Load Distribution Method and Admission control for Streaming Media QoS in Distributed Transcoding Servers *

Nansook Heo, Dongsun Lim, Dongmahn Seo, Inbum Jung, Yoon Kim
Department of Computer Science and Engineering
Kangwon National University
Hyoja, Chuncheon, Kangwon, 200-701, Korea
{nsheo, dslim, dmseo, ibjung, yooni}@snslab.kangwon.ac.kr

Abstract

The recent advance in wireless network technologies has enabled the streaming media service on the mobile devices such as PDAs and cellular phones. Since the wireless network has low bandwidth channels and mobile devices are actually composed of limited hardware specifications, the transcoding technology is needed to adapt streaming media to the given mobile devices. When large scale mobile clients demand the streaming service, load distribution Methods among transcoding servers highly impact on the total number of QoS streams. In this paper, the resource weighted load distribution Method is proposed for the fair load distribution and the more scalable performance in distributed transcoding servers. Our proposed Method is based on the weight of resources consumed for transcoding to classified client grades and the maximum number of QoS streams actually measured in transcoding servers. The proposed policy is implemented on distributed transcoding system. In experiments, we evaluate its fair load distribution and scalable performance according to the increase of transcoding servers.

1. Introduction

Based on recently the amazing growth of telecommunication, computer and image compression technologies, the streaming media service has been spotlighted in many multimedia applications. The large amount of network traffics and the high performance computing ability are inevitable to support the QoS streams [1, 2, 3]. However, since the wireless network has low bandwidth channels and many mobile devices compose of limited hardware specifications, the transcoding technology is needed to adapt the originally encoded MPEG media to the given mobile devices.

The transcoding system is usually composed of both the multimedia server with the originally encoded MPEG media and the transcoding servers to perform the adapting to the given environment. The multimedia server retrieves the MPEG media and sends them to the selected transcoding server. The transcoding server performs the transcoding to original MPEG video and also sustains the streaming service to the corresponding client. In particular, to provide QoS for clients, it is inevitable to guarantee streaming media without ceasing and jittering phenomena [3, 4, 5, 6, 7].

In this paper, the load distribution Methods for transcoding jobs are studied in distributed servers. The cluster server architecture has an advantage of the ratio of performance to cost and is easily extended from the general PCs [5]. This model usually consists of a front-end node and multiple backend nodes. In our research, the front-end node is used as a load distribution server and the backend nodes work as transcoding servers. Based on load distribution Methods, the load distribution server distributes the transcoding requests of clients into transcoding servers. To provide the QoS streams for various kinds of mobile clients, we propose the Resource Weight Load Distribution (RWLD) Method in the distributed transcoding servers. For the criteria of load distribution, we measure both the actual amounts of resources consumed and the maximum number of QoS streams by transcoding grades in each transcoding server. From the load weights by transcoding grades, the intrinsic property of streaming media can be reflected in the load distribution mechanism. And also, the two types of measured information are utilized as the threshold point of admission control to guarantee QoS for all clients. The proposed Method is implemented on distributed transcoding system together with other load distribution Methods. From our experiments, the RWLD Method shows the fair load distri-

*This research was financially supported by the Ministry of Commerce, Industry and Energy (MOCIE) and Korea Industrial Technology Foundation (KOTEF) through the Human Resource Training Project for Regional Innovation.

0-7695-2945-3/07 $25.00 © 2007 IEEE
DOI 10.1109/ICCSA.2007.21

bution in the heterogeneous transcoding servers and it leads to better performance scalability according to the increase of transcoding servers.

The rest of this paper is organized as follows. Sect. 2 describes related work for our research. In sect. 3, the RWLD Method is proposed to achieve the fair load distribution and more scalable performance in distributed transcoding servers. Sect. 4 explains our actual experimental environment. In sect. 5, the performance of the RWLD Method is evaluated and compared to other load distribution Methods. Sect. 6 concludes the paper.

2. Related Work

2.1. MPEG Profile

Mobile devices have their own the computing power, memory, network capacity. To adapt their working environment, the streaming media should be transformed from the original contents. There are MPEG media specifications to support the streaming media to mobile devices [8, 9]. Table 1 shows the MPEG profile composed of video size, frame rate, bit rate based on the operating environment of the streaming media service. As shown the Table 1, the MPEG media can be classified by 4 grades and each grade designates its own working mobile device.

