Harmonic Placement: File System Support for Scalable Streaming of Layer Encoded Object

Sooyong Kang
Department of Computer Science Education
Hanyang Univ., Seoul, 133-791, Korea
sykang@hanyang.ac.kr

Youjip Won* Seunghyun Roh
Dept. of Electrical & Computer Engineering
Hanyang Univ., Seoul, 133-791, Korea
{yjwon|gerran}@ece.hanyang.ac.kr

ABSTRACT

From file system's point of view, scalable streaming introduces another dimension of complexity in disk scheduling. This is particularly because when a subset of information is retrieved, the playback does not necessarily coincides with the sequential access. We propose a new file organization technique called *Harmonic placement*. The basic idea is to cluster the frequently accessed layers together. We develop elaborate analytical models for different file organization techniques: *Progressive placement*, *Interleaved placement* and *Harmonic placement*. We investigate the performance of file organization techniques under varying workload conditions. The models developed in this work enables us to predict the performance and efficiency of the storage system in scalable streaming environment. We find that the Harmonic placement outperforms other schemes in scalable streaming environment.

Key Words: File System, Multimedia, Layered Encoding, Scalable Streaming

1. INTRODUCTION

Compression technology and network transport technology have made significant advancement to efficiently provide multimedia service over dynamically changing user bandwidth availability. These technique bring greater flexibility in content management and save greater amount of storage space. In storage subsystem's aspect, numerous techniques have been proposed to efficiently support real-time multimedia I/O, e.g. disk scheduling, buffer cache management, dedicated file system for multimedia application, load balancing and etc.

To effectively handle heterogeneity in network bandwidth and hardware capability of client terminals, a number of compression schemes have been proposed[5, 7]. For a given bandwidth budget, it is important to select right set of layers in transmitting layer encoded object. A number of works proposed layer selection schemes under unicast and multicast setting[2, 1, 9, 10, 11]. Scalable streaming and layered encoding is an effective solution in video streaming for wireless environment[12, 3, 6]. While the above mentioned works well cover the issue of real-time multimedia streaming in network, compression, or file system area, the efficient support of layered streaming from file system's perspective is yet to be answered.

In this work, we aim at filling out the chasm between file system technology and scalable streaming technology of layer encoded content. We carefully conjecture that the placement of data block in scalable encoding needs to be treated differently from the legacy multimedia contents. When the file is encoded in scalable fashion, and playback rate dynamically changes, the notion of sequential playback does not necessarily coincide with sequential access on the file data blocks. Hence, scalable encoding scheme introduces another dimension of complexity from file system point of view. Legacy file system model and disk scheduling strategy does not incorporate this characteristics.

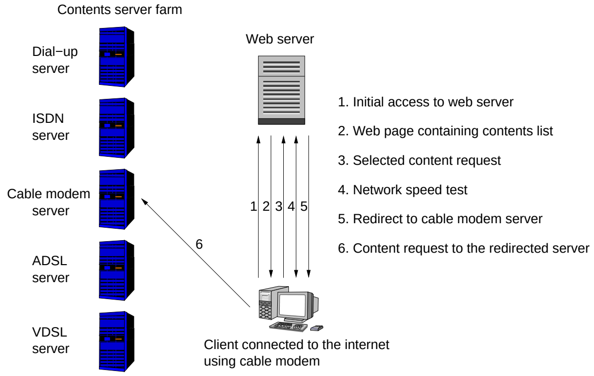

Figure 1: Service Model

Fig. 1 illustrates the underlying service environment. User can access the content via wide variety of different network medium whose bandwidth capability varies widely. To effectively cope with the variety in connection medium, content providing system consists of a number of servers and each server is dedicated to harbor the contents for a given bandwidth connection. Once the connection speed of the incoming request is determined, the incoming request is directed to the appropriate server and is serviced from the respective server.

In this work, we propose a file organization technique called *Harmonic placement*. The contribution of their work in two fold. First, we developed an elaborate model for file system

---

*Corresponding Author.

NOSSDAV '06 Newport, Rhode Island USA

technique. It is important that we have accurate model to predict the performance of a file system under a given workload. Second, we found that harmonic interleaving actually yields the most desirable performance against the existing file organization techniques. We examine the performance of a number of different file organization techniques. We compare the result obtained from our performance model and the result from the physical experiment. The result from these two sources well match each other and the harmonic interleaving exhibit superior performance under legacy streaming service environment.

