Enhancement of QoS in CloudFront through Optimization of Video Transcoding for Adaptive Streaming .

Rupal Kharya[1], Dr. Arpana Rawal[2]

[1]*M.Tech Computer Network Bhilai Institute of Technology, Durg (C.G.) – 491001 India*
[2]*Prof. Department of Computer Science and Engineering Bhilai Institute of Technology, Durg (C.G.) – 491001 India*

---***---

Abstract— *In present day to day, content is shifting paradigm towards additional interactive media, principally video. Availableness of assorted transmission formats, and devices in mobile, IPTV conjointly demands for specific necessities on content delivery networks, parturition importance on value effectiveness of feeding content to ever growing world net traffic, during this paper we tend to discuss numerous strategies of enhancing the value potency, through use of accommodative video streaming and proper orchestration of storage, information measure and computing resources. Main objective is to formulate associate optimization downside from the constant inputs, like transcoding value, resource allocation (caching, bandwidth) and quantify ability. Particularly, we tend to outline associate improvement downside, by analyzing a three-path exchange off between the storing, transcoding, and transmission capability value, at each edge server. At that time, we tend to receive a two-stage thanks to upset diagnostically abstract thought of the arrangement i.e the best transcoding setup conjointly, storing area allotment, separately, for every edge server. At last, we tend to check our answer throughout broad recreations. The outcomes demonstrate that, our approach accomplishes immense value contrasted and existing techniques utilized as a section of content delivery systems, is additional, we tend to likewise find the best methodology and its blessings may be influenced by a summation of framework parameters, together with the cost of varied assets, the bounce separation to the start server, the Zipf parameter of clients' demand styles, and therefore the settings of assorted bit rate forms for one portion.*

Keywords— *Media Cloud; reconciling Video; Resource Allocation; Content Delivery; Bandwidth.*

1. Introduction

Adaptive video streaming administration, inferable from the present entrance of quick net get to and also the increasing infamy of sight and sound applications over completely different end-gadgets, is fuelling associate exponential development of net activity as currently. Especially, Cisco's VNI report anticipated that, the worldwide video movement can entirely command the web by 2018, representing 80-90 pc of the combination web utilization [1]. In reality, such exceptional live of spilling data is eaten up by an appointment of heterogeneous end-gadgets together with Tv, PC, tablet, and cell phones, that need numerous video arrangements, resolutions, and bitrates. These pattern postures noteworthy difficulties to existing video profit suppliers. to begin with, the large scale dissemination of video substance demand monumental information transmission assets, significantly a midst peak times. This may be not very maintained by the customary static instrument of system quality arrangement [3]. Second, with regards to on-request versatile streaming, video content has varied variations to suit the system condition and also the utilization of assorted end-gadgets. Therefore, it's necessary to possess calculation and capability assets to transcode video and reserve them at some middle hubs within the system [4]. At long last, the absence of coordination on those resources prompts to poor price effectiveness and restricted ability. As of late, media cloud [5][6] has up as a promising structure to profitably circularize versatile video spilling administrations. Specifically, media cloud shapes a flexible virtual content delivery organized by powerfully allocation virtualized resourcing and capability assets to a summing up of edge servers, in reaction to the unsteady application requests. on these lines, such cloud-based style offers an opportunity to primarily diminish the operational price of conveyance versatile video gushing, by keenly organizing varied varieties of virtualized assets in associate on-request manner.

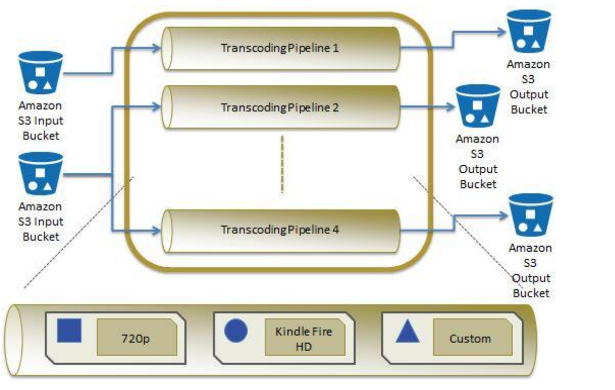

Figure 1: Amazon Transcoder

Amazon Elastic Transcoder is made exploitation the measurability and adaptability of different Amazon net Services. It runs your transcoding jobs victimization the Amazon Elastic reckon Cloud (Amazon EC2).It permits you to complete massive transcoding jobs quickly and faithfully, is made to figure with content you store in Amazon easy Storage Service (Amazon S3), thus you have sturdy and price effective storage for vast libraries, or tiny ones.

