International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:3, No:12, 2009

Data Placement in Heterogeneous Storage of Short Videos

W. Jaipahkdee[1] and C. Srinilta[2]

***Abstract*—**The overall service performance of I/O intensive system depends mainly on workload on its storage system. In heterogeneous storage environment where storage elements from different vendors with different capacity and performance are put together, workload should be distributed according to storage capability. This paper addresses data placement issue in short video sharing website. Workload contributed by a video is estimated by the number of views and life time span of existing videos in same category. Experiment was conducted on 42,000 video titles in six weeks. Result showed that the proposed algorithm distributed workload and maintained balance better than round robin and random algorithms.

***Keywords*—**data placement, heterogeneous storage system, YouTube, short videos

I. Introduction

STORAGE issue has become very important in a distributed system as storage has direct effect on overall system performance, reliability and availability. A number of storage related solutions addressing various aspects of storage issue have been proposed. Such solutions include RAID, Network Attached Storage (NAS), Storage Area Network (SAN), Object-Based Storage (OBS) [7]. SCADDAR (SCAling Disks for Data Arranged Randomly) uses REMAP functions to determine location of media blocks [10]. CRUSH (Controlled, Scalable, Decentralized Placement of Replicated Data) provides data distribution function for distributed object-base storage systems [11].

Nowadays, it is normal that storage system in an organization being composed of storage devices from many vendors. These devices differ in terms of capacity and capability. Workload distribution in heterogeneous storage environment has become a challenge. Each storage device should be given a workload according to its capability. Data placement is fundamental to workload distribution especially in an I/O intensive system; e.g., video playback service system. Video sharing websites such as YouTube [1] have been gaining acceptance world wide. Many short made-by-consumer videos are uploaded and viewed each day. Because the nature of short videos differs from that of commercial two-hour videos, storage and retrieval of short videos must be handled differently.

This paper focuses on data placement issue in short video sharing system with heterogeneous storage environment. A data placement technique that is aware of storage diversity when placing a new video is proposed. This technique also takes into account video characteristics such as video category and video life time span. Future workload is predicted from video access statistics.

This paper is organized as follows. Section II discusses background related to data placement and YouTube video sharing system. Section III explains the proposed data placement algorithm. Experiments and results are discussed in section IV. Finally, section V concludes the paper.

[1] W. Jaiphakdee is with the Computer Engineering Program, Faculty of Engineering, King Mongkut's Institute of Technology, Ladkrabang, Bangkok, Thailand (e-mail: minkbear@gmail.com).

[2] C. Srinilta is with the Department of Computer Engineering, Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand (phone: 66-2-739-2400 ext 127; e-mail: kschutim@kmitl.ac.th).

II. Background and Related work

A. Data placement

The main objective is to uniformly distribute the data in storage system. LH-based algorithms brought hashing mechanism into data distribution [2]. A hash function was used to generate a pseudo random number which was expected to result in a uniform distribution. However, data placement was determined by hash function only. Workload characteristics were not considered. Lee et. al. proposed an online assignment algorithm for real-time environment where file access rate was known in advance [3]. The goal was to minimize response time. Scheuermann et. al. presented a dynamic method that tracked the change of load [4]. Their method concentrated on balancing the heat (access rate) of disks using temperature (ratio between heat and block size) as the criterion.

Some studies had focused on data placement based on blocking probability. Tang et. al. designed a static genetic algorithm along with a heuristic bin-packing algorithm to perform offline placement in video on demand system [8], [9]. Feng et. al. proposed an adaptive object placement algorithm that minimized blocking probability [5]. They also proposed a method that tracked workload parameters.

B. YouTube

YouTube was founded in early 2005. YouTube is a video sharing website where user can upload and share short videos (duration less than 10 minutes). It is one of the fastest-growing websites today. As of October 25, 2009, YouTube ranks third in web traffic among all websites in the internet by Alexa’s traffic rank [12].

Cheng et. al. presented a systematic and in-depth measurement study on the statistics of YouTube videos [6]. They noticed that length, access pattern, growth trend and active life span of YouTube videos differed remarkably from those of traditional streaming videos.

