ELSEVIER

Journal of Systems Architecture 46 (2000) 163–179

JOURNAL OF SYSTEMS ARCHITECTURE

www.elsevier.com/locate/sysarc

# Performance analysis of video storage server under initial delay bounds

Shiao-Li Tsao *, Yueh-Min Huang, Jen-Wen Ding

*Department of Engineering Science, National Cheng Kung University, College of Engineering, Tainan 701, Taiwan*

**Abstract**

Previous studies on video storage servers focused on improving the disk throughput and reducing the server buffer size. However, the initial delay of a new request, one of the most important quality of service (QoS) parameters from the users' point of view, is almost neglected while designing a storage subsystem or evaluating its performance. For different types of video-on-demand (VOD) services such as interactive video game, digital library, or movie-on-demand system, the initial delay can vary from 0.5 s to 5 min. This criterion brings some impacts on designing a storage server for a particular VOD application. In this paper, we investigate the storage server design and the performance evaluation of VOD systems with different initial delay guarantees. We propose a new performance model on evaluating the efficiency of a video storage server so that a cost-effective configuration can be easily obtained under a specified initial delay bound. 

## 1. Introduction

Recently, the video-on-demand (VOD) system has been widely applied to entertainment and education. The most important design issues of a video storage server are to provide jitter-free video services as well as to promote the utilization of the storage bandwidth to accommodate more users. Disk array system which stripes video files on several hard disks is a common approach to increase the space and bandwidth capacities and to support a large number of users accessing a huge amount of video content for a video storage server [12,14,21]. Two basic data striping techniques, fine-grain and coarse-grain, have been employed on a disk array. For fine-grain striping, a video file is divided into access blocks, and each block is further striped onto a number of hard disks. Here, an access block is defined as the total amount of data retrieved once by a storage server for a particular read request. Therefore, the hard disks can

* Corresponding author. E-mail: sltsao@iis.sinica.edu.tw

1383-7621/00/$ - see front matter 
PII: S1383-7621(98)00065-4

serve a single read request parallel to their bandwidths. On the other hand, for coarse-grain striping, each access block is completely stored on a hard disk in order to increase the number of concurrent users served by the storage server at a time. In general, coarse-grain striping has a higher concurrency but a poorer parallelism than fine-grain striping. Oezden et al. studied on data striping techniques for VOD applications and concluded that coarse-grain striping is more suitable for the VOD system than fine-grain striping due to its low buffer requirement and high disk throughput [13]. To improve the disk throughput, disk scheduling policies and data placement schemes are widely exploited. Round-robin disk scheduling services periodical requests by a fixed order without taking the physical location of these requested blocks into account. This policy obtains fast response time, low buffer requirement, but results in poor disk bandwidth utilization owing to non-optimized disk seeking [7,11,18]. Different from round-robin scheduling, SCAN disk scheduling optimizes the block retrieval sequence by their physical locations. Thus, it reduces seek time overhead and promotes the disk throughput to a great extent. However, more delay will be introduced for a request, and since the retrieval order of the requested blocks is variant by the time, each user requires one additional buffer to compensate the uncertainty of data arrival time into the memory buffer [7]. Yu et al. proposed grouped sweeping scheduling (GSS) to get the compromise between the above two policies [1]. They partition access requests into several groups, apply round-robin between the groups and employ SCAN within a group. GSS has a higher disk throughput than a round-robin, and a lower buffer requirement and delay than SCAN. Occasionally, data placement schemes are employed to promote disk throughput, which can be categorized into three approaches: random, contiguous, and constrained data placement schemes. Contiguous data placement scheme sequentially stores a video file on a hard disk in order to eliminate extra seeks while retrieving an access block, which obtains a much higher disk throughput than the random data placement scheme. Rangan and Vin studied on the fundamental disk layout problem for a multimedia application, and presented a constrained data placement method which places consecutive blocks of the same media file within a fixed distance to avoid jitters on playback [2]. Some variants of constrained data placement called region-based or cluster-based schemes were explored in Refs. [3–5]. They partition a disk into a number of regions where a region is defined as a group of contiguous tracks on a disk, and store the blocks of a video file on these regions by a pre-determined sequence. Then, the disk head is confined to retrieve data blocks within a region during a period of time. After the period of time, the disk head moves to the nearby region and reads the requested blocks for the next period. By restraining the accesses within a region of a hard disk, the disk throughput can be improved by reducing the seeking distance. However, the users suffer from long initial delays. Change and Hector Garcia-Molina proposed several solutions on reducing initial delay for region-based data placement, but they require extra storage space to replicate the first several blocks of each video file [16]. To evaluate the performance of a video storage server, some work emphasized the trade-off between data striping strategies, buffer requirement, and disk throughput [6]. Some studies presented performance models based on the number of supported users per hard disk [6] or cost-effective models based on cost per user [9,10]. Unfortunately, the initial delay of new requests is never taken into consideration among them.

Not only the jitter-free during playback but also the initial delay, one of the quality of service (QoS) parameter, are most concerned by users for the demand service. The acceptable initial delay varies from less than 0.5 s to more than 5 min for