You'll be able to even get notified concerning the standing of your transcoding jobs via Amazon easy Notification Service (Amazon SNS).

It produce a transcoding pipeline that specifies the input Amazon S3 bucket, the output Amazon S3 bucket, and an AWS Identity and Access Management (IAM) role that's employed by the Transcoder to access your files.

Produce a transcoding job by specifying the input data, computer file, and transcoding planned to use (you will select from a collection of pre-defined transcoding presets as an example 720p or produce your own custom transcoding preset. Optionally, you'll be able to specify thumbnails and job specific transcoding parameters like frame rate and backbone,

The status of transcoding jobs, manage your transcoding jobs by stopping, beginning or cancelling them.

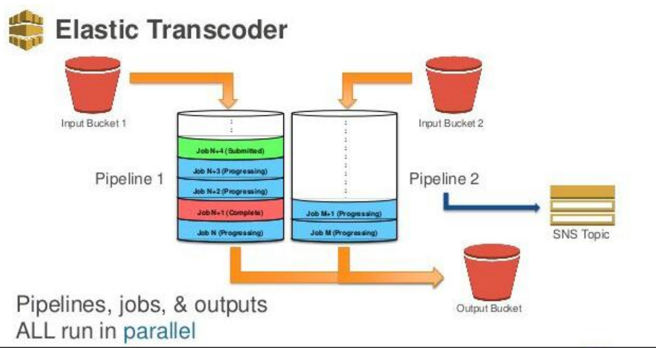

Figure 2 :Amazon Elastic Transcoder

Pipelines are queues that manage your transcoding jobs. Once you produce a job, you specify the pipeline to that you wish to feature the job. Elastic Transcoder starts process the roles in a very pipeline within the order within which you value-added them. There are alternative jobs in a very pipeline once you produce employment; Elastic Transcoder starts process the new job once resources are accessible. A pipeline will method over one job at the same time, and also the time needed to finish a job varies considerably supported the scale of the file you are changing and therefore the job specifications. As a result, jobs do not essentially complete within the order within which you create them.

You can quickly pause a pipeline therefore it stops process jobs. This can be useful if you would like to cancel one or additional jobs, which we do solely till Elastic Transcoder starts processing the jobs.

2. Literature Review

Media cloud, as a rising worldview to effectively distribute video spilling administrations, has force in noteworthy attention. Zhu et al. [5] projected a media-edge cloud style, within which reposition and registering assets square measure selected at the sting of media cloud, to perform consumer driven video handling what is a lot of, QoS (Quality of Service) adjustment for numerous end-gadgets. Dıaz-Sanchez et al. [10] planned the media cloud as associate degree open middleware for WAN (Wide space Network) scale content administration and conveyance. Jin et al. [11] utilized a summing up of virtual machines to flexibly fabricate a virtual content conveyance overlay that is suitable minimizing the routing price whereas fulfilling the desired QoS. Wang et al. [12] displayed Cloud-Assisted Live Media Streaming (CALMS) as a nonexclusive cloud-based structure to encourage relocating live media streaming from the foundation server to the media cloud.

Wang et al. [13] examined a perfect cloud-helped versatile video spilling set up, that numerous video variations square measure flexibly pre-brought and put away within the media cloud. All the additional as currently, Wen et al. [6] condensed these takes a trial at video spilling over media cloud, beside a line of key define standards, into a brought along superimposed administration system.

The arrangement on varied assets, joined of the key define contemplations in media cloud, likewise had been intensively researched in varied different application things. Qazi et al. [14] familiar quality administration module with alter the reckoning quality utilization at every mid-tier nodes, within the setting of programming characterized organizing. Grandl et al. [15] organized completely different assets by means that of a heuristic pressing approach for bunch computer hardware, to contour the utilization finishing time. Zhou et al. [16] designed up a visually impaired quality booking calculation for versatile media cloud, by description a restricted time skyline sweetening issue and statically steering consumer solicitations to fitting media profit hubs. Wu et al. [17] received on-request cloud quality provisioning to beautifully meet the dynamic and heavy quality requests of video on-request edges, by concentrating on the virtual machine use what is a lot of, stowage rental. This work plainly varies from those connected appearance into essentially in 2 ways that to start with, we have a tendency to think about multi-asset orchestration within the media cloud, with the goal to advance the mixture operational price caused by transference on-request versatile video spilling. the particular circumstance and therefore the goal of such coordination issue aren't quite constant as those current inquires concerning. Second, we have a tendency to build up associate degree expositive structure to regulate the three-path exchange off between capabilities, reckoning additionally, transfer speed assets, and infer a shut frame arrangement.