III. Our Approach

A. *YouTube dataset*

Metadata of YouTube video is explained in Table I [13]. Metadata contains descriptive information; e.g., video title and category; as well as, statistical information; e.g., cumulative number of views and number of comments. YouTube also provides YouTube API as a mean to access video metadata.

TABLE I
DESCRIPTION OF VIDEO METADATA

Video ID	Every video from YouTube has a unique id which is 11 digit composing 0-9, a-z, A-Z, - and _.
Title	Title is the short text to describe a video.
Description	The description of a video is a field of text including information about the content of the video. This text can only be edited by the owner of the channel
Uploader	Uploader is the name of register user who upload video.
Category	In YouTube, user can select a category of 15 category provided by YouTube when uploading video. There are : Autos & Vehicles, Comedy, Education, Entertainment, Film & Animation, Gaming, Howto & Style, Music, News & Politics, Nonprofits & Activism, People & Blogs, Pets & Animals, Science & Technology, Sports and Travel & Events
Published date	The published date is the date when uploader upload and publish video.
Video length	Video length is a number of duration in second format.
Number of views	Number of views is a cumulative value of views.
Rating	Rating is the average number of stars which users have given the video (5 being the highest/best rating, 1 being the worst).
Number of comment	It's a number of comment which users to provide information related to a video.
Related video	A list of related videos might be related to the video by subject matter, so that you may find it easier to search out other videos based on the same or similar subject.

YouTube datasets used in our experiment were obtained from http://netsg.cs.sfu.ca/youtubedata/ [6].

Set A Data in set A was collected almost everyday from February 22 to May 18, 2007 (85 days). Total number of videos was 130,000 unique titles. The crawler started crawling from videos in "Recent Featured", "Most Viewed", "Top Rated" and "Most Discussed" lists, followed by their "Related videos" down to the depth of four.

Set B Data in this set was an updated statistics of number of views of 42,000 video titles collected weekly from March 5 to April 16, 2007 (6 weeks).

In 2007, YouTube videos were categorized into twelve categories. Video distribution of videos in Set A is depicted in Figure 1. The largest group, "Music", contributed as much as 22.90% where the smallest group, "Pets & Animals", contributed only 1.90%. The difference in number was more than 10 times.

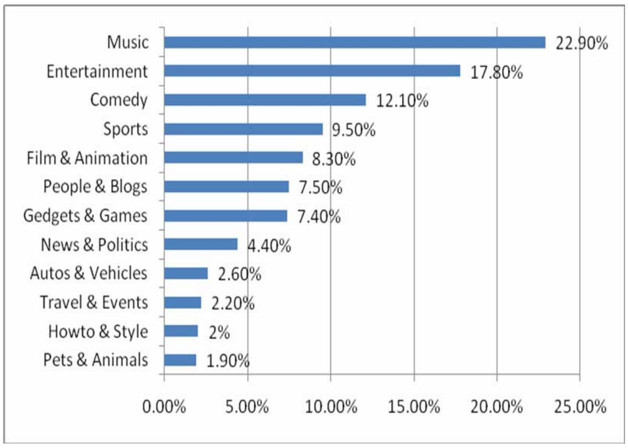

Fig. 1 Videos distribution by category

B. *Lifetime span*

It is normal that some videos are more popular than other. In addition, popularity of video changes over time. Usually, videos become less popular as they age.

"Lifetime" of a video on YouTube begins from the time when the video is uploaded. Since YouTube has no policy to remove video, lifetime of YouTube video is infinite.

Number of views of a video may increase quickly in the beginning and increase more slowly as time goes. Eventually, the video may not be viewed anymore after a certain point in time. There is a period when a video is considered "active". A change in number of views indicates video's "active" lifetime span; i.e, when weekly growth in number of views is less than a certain factor (life span factor), the "active" lifetime span is considered over [6].