2. FILE ORGANIZATION TECHNIQUES

We view a media file as a collection of logical storage units, called segment. A segment can be a frame or group of pictures. We assume that the amount of data read in a round (or period) is determined based on the Constant Time Length (CTL) scheme. We examine three different file organization technique: *Progressive placement*, *Interleaved placement*, and *Harmonic placement*.

Progressive placement strategy clusters the data blocks in a layer together. This allocation strategy manifests itself when network bandwidth availability is very limited and when the streaming server can transport the lowest layers in most of the time. Progressive placement scheme entails significant disk head movement overhead when the server transports larger numbers of layers. We call the frame-major placement scheme as *Inter-leaved placement*. This is plain sequential placement. When the streaming server retrieves data blocks in all layers, disk access yields sequential access pattern not only from a logical aspect but also from a physical aspect. When the server transports only the proper subset of layers, either file access entails undesirable seek operation or server needs to discard some of the retrieved information. Interleaved placement scheme manifests itself when the server transports most of the layers.

The *Progressive* placement and *Interleaved* placement strategies can be thought as two extremes in a wide spectrum of file organization techniques. In practice, the network bandwidth availability varies widely. Also, the speed of the client connection varies from a few hundreds Kbits/sec (e.g. W-CDMA) to hundreds of Mbits/sec. Data placement strategy should be carefully designed so that it can efficiently support a wide variety of QoS requirements. We believe that neither *Progressive Placement* nor *Interleaved Placement* strategy is desirable from the perspective of file system efficiency. In this study, we propose a novel file organization strategy, *Harmonic Placement*. Fig. 2 illustrates the file organization under the *Harmonic placement* strategy.

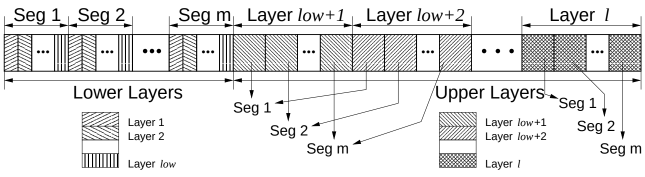

Figure 2: Harmonic Placement: In the figure, *low* means the number of lower layers in the object.

In Harmonic placement, the layers are partitioned into two groups: a set of lower layers and a set of upper layers. For example, with five layers, the layers can be partitioned as $\{L_1, L_2, L_3\}$ and $\{L_4, L_5\}$. In this case, we assume that upto layer 3 are frequently requested and layer 4 and layer 5 are less frequently accessed. Harmonic Placement adopts Progressive Placement for inter-group placement and Interleaved Placement for intra-group placement. Using this scheme, we can reduce disk seek time by clustering the frequently accessed layers together. Hence, when only lower layers are accessed in most of the time and information in the upper layers is rarely used, this scheme outperforms the other schemes. However, if the upper layers are accessed frequently, the disk seek overhead can result in lower performance. Therefore, layers needs to be carefully partitioned by considering both network bandwidth and the client device. The effectiveness of Harmonic placement is subject to the layer partitioning policy and the variability in network bandwidth availability.

Notation	Description
B	maximum disk bandwidth (MBytes/sec)
R	length of a round length (sec)
l	number of layers
$T(x)$	time to seek x cylinders
r_i	data rate of layer i
N	total number of sessions
L_o	size of the object in number of cylinders
L_i	size of layer i in number of cylinders
N_i	number of sessions accessing layer i in a round
L_o^i	size of object i in number of cylinders
L_j^i	size of layer j of object i in number of cylinders
N_o	total number of objects
N^i	number of sessions accessing object i
N_j^i	number of sessions accessing layer j of object i
w	index of the first object accessed in a round
z	index of the last object accessed in a round

Table 1: Notations

3. MODELING THE FILE SYSTEM

3.1 Background

Developing a file system performance model for a given workload is very challenging task. In generic multimedia service environment, it is common that a server harbors multiple video files and a number of clients access these contents in *On-Demand* fashion. We develop analytical performance model for individual placement scheme for this streaming environment.