Interestingly, most past works simply focused around one or 2 quality components. Plus, few of them came up with shut form arrangements. To the simplest of our insight, this work is that the principal endeavor to with efficiency analyzes the best numerous resources designing for effective versatile video spilling appropriation over the media cloud.

3. Problem Formulation

In this section, we tend to initial describe the system design to supply necessary background. Then, we tend to illustrate the basic trade-off via a motivation example. Following that, we tend to discuss four system models. Finally, exploitation those models, we tend to formulate a unnatural optimization downside for multi-resource allocation. For clarity within the following discussion, we tend to summarize those key notations as shown table I.

To accomplish cost-proficient on-request versatile spilling administrations, it's basic to reduce the combination operational value caused by composition varied virtualized assets for media cloud profit provider. In fact, such operational value is chosen by the organization on reserving, transcoding and information transmission assets.

Table 1 Video Fragment

Symbol	***Definition***
G(V,E)	*Cloud representation as graph, V as nodes and E as Edge*
D_i	*Hop distance*
n_{br}	*Bit rate versions*
n_s	*Number of segments*
S(i,j)	*Segment of rank i, and j'th bit rate*
b_t, b_m, b_h	*Total, Mean and Max bitrates*
$P(s_{ij})$	*Probability of expecting S(i,j)*
P(j)	*Prob. of j'th bit rate version*
C(i), x(i), h(i)	*Cache, bitrates versions, and segment count at node i.*
w_c,w_b,w_t	*Caching, bandwidth and transcoding costs*

3.1 System Architecture

Figure 3 shows a scientific end-to-end read of delivering on-demand adjective streaming over media cloud. It primarily consists of 3 elements, together with content suppliers, end-users with totally different devices, and media cloud service suppliers.

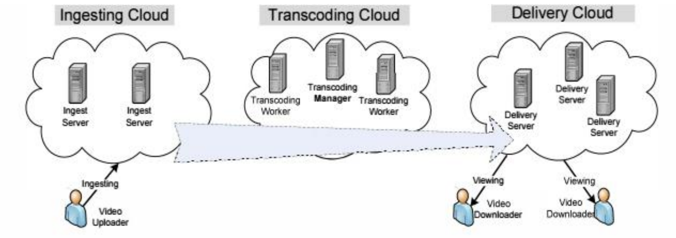

Figure 3: System Architecture

Content Providers Content suppliers distribute all video fragments with all profit capable bit rate variants by suggests that of their birthplace server [18]. Those video fragments are sent to end-gadgets in varied arrangements with completely different bitrates upon shopper demands, in the course of a media cloud, wherever each edge server is an administration section purpose connected with a get to network to total shopper demands. Media Cloud Service suppliers pull framework assets from typical cloud profit suppliers, and within the meanwhile, provide metered media administrations to end-clients [6] [19]. Specifically, they will powerfully instantiate virtualized storing what is a lot of, transcoding assets, and opt for a way to assign them on each edge server. Consequently, some shopper solicitations will be served squarely by edge servers in light-weight of in-system reserving on the opposite hand web transcoding [20] [21]. this offers Associate in Nursing open door to basically spare the info transfer capability value of convincing versatile streaming from substance suppliers to end-clients.

Specifically, from one viewpoint, noteworthy information transmission value will be spared, by allotting a lot of store areas to a foothold server to form the substance abundant nearer to end-clients. Additionally, by moving some transcoding assets to the sting server, the neighborhood store hit proportion and also the information transfer capability utilization will be additional affected forward. Then again, bound operational expenses are going to be caused by utilizing either storing or transcoding assets. Such value might overpower the investment on the transmission capability utilization. Afterwards, there's an opportunity to reduce the combination operational value by watching the three-route exchange off between the caching, transcoding and information measure value.

3.2 Computation model

We model the topology of media cloud as AN aimless graph G = (V, E), wherever V denotes the set of edge servers, n = |V | denotes the full variety edge servers, and E denotes the set of network links between those servers.