Table 1. Specification of MPEG Profile

Grade	Video size	Frame rate	Bit rate (kbps)	Mobile device device
SQCIF	128 X 96	15	50	Cellular phone
QCIF	176 X 144	15	70	PDA
CIF	352 X 288	26	100	Laptop PC
4CIF	704 X 576	30	200	Desktop PC

2.2. Load Distribution Methods

Many researches were undertaken for the load distribution Methods in distributed servers. In particular, the distributed server architecture has been utilized in the Web server, game server and file server areas. As representative Methods in these areas, there are RR(Round Robin), LC(Least Connection), WRR(Weighted Round Robin), DWRR(Dynamic Weighted Round Robin) and so on.

The RR Method allocates servers according to the sequence of job arrival. Since the RR does not consider the state of servers and the intrinsic features of jobs, it is difficult to attain the effective load distribution among servers. The LC Method uses the count of clients connected to each server. This Method chooses the server with the least count value. The WRR Method designates the different weight to each server based on the capability of servers. This approach can not reflect the state of servers dynamically changed. To address the problem, the DWRR Method is suggested. For jobs distributing to servers, this Method considers the current state of backend servers.

3. Resource Weight Load Distribution Method

To provide the QoS streams for various kinds of mobile clients, we propose the Resource Weight Load Distribution (RWLD) Method. For the RWLD Method, the actual amounts of resources consumed for transcoding should be measured on the individual transcoding servers by the grades of mobile device. After that, the maximum numbers of QoS streams by transcoding grades are measured on each transcoding server. Based on the two types of measured information, the RWLD Method manages the fair load distribution among heterogeneous distributed servers as well as provides the scalable performance according to the increase of transcoding servers.

3.1. Resource Consumption by Transcoding Grades

To find the actual amount of resources consumed for each transcoding grade, we measure the usage of CPU, memory and network bandwidth exhausted by the classified grades described in the Table 1. A Desktop PC has a role for a transcoding server which is composed of 1.4 GHz CPU, 256 Mbytes Memory, and 100 Mbps Network Bandwidth. The Linux operating system is deployed and the FFMPEG program is used for the transcoding of MPEG media [4].

Table 2 shows the experimental results for transcoding 10 4CIF grade movies into SQCIF, QCIF and CIF grade respectively. As experimental results, we find that the transcoding for the same grade results in the almost same resource consumption rates regardless of which movies are selected. As shown in this Table, the CPU consumption rate is the highest among all resources. Based on the constant resource consumption rates, the resource weight for the corresponding transcoding grade can be computed in each transcoding server.

3.2. Resource Weight Table

Under the RWLD Method, the load distribution server uses the Resource Weight Table (RWT) for the fair load distribution and the admission control for guaranteed the QoS. The RWT is composed of 4 items such as the resource weight, maximum streams, total resource weight and accumulated weight. The first item means the relative resource consumption weights by transcoding grades. It is driven by

Table 2. Resource Consumption Rates by Transcoding Grades

Grage	CPU (%)	Memory (Mbytes)	Network (Kbps)
SQCIF	8.3	5.7	50
QCIF	8.5	5.8	70
CIF	16.3	6.4	100

the fastest exhausted resource when each transcoding server transcodes the original MPEG media into the corresponding grades. Table 3 shows the pseudo codes for computing the relative weight of transcoding grades in each transcoding server. M is the available memory in a transcoding server. B is the available network bandwidth. C is the available CPU capacity. Using the index i for the transcoding grade, Qci, Qri and Qmi are denoted as the CPU usage, network usage and memory usage for the corresponding transcoding grade i. For example, if we have 4 grades such as SQCIF, QCIF, CIF, 4CIF, the notations of CPU usages are Qc1, Qc2, Qc3 and Qc4 respectively. And also, the Wn means the relative resource consumption weight for transcoding grade n.