TABLE II
AVERAGE ACTIVE LIFETIME SPAN BE CATEGORY

Category	Average active lifetime span (days)
Autos & Vehicles	96.38
Music	94.58
Film & Animation	91.84
Gadgets & Games	89.36
Howto & DIY	85.17
Travel & Places	83.15
Pets & Animals	78.34
Entertainment	76.61
Comedy	70.44
Sports	68.71
People & Blogs	56.79
News & Politics	54.22

Videos in Set A were processed with life span factor of 0.05. Their average active lifetime spans, by category, are shown in Table II. Videos can be divided into three groups according to their active lifetime spans; i.e., short life (less than 60 days), medium life (60-90 days) and long life (more than 90 days). Active lifetime span of "Music", the largest category, is the second longest among all twelve categories. “News & Politics” which is time sensitive category has the shortest active lifetime span.

C. Number of views

We noticed that changes in number of views of videos in same category varied in similar manner. Hence, we attempt to find an equation representing number of views of videos in each category. A polynomial degree six shown in equation (1) is used to fit changes in number of views. Coefficients and a constant term corresponding to each video category are described in Table III. We will use the equation to estimate number of views contributed by new video.

Videos in Set B were sampled and used to determine parameters in the equation. Actual numbers of views of short-life, medium-life and long-life categories are shown in Figures 2, 3 and 4, respectively. The fitted curve for each category is also included in the figure.

$$f(x) = ax^6 + bx^5 + cx^4 + dx^3 + ex^2 + fx + g \qquad (1)$$

TABLE III
COEFFICIENT AND CONSTANT TERM FOR EQUATION (1)

Category	a	b	c	d	e	f	g
Autos & Vehicles	0.1	-2.8	28	-137	370	-526	444
Comedy	0.8	-20.3	223	-1262	3898	-6223	4135
Entertainment	0.6	-16.5	183	-1055	3339	-5509	3831
Film & Animation	0.3	-7.5	75	-380	1036	-1435	913
Gadgets & Games	0.7	-19.6	208	-1142	3401	-5228	3346
Howto & DIY	0.2	-4.9	49	-255	739	-1117	803
Music	0.4	-11.1	119	-656	1973	-3073	2097
News & Politics	0.8	-20.9	225	-1249	3785	-5944	3875
People & Blogs	2.2	-59.8	653	-3687	11322	-17898	11447
Pets & Animals	0.4	-9.5	102	-562	1681	-2577	1613
Sports	1.5	-39.5	434	-2476	7703	-12408	8230
Travel & Places	0.3	-6.4	62	-308	811	-1075	603

D. Balancing workload

Since storage devices in heterogeneous storage environment differ not only in capacity but also capability, balancing workload is more complicated compare to that of the environment where all storage devices are similar. In heterogeneous environment, more workload should be given to device that can do more. When every device is working at its most comfortable level, the workload is then balanced.

Let a storage system consist of N different storage groups. Each group of storage is assigned a load factor (w). Load factor is a number ranging from 0 to 1. It represents the capability of a storage group with respect to capability of other groups. A storage group with best performance is given a load

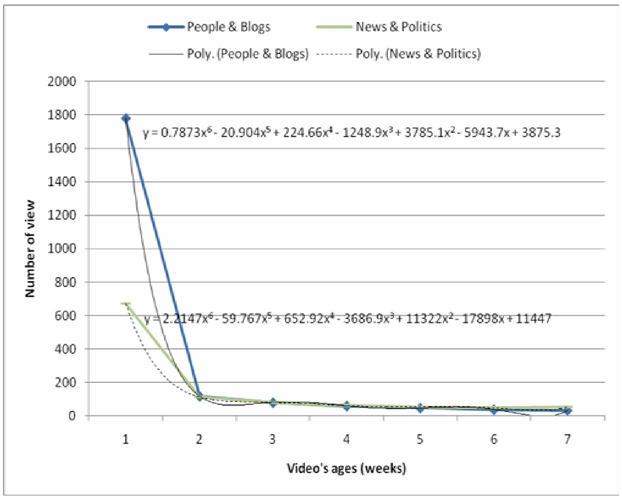

Fig. 2 Trend of Number of Views (short life videos)