For simplicity, this work considers there's just one origin content supply s ∈ V, with Di denoting the hop distance from the node i to s. this may be simply extended to the case of multiple different sources, by one by one generating the answer for every of them mistreatment the identical technique. This topology G (V, E) is constructed on high of multiple resources. Specifically, every node v ∈ V is related to each caching and transcoding resources, and every link e ∈ E is related to information measure resources. The consumption on totally different resource part can incur a corresponding quantity of price, in an exceedingly pay-per-use manner.

We contemplate there square measure in total ns totally different video segments to be distributed, wherever every phase has nbr totally different bit rate versions starting from the best bit rate b1 to the littlest bit rate br. we have a tendency to assume the length of every phase is that the same on the average, and everyone those nbr bitrates square measure of times accessed (i.e., we have a tendency to don't contemplate the versions that square measure rarely used). As we have a tendency to concentrate on the adaptive streaming service, ns is usually larger than one. As such, we've got overall nbr ∗ns totally different versions of all video segments. Every phase is denoted as S(i,j) , wherever i = one, ..., ns, refers to its quality rank, and j = 1, ..., r, refers to its bit rate rank.

We assume the mass user requests towards those nsdifferent segments follow the Zipf-like distribution [22], [23].We additionally adopt the idea from [24], [25], that the user requests severally make every edge server within the media cloud, following the freelance reference model [26]. Thus, the chance of requesting the j-th bit rate version of the i-th standard phase S (i,j) at a grip server is,

$$P(S(i,j)) = \frac{p_j / i_\alpha}{\sum_{1}^{ns} (1/k_\alpha)} = p_j / i_\alpha * H_{ns,\alpha} , \ni i = 1..n_s$$

Equation 1 - Probability of j'th bitrate version on i'th segment

Where H is that the m-th generalized harmonic variety, and α is that the Zipf exponent characterizing the distribution, wherever α should be positive. an outsized α indicates a lot of requests on standard segments and fewer requests on less-traveled ones. Typically, α is between zero.5 and 1.5, we tend to adopt the idea from [25] that, α are often closed to 1, however normally not specifically up to one.

Under this theme, every edge server strategically caches all bit rate versions for some high standard video segments, and solely keeps the best bit rate versions for a few alternative less standard segments, strained by the allotted cache area. Mathematically, we tend to introduce a parameter x(i) ∈ [0, c(i) / bs], wherever c(i) is that the allotted cache area for node i, and bachelor's degree is that the total size of all bit rate versions of 1 video section. below this configuration, node i caches all bit rate versions of the x(i) hottest segments, and solely the best version of the segments, whose quality rank is between x(i) and h(i) . During this case, h(i) are often calculated by,

h(i) = x(i) + (c(i) – x(i)*bs) / bh

In this approach, there would be 3 doable states to serve every user request at node i, as follows.

- There is a particular Cache Hit, if the requested video section is accessible within the native cache area. during this case, this request are often directly served, while not acquisition any further expense on information measure or computing resources.
- There may be a Transcoding Hit, if the recognition rank of the requested video section is within the vary of x(i) and h(i) , and it's not the best bit rate version. During this case, this request still is often served domestically, supported period of time transcoding, which involves the usage of computing resources at the sting server. In apply, once the native transcoding hit happens, it'll appraise whether or not the native transcoding price is over retrieving the requested section directly from the origin server, and decide the operation with lower price.
- There may be a native Cache Miss, if the requested video section isn't among the highest h i ones. During this case, we tend to adopt a straightforward however sensible mechanisms that, all that cache misses are directly forwarded to the origin server. This prices an explicit quantity of information measure resources to transmit the desired section from the origin server to the sting.

So we tend to get combined case hit quantitative relation likelihood as followed for hit and transcoding hit:

$$P_i^{hit} = \sum_{k=1}^{h(i)} \sum_{j=1}^{n_{br}} P(S(k,j)) = H(h(i),\alpha) / H(n_s,\alpha)$$

$$P_i^{tr} = \frac{(1-p1)*(H(h(i),\alpha) - H(x(i),\alpha))}{H(n_s,\alpha)}$$

3.3 Analytic solution formulation

First, for a given cache area c(i), we have a tendency to derive the optimum transcoding configuration x⋆(i) with relation to c(i) . Second, we have a tendency to substitute x⋆ (i) into the initial objective operate (9), and additional acquire a closed-form of the optimum allotted cache area c⋆(i) . Finally, we have a tendency to use the obtained results to on paper quantify the performance gain. The weight ages through empirical observation taken will offer the value estimations.