Table 3. Pseudo Codes for Resource Weight Computation

if (M/$Q_{mi} \geq$ B/$Q_{ri} \geq$ C/C_{mi}) {
// CPU is exhausted firstly
$W_n = \frac{Q_{C_n} \times 100}{\sum_{k=1}^{i} Q_{C_k}}$ (n=1,2,...,i) — equation (1)
}
else if (B/$Q_{ri} \geq$ C/$Q_{ri} \geq$ M/Q_{mi}) {
// Memory is exhausted firstly
$W_n = \frac{Q_{m_n} \times 100}{\sum_{k=1}^{i} Q_{m_k}}$ (n=1,2,...,i) — equation (2)
}
else if (C/$Q_{ci} \geq$ M/$Q_{mi} \geq$ B/C_{ri}) {
// Network is exhausted firstly
$W_n = \frac{Q_{r_n} \times 100}{\sum_{k=1}^{i} Q_{r_k}}$ (n=1,2,...,i) — equation (3)
}

Since the firstly exhausted resource restricts the total number of transcoding requests, the RWLD Method uses its property to compute the resource weight W_n. As shown in the following pseudo codes, the resource weight W_n for each transcoding grade is determined by the firstly exhausted resources. The C/Q_{ci}, M/Q_{mi}, B/Q_{ri} designate the number of transcoding requests under available the CPU capacity, the memory space and the network bandwidth respectively. Among them, the smallest number determines the relative resource weight W_n of all transcoding grades in the corresponding server. If the CPU is the firstly exhausted resource in a transcoding server, the equation (1) of the Table 3 is chosen to compute the relative resource weights. After that, the results are recorded into the first item to the corresponding server in the resource weight table, as shown as Table 4.

Table 4. Snapshot of Resource Weight Table on Initial Stage

	Transcoding Server A			
	resource weight	maximum streams	total resource weight	accumulated weight
SQCIF	25	8	200	0
QCIF	35	7	245	0
CIF	40	6	240	0

The maximum streams means the maximum number of QoS streams by transcoding grades in each transcoding server. This value is also achieved throughout the actual measurement. The total resource weight is computed by multiplying the resource weight item and the maximum stream. This value represents the total resource weight guaranteed the QoS by transcoding grades in each transcoding server. The accumulated weight means the resource weight accumulated in the corresponding transcoding server by currently executing transcoding jobs. In initial stage, the accumulated weight is zero.

3.3. Load Distribution and Admission Control

In the distributed server architecture, each server has the same hardware specifications or not. Using the heterogeneous transcoding servers, each server shows up different resource consumption rates during transcoding operations. In the RWT, the resource weight and accumulated weight items are exploited for the load distribution among heterogeneous transcoding servers. By looking at the performance of individual servers on the classified transcoding grades, the RWLD Method can apply the fair load distribution to heterogeneous distributed transcoding servers.

To guarantee QoS to all serviced streams, the admission control is inevitable in the streaming media service. If a new transcoding request ruins the QoS for currently serviced all streams, the admission control should reject the new client request to protect the existing clients. In our RWLD Method, the load distribution server performs the load distribution role as well as the admission control mission.

Fig. 1 is the flow chart of the load distribution and the admission control in the RWLD Method. As shown in this

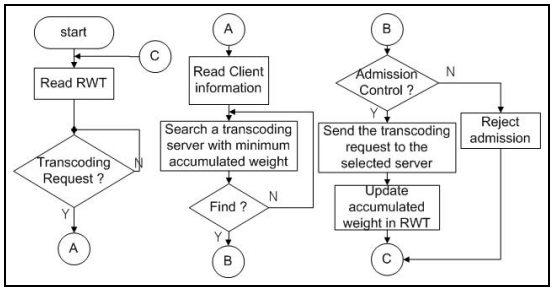

Figure 1. Flow Chart of RWLD Method

figure, the load distribution server initializes the RWT information and waits for client requests. To every transcoding requests, the RWLD Method searches a transcoding server with the minimum accumulated weight so that the fair load distribution can be maintained. In addition, to guarantee the QoS for currently serviced streams, the RWLD Method performs the admission control to the new transcoding request. If the admission is accepted, the new client request is sent to the selected transcoding server and its accumulated weight is updated. However, if the accumulated weight including the new request is over the total resource weight of the selected transcoding server, it is regarded as not eligible state for guaranteeing the QoS. In this case, since the new client request can destroy the QoS for currently serviced all clients, the admission control rejects the new client request.

4. Experimental Environment

In our experiment, the transcoding servers are composed of the 3 kinds of cluster systems. Total number of transcoding servers is 23 nodes. The cluster 1, 2 systems have 8 nodes respectively and the cluster 3 has 7 nodes. All nodes within a cluster system have the same hardware specification but the cluster systems have different hardware specifications.

We use the yardstick program to measure the performance of our distributed transcoding servers [10]. The yardstick program consists of the *virtual load generator* and the *virtual client daemon*.