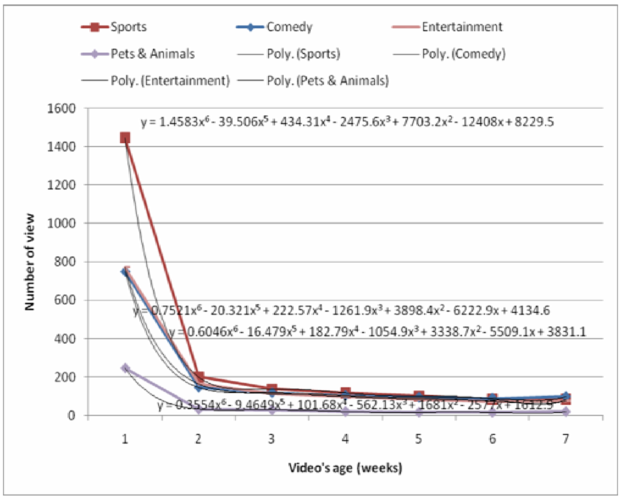

Fig. 3 Trend of Number of Views (medium life videos)

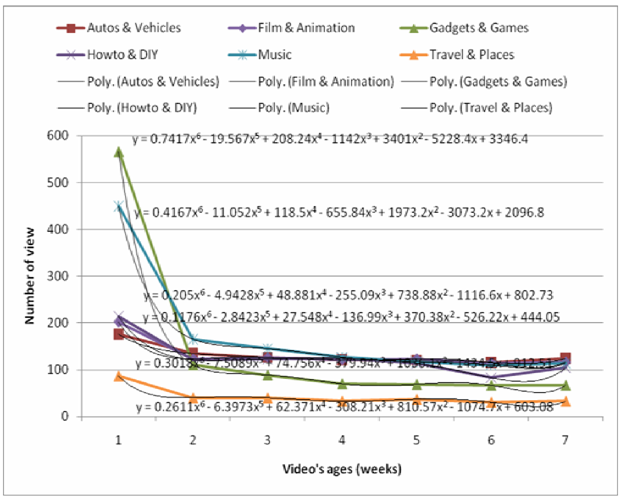

Fig. 4 Trend of Number of Views (long life videos)

International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:3, No:12, 2009

factor of value 1. The higher the load factor, the higher the performance.

In ideal situation where workload is uniformly distributed according to storage capability, storage group *k* carries a load of v_{i_k}. Ideal workload of storage group *k* (L_{i_k}) comes from equation (2). Workload of all storage groups are kept equal. Therefore, equation (3) is maintained.

$$L_{i_k} = w_k \times v_{i_k} \tag{2}$$

$$w_1 v_{i_1} = w_2 v_{i_2} = ... = w_N v_{i_N} \tag{3}$$

Total load in the system (V) is a summation of ideal load (v_i) assigned to each storage group, as shown in equation (4).

$$V = v_{i_1} + v_{i_2} + ... + v_{i_N} \tag{4}$$

However, actual situation may not be exactly the same as ideal situation. Let a storage group *k* carries actual load of v_{a_k}. The difference between actual load (v_{a_k}) and ideal load (v_{i_k}) indicates how well the load is distributed. The d_k in equation (5) represents the difference between the actual load and ideal load of storage group *k*.

$$d_k = \left| v_{a_k} - v_{i_k} \right| \tag{5}$$

The closer v_{a_k} is to v_{i_k}, for all storage group in the system, the better the overall distribution.

E. Data placement algorithm

In video sharing environment, number of views can be considered as load to storage system. Number of views of a video at any given point in time represents workload of that video at that time.

When a new video is uploaded to the system, it must be stored in one of the storage groups. In doing so, our data placement algorithm attempts to estimate the future workload contributed from that new video by using equation (1). Parameters in Table III are picked according to category of the new video.

In order to make sure that workload is always balanced, we take into account the load (number of views) expected to occur throughout the entire active lifetime span of the new video. The maximum number of views ($v_{\max}$) in the active lifetime span is brought into attention because it is the heaviest load in the active lifetime span. If a storage group is comfortable with the load at this level, it should handle the rest of the load just fine.