4. Methodology

4.1Video Transformation Subsystem

The Video Transformation Subsystem is composed by the Transcoder and the Segmenter modules, as illustrated in Figure 4.

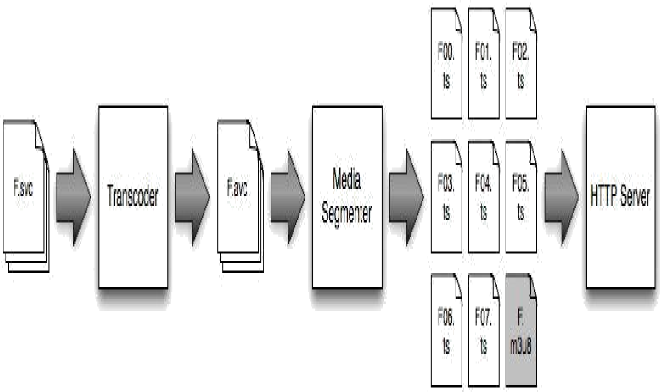

Figure 4: Distribution Side -Video Transcode and Partitioning.

The task of the Transcoder is to make a bit stream reordering conversion of the H.264/SVC encoded media in order to feed a SVC format with the desired quality level. This works by converting all layer representations. The video needs to be encoded with a special inter layer prediction flag with the value equal to one.

The resulting H.264/SVC converted (and multiplexed with segment) to a MPEG-2transport stream feeds the Segmenter module that partitions it into file segments of relative equal duration for distribution over HTTP. The Segmenter also creates a Playlist (an index) file containing references to the individual media segments (relative URLs).

4.2 Playlist File

The Playlist file is saved as .mp4 playlist, which is a format extension of the .m3u format used for MP3 playlists. The creation of this file is important since it contains the information necessary for the play-out session. Each entry of the file encloses a specification that can be of the following types:

- Local pathname relative to the m3u file location;
- Absolute local pathname of the files to play
- An URL of the file to play;

The process of downloading the video file at the end of AWS cloud utility transcoder is illustrated in figure 5, whose output representation is available as playlist file streams.

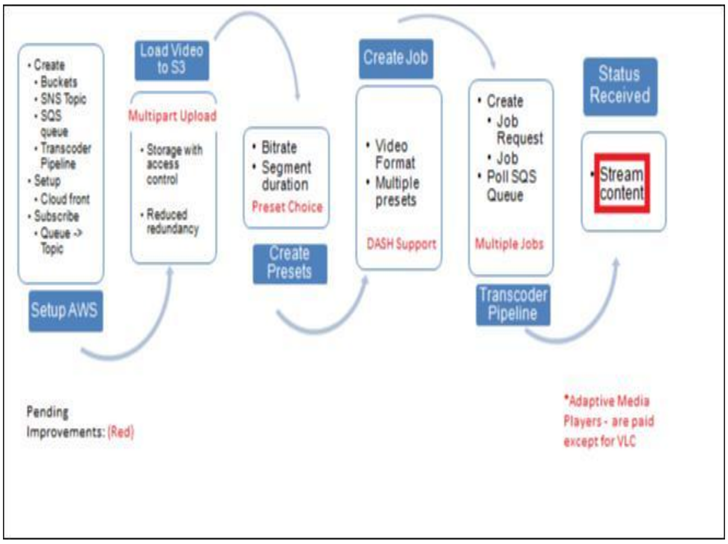

Figure 5: Preset Statuses in Adaptive Video Streaming

- In this status we managed to transcode a multiple video file, using the above flow.
- Multiple transcoding jobs and Preset selection are future improvements. We converted video for now into standard Apple HLSv Format.
- DASH support can be implemented next, and comparative evaluation can be done.
- Video players for adaptive streaming are licensed and charge approx.300$ (JWPlayer), we used VLC media player to test video output received.

- Multiple jobs and the optimality of transcoding need to be evaluated using the scheme, in terms of size, time/duration and cost.