The virtual load generator is located in the load distributed server. It generates client's transcoding requests based on the 3 parameters such as the distribution of transcoding grades, client's preferences to movies and client's arrival rate. Among the mobile devices, since the cellular phone takes a larger portion, we apply the Zipf distribution with the skew factor 0.271 to the transcoding from 4CIF grade to SQCIF grade [11]. The movies used in the Sect. 3.1 are used in our experiments. We regard that the popularity of each movie also follows a Zipf distribution with the skew factor 0.271. To the client's arrival rate, we use the Poisson distribution with λ=0.25 [10, 12]. The virtual client daemon locates in test-bed PCs for clients. Based on the MPEG profile specification of Table 1, the virtual client daemon measures the time elapsed for receiving the stipulated frame rate and bit rates of the requested transcoded movies. If the elapsed time is below 1 second, the virtual client daemon remains in an idle state until 1 second period passes.

5. Performance Evaluation

From the implemented distributed transcoding system, the performance of the RR, DWRR and RWLD Methods are measured. As performance metrics, we designate 2 metrics. The first is the amount of CPU consumed according to the increase of clients because the CPU is the fastest exhausted resource in our previous experiment. As a second metric, the total number of QoS streams is selected to evaluate the scalable performance of tested Methods.

5.1. CPU Consumption Rates

Fig. 2 shows the amount of CPU usage of transcoding servers under RR, DWRR, RWLD Methods. We used 23 transcoding servers involved in 3 kinds of cluster system. On account of space in this figure, we chose 2 transcoding servers from each cluster system. The A node and B node is from the cluster system 1. The C node and D node belongs to the cluster system 2. The E node and F node is from the cluster system 3.

As shown in the Fig. 2, the RR Method results in the different amounts of CPU usage among transcoding servers. The reason is that transcoding jobs are distributed based on just the arrival order. In particular, since the RR Method does not distinguish transcoding grades, it allows the overloaded transcoding servers and the underloaded servers to exist together. In the point of 120 clients, the CPU of the server A, C, E becomes saturate as 100% utilization rates, whereas the other servers do not.

In the DWRR Method, transcoding servers send their current resource usages to the load distribution server by periodically. Based on this information, this Method maintains the load distribution among transcoding servers. If the CPU usage of some transcoding servers reaches 100% utilization, this Method does not require additional transcoding jobs to these servers. Since the workload congestion to some specific transcoding servers is avoided, the DWRR Method does not destroy the QoS of all serviced streams. However, the load distribution server has overheads to communicate with transcoding servers. In addition, since the DWRR Method uses just the CPU utilization rate as an admission control, it does not reflect the intrinsic characteristic of streaming media in real time require-

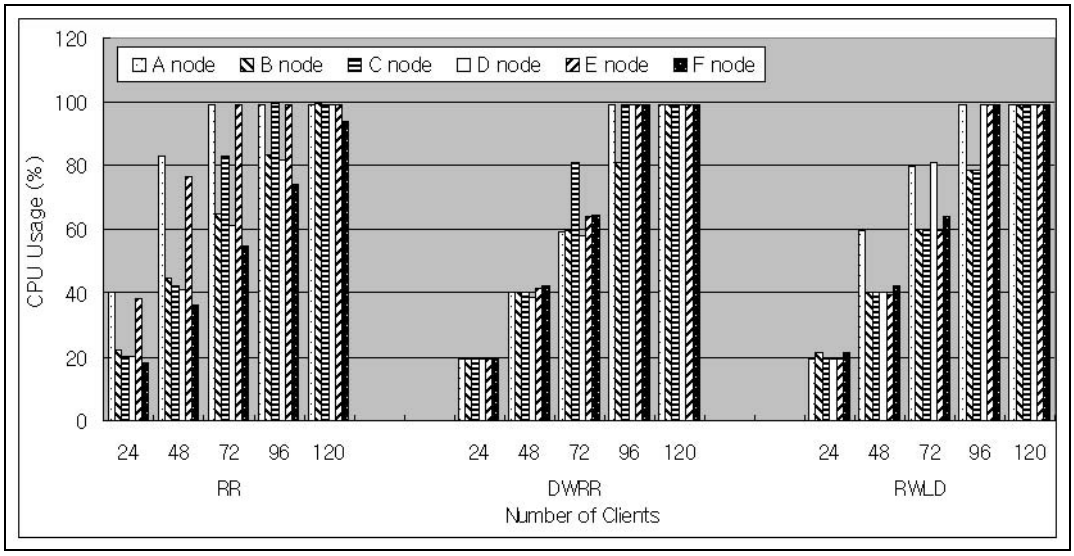

Figure 2. RR (Round Robin) Method

ment. Thus, even if the CPU utilization reaches 100%, the additional transcoding requests could be serviced to clients within the limited range. However, as shown in the figure 2, the DWRR Method shows fair load distribution among transcoding servers and does not ruin the QoS of all streams being serviced.

