

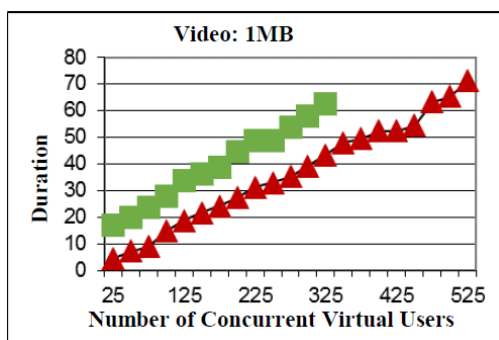
$$DL_t = DL_s - DL_e \tag{1}$$

where DL_t is the downloading time, DL_s is the time when the downloading operation started and DL_e is the time when the downloading operation ended.

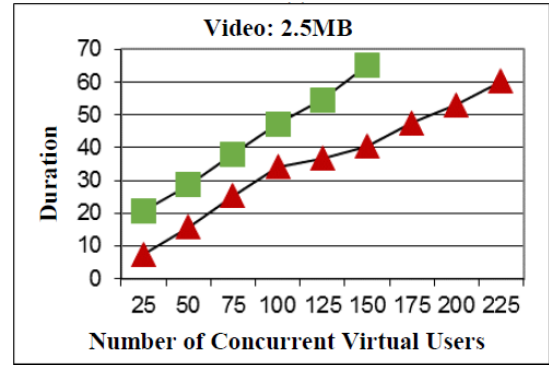
B. Results and Analysis

Fig. 4 shows the downloading time for 3 videos with different file sizes: a) 1 MB, b) 2.5 MB and c) 5 MB. The y-axis shows the time taken (in second) for downloading each video and the x-axis denotes the number of concurrent virtual users downloading the video file.

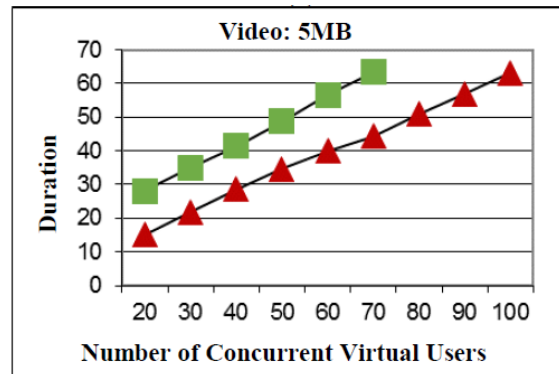
When using the cloud-based solution, not only the video downloading time can be reduced, but the maximum number of concurrent connections for video requests can also be increased. Fig. 4(c) shows that by using the proposed solution, 100 concurrent users are able to download a 5 MB video file compared to just 70 users when using the server-based approach. These figures increase to 225 users (Fig. 4(b)) and 525 users (Fig. 4(a)) when the proposed solution is used.



(a)



(b)



(c)

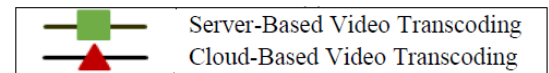


Fig. 4. The video downloading time for: a) 1 MB, b) 2.5 MB and c) 5 MB video file.

As expected, the cloud-based transcoding operation was capable of reducing the video downloading time, since the task of video transcoding can be divided among four virtual machines in two separate physical video servers, thus allowing more users to download the video concurrently at a slightly faster rate.

IX. CONCLUSIONS

This paper designs and implements a cloud-based video transcoding architecture for non-streaming video in a campus learning environment. The proposed solution was compared with the server-based content switching method. It was implemented using the same hardware specification as the server-based approach. An experiment was conducted to evaluate the proposed method under an extensive simulated workload. The results indicate that the proposed architecture is capable of dealing with many concurrent users’ access and is capable of reducing the downloading time.

A weighted round robin algorithm will be considered as future work in which the server will consider the resource capabilities of the virtual machine and allocates a certain weightage to each virtual machine based on number of tasks and capacity.

ACKNOWLEDGMENTS

The authors acknowledge with gratitude the great support from the management of Faculty of Computer and Mathematical Sciences, UiTM Shah Alam, JPBSM of UiTM Shah Alam, Malaysia Research Assessment (MyRA) and the

Institute of Research Management and Innovation (IRMI),
UiTM.