5. Expected Results

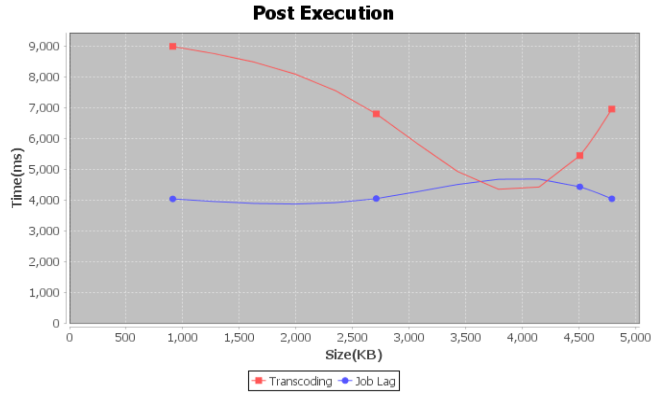

Figure 6: Performance between Job lag and Time

This graph we see that lag is constant, while comparing performance based on size, moving up from 3 MB we get a rising curve, which reaches peak around 4 Mb and then plummets rapidly.

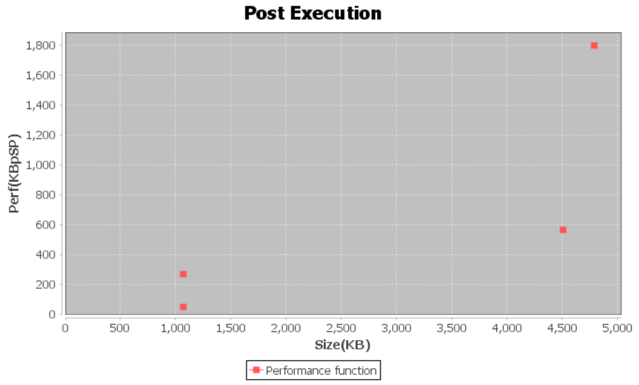

Figure 7: Analysis for Job lag and Time

Hence for normal 5 Mb files, we find the timing improves significantly. So our transcoding project performs better in terms of timing based on some relationship of size and resolution of video input getting better after a block size. 7ms of lag was found constant.

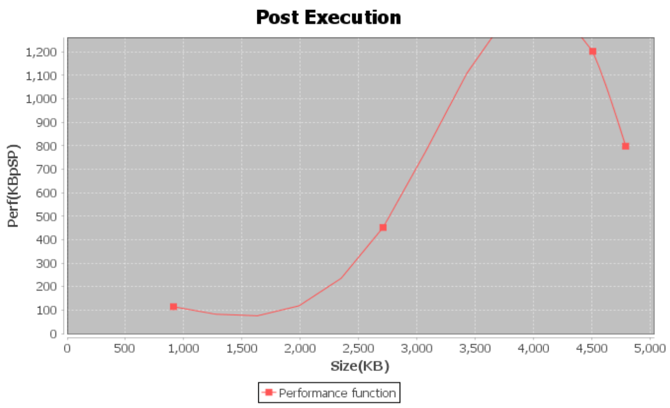

Figure 8: Performance function for video transcoding

This performance function is a simple evaluator based on following data inputs, such as file size, number of presets, transcoding time and lag.

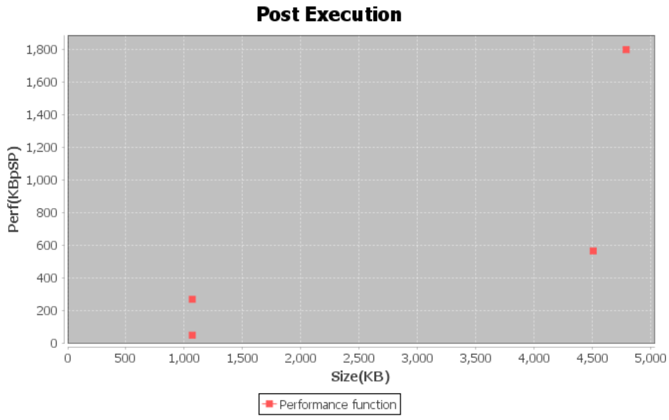

Figure 9*:* Preset Analysis for video transcoding

This is a proportionality based evaluation, performance being directly proportional to size, and inversely proportional to sum of transcoding time and lag multiplied by number of presets.