As shown in this Figure, the RWLD Method maintains the fair load distribution among transcoding servers like the DWRR Method. Since the DWRR uses the resource weights and the maximum streams according to the transcoding grades as the criteria of the load distribution and the admission control, there are no communication overheads between transcoding servers and the load distribution server. In addition, even if the CPU utilization reaches 100%, the additional transcoding jobs could be accepted in the range of proposed admission control mechanism. By considering the intrinsic property of streaming media, the RWLD Method contributes the fair load distribution as well as the scalable performance in distributed transcoding servers.

5.2. Performance Scalability

Fig. 3 shows the total number of QoS streams supported by RR, DWRR, RWLD Methods accordingly as the number of transcoding servers is increased. The QoS is the most important mandatory requirement in the streaming media service. If the serviced streams are insufficient to guarantee the QoS requirement by transcoding grades, those streams can not involve in the total number of QoS streams. For our experiments, the load generator invokes 294 transcoding jobs. Under the Zipf distribution with 0.271 skew factor, the SQCIF grade is 44, the QCIF is 86 and the CIF is 64.

As illustrated in Fig. 3, the maximum number of clients increases proportional to the number of transcoding servers in all Methods. In the RR Method, the overloaded servers with the congestion of transcoding jobs can not satisfy the QoS requirement. In particular, new transcoding requests allocated to the saturated servers has a negative impact on other QoS streams being serviced. From this reason, the RR Method shows the relatively low performance improvement across the increase of transcoding servers.

The DWRR Method does not consider the minimum amount of CPU consumed for transcoding to the desired transcoding grade. Even if the CPU utilization reaches 100%, it is possible to perform additional transcoding and streaming jobs within the range of satisfying the QoS requirement. The DWRR Method does not consider this characteristic of streaming media. In addition, to monitor the CPU usages of transcoding servers, it has the communication overhead between transcoding servers and the load distribution server periodically. This overhead results in the further increase of the CPU usage in transcoding servers. As a result, the overhead itself and the failure to notice for the intrinsic property of streaming media have a negative impact on the performance scalability.

On the other hand, the RWLD Method uses both the resource weight consumed and the maximum number of streams by transcoding grades as the criterion of the load distribution and the admission control. Based on these two types of pre-measured information, this Method not only fully reflects the intrinsic property of streaming media but also has no communication overheads to monitor the state information of the resources in transcoding servers.