The current total workload is added by $v_{\max}$ as new video is now part of the system. Then ideal load (v_i) of each storage group is determined by equations (3) and (4). After that, the difference between actual and ideal load of each storage group (d_k) is determined (equation (5)). Lastly, a storage group with the maximum d is chosen to store the new video.

IV. Experiment and Result

Objective: The objective of the experiment was to compare the proposed data placement algorithm against traditional data placement algorithms in terms of ability in balancing workload in heterogeneous storage of short videos.

Environment:

Storage: Three different groups of storage devices with unlimited capacity. The first group was the fastest, the third group was the slowest and the second group stood in between. Load factors of 1, 0.75 and 0.5 were given to the first, the second and the third groups, respectively.

Dataset: The entire data in Set B (Section III A) was used in the experiment. It consisted of 42,000 video titles published from February 15 to March 3, 2007. The load of those videos was collected weekly for six weeks from March 5 to April 16, 2007.

Data placement algorithms: Three data placement algorithms were used in the experiment. Result was collected and compared. Such algorithms were round-robin algorithm, random algorithm and our algorithm. In round-robin algorithm, videos were placed into storage groups in round-robin fashion starting from storage group number 1. In random algorithm, a storage group was randomly chosen to store new video. In our algorithm, storage group was chosen based on current and future workload. Storage capability was taken into account when workload was determined.

Metric: We used *degree of balance* (db) to indicate the distribution of workload in storage system. As illustrated by equation (6), a degree of balance was evaluated from the difference between actual load and ideal load (d) of every storage group in a system. The maximum value of the ratio of d and v_i was chosen as it represented the case where the load was off-balanced the most.

$$db = 1 - \max\left(\frac{d_1}{v_{i_1}}, \frac{d_2}{v_{i_2}}, ..., \frac{d_N}{v_{i_n}}\right) \tag{6}$$

Degree of balance equals to 1 means that workload was distributed uniformly across storage groups. The closer the degree of balance is to 1, the better the distribution.

Experiment Simulations were performed in such a way that videos were uploaded to the system one by one in the order of their actual published dates at YouTube. Upon arrival of a new video, depending on data placement algorithm in use, the video was stored in one of the storage groups. We ran three simulations, one for each data placement algorithm. Same dataset and measurement were used in all simulations.

Workload on each storage group was collected every seven day period starting from March 5 to April 16, 2007. In each week, degree of balance was calculated and plotted to compare.

Result: Figure 5 shows degree of balance from the three simulations.

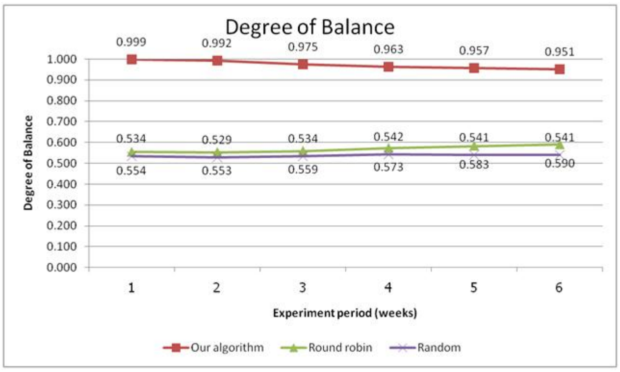

Fig. 5 Degree of Balance from three data placement algorithms

As seen in Figure 5, degree of balance from our algorithm was the closest to 1 throughout the six-week experimental period. This means that it distributed videos in the storage system better. However, the degree of balance dropped in later weeks. This was because a single load value (the maximum load value, $v_{\max}$) was used to represent the load during the entire active lifetime span while the actual load varied. The actual load was dropping toward the end of active lifetime span. Therefore, the degree of balance was lower at the end.

Round robin algorithm distributed videos slightly better than random algorithm. Degree of balance did not change much in experimental period.