After the modeling of the simulation we have a tendency to expect to know the impact of varied parameters as followed:

- Impact of bit rate versions
- Unit transcoding prices
- Unit information measure prices
- Unit storage price
- User request pattern

5. Conclusion

In this paper, we studied the optimal transcoding and system for transcoding in a cloud computing platform is provided. The system comprises a receiver configured to receive a first uploading file which may be a multimedia file, one data block at a time wherein the data block may have a size of several kilobytes, a storage configured to store a plurality of received data blocks of the first uploading file, a split-while-uploading module configured to generate a first small segment file from the plurality of received data blocks while the receiver still receiving a new data block of the first uploading file, and a transcoder configured to transcode the first small segment file from one format such as a bit rate or a frame size to another while the receiver is receiving a new data block of the first uploading file. The system may comprise a second transcoder which transcodes a second small segment file generated from a second plurality of received data blocks while the receiver is still receiving a new data block of the first uploading file. Following experiments were performed relatively normal load with low arrival rate, relatively normal load with high arrival rate and highly variable load with low arrival rate.

The application developed for this project used free tier amazon web services account which has limited features such as number of pipelines and we used video only presets for evaluation. . Cross region streaming using CDN to deliver fast streaming is very important, Cloud front CDN from Amazon has the features to implement an application like 'Netflix'.

Acknowledgment

We would like to thank the referred authors and our department and management faculties for the support and help in our research endeavor

References

[1] Cisco,"Cisco visual networking index: Forecast and methodology, 2013-2018," White Paper, 2013.

[2] Accenture, "Video-over-internet consumer survey 2013," [online],Video-Over-Internet-Consumer-Survey-2013.pdf, 2013.

[3] G. Pallis and A. Vakali, "Insight and perspectives for content delivery networks," Communications of the ACM, Vol. 49, no. 1, pp. 101-106,2006.

[4] Y. Liu, F. Li, L. Guo, B. Shen, and S. Chen, "A servers perspective of Internet streaming delivery to mobile devices," in IEEE INFOCOM, 2012, pp. 1332–1340.

[5] W. Zhu, C. Luo, J. Wang, and S. Li, "Multimedia cloud computing," IEEE Signal Processing Magazine, vol. 28, no. 3, pp. 59–69, 2011.

[6] Y. Wen, X. Zhu, J. Rodrigues, and C. Chen, "Cloud mobile media: Reflections and outlook," IEEE Transactions on Multimedia, vol. 16, no. 4, pp. 885-902, 2014.

[7]Google,"Google+ hangout," [online], http://www.google.com/+/learnmore/hangouts/, 2013.

[8] Y. Wu, Z. Zhang, C. Wu, Z. Li, and F. C. Lau, "Cloudmov: Cloud-based mobile social TV," IEEE Transactions on Multimedia, vol. 15, no. 4, pp. 821-832, 2012.

[9] Y. Jin, Y. Wen, and H. Hu, "Minimizing monetary cost via cloud clone migration in multi-screen cloud social TV system," in IEEE GLOBECOM, 2013, pp. 1747-1752.

[10] D. D´ıaz-S´anchez, F. Almenarez, A. Mar´ın, D. Proserpio, and P. Arias Cabarcos, "Media cloud: an open cloud computing middleware for content management," IEEE Transactions on Consumer Electronics, vol. 57, no. 2, pp. 970-978, 2011. 11

[11] Y. Jin, Y. Wen, G. Shi, G. Wang, and A. Vasilakos, "Codaas: An experimental cloud-centric content delivery platform for user-generated contents," in IEEE International Conference on Computing, Networking and Communications, 2012, pp. 934-938.

[12] F. Wang, J. Liu, and M. Chen, "CALMS: Cloud-assisted live media streaming for globalized demands with time/region diversities," in IEEE INFOCOM, 2012, pp. 199-207.

[13] X. Wang, T. T. Kwon, Y. Choi, H. Wang, and J. Liu, "Cloud-assisted adaptive video streaming and social-aware video prefetching for mobile users," IEEE Wireless Communications, vol. 20, no. 3, pp. 72-79, 2013.

[14] Z. A. Qazi, C.C.Tu, L. Chiang, R. Miao, V. Sekar, and M. Yu, "SIMPLE-fying middlebox policy enforcement using SDN," in ACM SIGCOMM, 2013, pp. 27-38.

[15] R. Grandl, G. Ananthanarayanan, S. Kandula, S. Rao, and A. Akella, "Multi-resource packing for cluster schedulers," in ACM SIGCOMM, 2014, pp. 455-466.