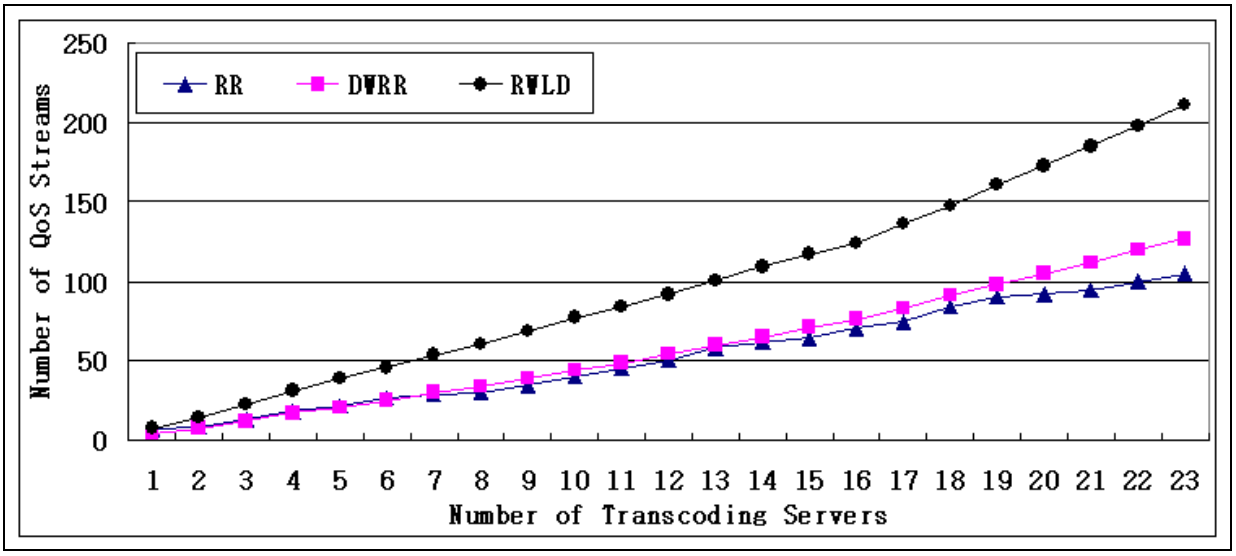

Figure 3. Performance Scalability

Based on these advantages, even if the CPU utilization reaches 100%, the RWLD Method can require the additional transcoding jobs within the range of satisfying the QoS requirement corresponding to each transcoding grade. As a result, the RWLD Method has been the best scalable performance among the experimented load distribution Methods.

6. Conclusion

In this paper, the load distribution Methods are studied in the distributed transcoding servers. The load distribution Method should provide the fair load distribution and scalable performance. We proposed the RWLD Method used the actual amount of resources consumed by transcoding grades and the maximum number of QoS streams in transcoding servers.

In our heterogeneous distributed transcoding servers, we had evaluated the fair load distribution and the scalable performance of the RR, DWRR and RWLD Methods. The RWLD Method maintained the fair load distribution among transcoding servers. This Method used the resource weights and the maximum streams as the criteria of the load distribution and the admission control. By the two types of pre-measured information, this Method not only reflects the intrinsic property of streaming media but also has no communication overheads to monitor the working state of transcoding servers. From our experiments, since the RWLD Method performed the admission control based on the QoS requirements of the classified transcoding grades, it showed more linear performance scalability than other Methods.

References

[1] Dinkar Sitaram, Asit Dan: Multimedia Servers: Applications, Environments, and Design. Morgan Kaufmann Publishers, 2000

[2] W.C. Feng, M. Lie: Critical Bandwidth Allocation Techniques for Stored Video Delivery Across Best-Effort Networks. The 20th International Conference on Distributed Computing Systems, pp.201–207, 2000

[3] D.H.C. Du, Y. J. Lee: Scalable Server and Storage Architectures for Video Streaming. IEEE Inter-national Conference on Multimedia Computing and Systems, pp.191–206, 1999

[4] Florin Lahan, Irek Defee, Marius Vlad, Aurelian Pop, Prakash Sastry: Integrated system for multimedia delivery over broadband ip networks. IEEE Transactions on Consumer Electronics, Vol. 48, No.3, pp.564–565, 2002

[5] C. Li, G. Peng, K. Gopalan, and T. Chiueh: Performance guarantees for cluster-based internet services, Proceedings of the 23rd International Conference on Distributed Computing Systems, pp378–385, May 2003

[6] Sumit Roy, Michele Covell, John Ankcorn, and Susie Wee: A System Architecture for Managing Mobile Streaming Media Services, 23rd International Conference on Distributed Computing Systems Workshops (ICDCSW'03), pp.408–419, 2003

[7] J. Guo, F. Chen, L. Bhuyan, and R. Kumar: A cluster-based active router architecture supporting video/audio stream transcoding services, Proceedings of the 17th International Parallel and Distributed Processing Symposium, pp.446–453, April 2003

[8] http://www.mpeg.org

[9] C. K. Hess, D. Raila, R.H. Cambell, and D. Mickunas: Design and performance of mpeg video streaming to palmtop computers, Proceedings of SPIE/ACM Multimedia Computing and Networking (MMCN2000), January 2000.

[10] Brian K. Schmidt, Monica S. Lam, J. Duane Northcutt: The interactive performance of SLIM: a stateless, thin-client architecture. ACM SOSP'99, pp.31–47, 1999

[11] C.C.Aggarwal, J.L.Wolf, and P.S.Yu: On optimal batching policies for viedo-on-demand storage servers, Proc. of IEEE ICMCS'96, pp.253–258, Hiroshima, Japan, June 1996

[12] Surendar Chandra, Carla Schlatter Ellis and Amin Vahdat: Differentiated Multimedia Web Services Using Quality Aware Transcoding, Proceedings of IEEE INFOCOM Conference, March 2000