REFERENCES

- [1] Z. Mahmood, "Data location and security issues in cloud computing," in *Proc. the 2nd International Conference on Emerging Intelligent Data and Web Technologies*, 2011, pp. 49-54.
- [2] Z. Xiao and Y. Xiao, "Security and privacy in cloud computing," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 2, pp. 843-859, 2013.
- [3] Y. Sun, J. Zhang, Y. Xiong, and G. Zhu, "Data security and privacy in cloud computing," *International Journal of Distributed Sensor Networks*, vol. 10, no. 7 pp. 1-9, 2014.
- [4] Google Compute Engine. (June 19, 2015). [Online]. Available: <https://cloud.google.com/compute/>
- [5] Google Compute Engine Transcoder. (June 19, 2015). [Online]. Available: <https://cloud.google.com/solutions/media/use-cases/video-encoding-transcoding/>
- [6] Amazon Elastic Transcoder. (October 1, 2015). [Online]. Available: <https://aws.amazon.com/elastictranscoder/>
- [7] Amazon Elastic Compute Cloud. (October 1, 2015). [Online]. Available: <http://aws.amazon.com/ec2/>
- [8] H. Kllapi, E. Sitaridi, M. Tsangaris, and Y. Ioannidis, "Schedule optimization for data processing flows on the cloud," in *Proc. the 2011 ACM SIGMOD International Conference on Management of Data*, 2011, pp. 289-300.
- [9] H. Ma, B. Seo, and R. Zimmermann, "Dynamic scheduling on video transcoding for MPEG DASH in the cloud environment," in *Proc. the 5th ACM Multimedia Systems Conference*, 2014, pp. 283-294.
- [10] F. Jokhio, A. Ashraf, S. Lafond, I. Porres, and J. Lilius, "Prediction-based dynamic resource allocation for video transcoding in cloud computing," in *Proc. the 21st Euromicro International Conference on Parallel, Distributed and Network-Based Processing*, 2013, pp. 254-261.
- [11] Y. Jinlai, "The building of mobile interactive learning system based on cloud computing," *International Journal of Digital Content Technology and Its Applications*, vol. 6, no. 8, pp. 266-274, 2012.
- [12] P. Mell and T. Grance. (September 2011). The NIST definition of Cloud Computing, (Artikelnr 800-145), National Institute of Standards and Technology. [Online]. pp. 2-3. Available: <http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf>
- [13] M. C. English and A. Kitsantas, "Supporting student self-regulated learning in problem- and project-based learning," *Interdisciplinary Journal of Problem-Based Learning*, vol. 7, no. 2, pp. 128-150, 2013.
- [14] G. Gao, Y. Wen, and C. Westphal, "Resource provisioning and profit maximization for transcoding in clouds: A two-timescale approach," *Journal IEEE Transactions on Multimedia*, vol. 19, no. 4, pp. 836-848, 2017.
- [15] A. R. Kaced and J. C. Moissinac, "Multimedia content authentication for proxy-side adaptation," in *Proc. the International Conference on Digital Telecommunications*, 2006, pp. 62-71.
- [16] D. K. Krishnappa, M. Zink, and R. K. Sitaraman, "Optimizing the video transcoding workflow in content delivery networks," in *Proc. the 6th ACM Multimedia Systems Conference*, pp. 37-48, 2015.
- [17] L. Wei, J. Cai, C. H. Foh, and B. He, "QoS-aware resource allocation for video transcoding in clouds," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 1, pp. 49-61, 2017.
- [18] FFmpeg. (January 15, 2015). [Online]. Available: <https://www.ffmpeg.org/>



Mohd Faisal bin Ibrahim is a lecturer in the Faculty of Computer & Mathematical Sciences, Universiti Teknologi MARA, Malaysia. He received his Ph.D. in Electrical Engineering at the Faculty of Electrical Engineering, Universiti Teknologi MARA (UiTM) in 2017. His research interests include the Internet of things (IoT), wireless sensors, cloud-computing, and content adaptation.