[16] L. Zhou and H. Wang, "Toward blind scheduling in mobile media cloud:Fairness, simplicity, and asymptotic optimality," IEEE Transactions on Multimedia, vol. 15, no. 4, pp. 735-746, 2013.

[17] Y. Wu, C. Wu, B. Li, X. Qiu, and F. C. Lau, "Cloudmedia: When cloud on demand meets video on demand," in IEEE International Conference on Distributed Computing Systems, 2011, pp. 268-277.

[18] T. Stockhammer, "Dynamic adaptive streaming over http: standards and design principles," in ACM MMSys, 2011, pp. 133-144.

[19] J. He, D. Wu, Y. Zeng, X. Hei, and Y. Wen, "Toward optimal deployment of cloud-assisted video distribution services," IEEE Transactions on Circuits and Systems for Video Technology, vol. 23, no. 10, pp. 1717–1728, 2013.

[20] L. Xu, S. Kwong, H. Wang, Y. Zhang, D. Zhao, and W. Gao, "A universal rate control scheme for video transcoding," IEEE Transactions on Circuits and Systems for Video Technology, vol. 22, no. 4, pp. 489-501, 2012.

[21] T. Shanableh, E. Peixoto, and E. Izquierdo, "MPEG-2 to HEVC video transcoding with content-based modeling, IEEE Transactions on Circuits and Systems for Video Technology, vol. 23, no. 7, pp. 1191-1196, 2013.

[22] L. Guo, S. Chen, and X. Zhang, Design and evaluation of a scalable and reliable P2P assisted proxy for on-demand streaming media delivery, IEEE Transactions on Knowledge and Data Engineering, vol. 18, no. 5,pp. 669-682, 2006.

[23] W.-P. Yiu, X. Jin, and S.-H. Chan, "VMesh: Distributed segment storage for peer-to-peer interactive video streaming," IEEE Journal on Selected Areas in Communications, vol. 25, no. 9, pp. 1717-1731, 2007.

[24] S. Guo, H. Xie, and G. Shi, "Collaborative forwarding and caching in content centric networks," in IFIP Networking, 2012, pp. 41-55.

[25] Y. Li, H. Xie, Y. Wen, and Z.-L. Zhang, "Coordinating in-network caching in content-centric networks: Model and

analysis," in IEEE International Conference on Distributed Computing Systems, 2013, pp. 62-72.

[26] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and Zipf-like distributions: Evidence and implications," in IEEE INFOCOM, vol. 1, 1999, pp. 126-134.

[27] Y. Jin and Y. Wen, "PAINT: Partial in-network transcoding for adaptive streaming in information centric network," in IEEE/ACM IWQoS, 2014, pp. 208-217.

[28] K. Li, H. Shen, F. Y. Chin, and W. Zhang, "Multimedia object placement for transparent data replication," IEEE Transactions on Parallel and Distributed Systems, vol. 18, no. 2, pp. 212-224, 2007.

[29] Z. Wang, L. Sun, C. Wu, W. Zhu, and S. Yang, "Joint online transcoding and geo-distributed delivery for dynamic adaptive streaming," in IEEE INFOCOM, 2014, pp. 94-102.

[30] M. Bjorkqvist, L. Y. Chen, M. Vukolic, and X. Zhang, "Minimizing retrieval latency for content cloud," in IEEE INFOCOM, 2011, pp. 1080-1088.

[31] R. Grandl, K. Su, and C. Westphal, "On the interaction of adaptive video streaming with content-centric networking," arXiv preprint arXiv: 1307.0794, 2013.

[32] C. Peng, M. Kim, Z. Zhang, and H. Lei, "VDN: Virtual machine image distribution network for cloud data centers," in IEEE INFOCOM, 2012, pp. 181-189.

[33] S. Lederer, C. Mueller et al., "Distributed dash dataset," in ACM MMsys, 2013, pp. 131-135.

[34] S. K. Fayazbakhsh, Y. Lin et al., "Less pain, most of the gain: Incrementally deployable ICN," in ACM SIGCOMM, 2013, pp. 147-158.

[35] Y. Jin, Y. Wen, H. Hu, and M. Montpetit, "Reducing operational costsin cloud social TV: an opportunity for cloud cloning," IEEE

[36]http://docs.aws.amazon.com/elastictranscoder/latest/developerguide/working-with-pipelines.html.

[37https://www.cloudave.com/26295/using-elastic-transcoder-in-amazon-web-services.