V. Conclusion

In this paper, we proposed a data placement algorithm for heterogeneous storage system storing short videos. Our algorithm took into account the diversity of storage capability. We tried to distribute workload in such a way that each storage group handled workload appropriate to its capability. In addition, we analyzed viewing statistics in the past and tried to foresee future workload of a video. This information was used in the placement algorithm as well. The experiment was performed using real YouTube data in the period of six weeks. The result showed that the proposed algorithm gave better workload distribution when compared against round-robin and random algorithms.

References

[1] "YouTube", http://www.youtube.com/
[2] R.J. Honicky and E.L. Miller. Replication under scalable hashing: a family of algorithms for scalable decentralized data distribution. In *Proceedings of 18th International Parallel and Distributed Processing Symposium (IPDPS '04)*, Santa Fe, NM, Apr. 2004
[3] L.W. Lee, P. Scheuermann and R. Vingralek. File assignment in parallel I/O systems with minimal variance of service time. IEEE Trans. on Computers, Vol 49, No.2. (2000) 127-140
[4] P. Scheuermann, G. Weikum and P. Zabback. Data partitioning and load balancing in parallel disk systems. The VLDB Journal - The International Journal on Very Large Data Bases, Vol. 7, No. 1. (1998) 48-66
[5] D. Feng and L. Qin. Adaptive Object Placement in Object-Based Storage Systems with Minimal Blocking Probability. In *Proceedings of the 20th International Conference on Advanced Information Networking and Applications (AINA '06)*, Vienna, Austria, Apr. 2006
[6] X. Cheng, C. Dale and J. Liu. Statistics and social network of youtube videos. In *Proceedings of the 16th IEEE International Workshop on Quality of Service (IWQoS '08)*, Enschede, Netherlands, Jun. 2008
[7] M. Factor, K. Meth, D. Naor, O. Rodeh and J. Satran. Object Storage: The Future Building Block for Storage Systems. In *Proceedings of the Second International IEEE Symposium on Emergence of Globally Distributed Data*, Sardinia, Italy, Jun. 2005
[8] K. S. Tang, K. T. KO, S. Chan and E. Wong, "Optimal file placement in VOD system using genetic algorithm", IEEE Industrial Electronics, Vol 48, No.5. (2001) 891-897
[9] W.K.S. Tang, E.W.M. Wong, S. Chan and K.-T. Ko, "Optimal video placement scheme for batching VOD services", IEEE Trans. on Broadcasting, Vol 50, No.1. (2004) 16-25
[10] A. Goel, C. Shahabi, D. S. Yao and R. Zimmermann. SCADDAR: An Efficient Randomized Technique to Reorganize Continuous Media Blocks. In *Proceedings of The 18th IEEE International Conference on Data Engineering (ICDE '02)*, San Jose, CA, Feb. 2002
[11] S. A. Weil, S. A. Brandt, E. L. Miller and C. Maltzahn. CRUSH: Controlled, Scalable, Decentralized Placement of Replicated Data. In *Proceedings of the 2006 ACM/IEEE Conference on Supercomputing (SC '06)*, Tampa, FL, Nov. 2006.
[12] "Alexa Top 500 Global Sites", http://www.alexa.com/topsites.
[13] "YouTube Metadata", http://code.google.com/apis/youtube/2.0/reference.html.

Wiboonsak Jaiphakdee was born in Bangkok (Thailand), November 27th 1982, earned B.S. in Applied Mathematics from Faculty of Science, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand in 2005. He is currently a graduate student of Computer Engineering, faculty of Engineering, from King Mongkut's Institute of Technology, Ladkrabang, Bangkok, Thailand. His current research project is related to data placement of short videos in heterogeneous storage system.

Chutimet Srinilta earned a B.Eng. degree in Electrical Engineering from Kasetsart University, Bangkok, Thailand. She holds M.S. and Ph.D. degrees in Computer Engineering from Syracuse University, USA. At present, she is a faculty member of Computer Engineering department, King Mongkut's Institute of Technology Ladkrabang. Her current research interest is relevant to data storage and retrieval in social networks.