We use *Disk Operation Efficiency* as a performance metric for file system. Disk Operation Efficiency is defined as $\frac{T_{read}}{T_{read}+T_{overhead}}$, where T_{read} and $T_{overhead}$ corresponds to data transfer time and the disk overhead which includes seek time, rotational latency, command queueing and etc. Table 1 shows notations used in this work.

There exist multiple video objects in the storage and multiple sessions are accessing them. *Session* in this paper refers to only disk retrieval session and does not include the playback from the buffered contents. At any given time instance, each session accesses different portion of an object or different object.

Before defining seek overhead model for each scheme, we first develop the common data read time model. Since the total amount of data that should be read in the round is $\sum_{i=1}^{N_o} \sum_{j=1}^{l} r_j \cdot R \cdot N_j^i$, we can model the time to transfer data blocks read from the disk as $T_{read} = \frac{\sum_{i=1}^{N_o} \sum_{j=1}^{l} r_j \cdot R \cdot N_j^i}{B}$.

To establish the seek overhead model, we categorize seeks into *intra-object seek*, *inter-object seek* and *return seek*. Intra-object seek occurs in an object while reading data blocks in different segments or layers, inter-object seek occurs after reading the last necessary data block in an object and going forth to the first necessary data block in the next object and the return seek is a seek from the last block in a round to the first block of the next round. In the rest of the paper, we call the intra-object seek as *intra seek* and the inter-object seek as *inter seek*.

Let us model the inter-seek distance first. Assuming that we are accessing object i and object j in sequential fashion, the inter-seek distance between object i and object j consists of three components. The first component, *distance A*, is the seek from the last read block of object i to the end of object i. The second component, *distance B*, is the seek distance to skip the objects between object i and j. The third component, *distance C*, is the seek distance from the beginning of object j to the first data block to be read in object j. For convenience, let us include *distance C* to the intra seek model and hence we can only consider *distance A* and *distance B* for inter seek overhead model. Let the *last-read cylinder* of an object be the cylinder where the last block read in the object locates. Then, the *distance A* of an object becomes the distance between the *last-read cylinder* to the last cylinder of the object. Assuming object k resides between object i and object j, the *distance B* can be thought of as the *distance A* of object k regarding the first cylinder of the object as the *last-read cylinder* of the object. Therefore, we can model the inter seek overhead in per object base with only *distance A*. Let Δ denote the distance between the first cylinder of the object and the *last-read cylinder* of the object. Then, the *distance A* of each object can be calculated as the size of the object subtracting Δ.

Assume that the popularity of each object is the same. Let w and z be the indices of the first and the last object accessed in a round, respectively. Then the expected values of w and z correspond to $\left\lceil \frac{N_o}{N+1} \right\rceil$ and $\left\lceil \frac{N_o}{N+1} N \right\rceil$, respectively. The distance of return seek can be approximated as the sum of the distance from the first cylinder of object w to the last cylinder of object $z-1$, $\sum_{i=w}^{z-1} L_o^i$, and Δ of object z (Fig. 3).

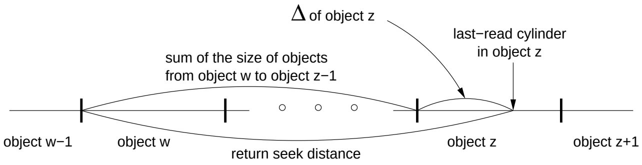

Figure 3: The return seek distance

3.2 Progressive Placement Scheme

Progressive placement scheme clusters the data blocks in the same layer together and the number of seeks for layer j of object i in a round corresponds to the number of sessions but cannot exceed the number of cylinders occupied by the respective layer. Therefore, the number of seeks can be represented as $\min\{L_j^i, N_j^i\}$. the total number of intra-layer seeks corresponds to $\sum_{j=1}^{l} \min\{L_j^i, N_j^i\}$. The average distance of individual seeks corresponds to $\lfloor \frac{L_j}{\min\{L_j, N_j\}+1} \rfloor$. After reading all requested data blocks in a layer, disk head moves to the appropriate cylinder in the next layer(inter-layer seek).

The overhead of intra seek to read data blocks in layer j of object i, $T_{intra,j}^i$, can be modeled as $T_{intra,j}^i = T\left(\lfloor \frac{L_j^i}{\min\{N_j^i+1, L_j^i} \}\rfloor\right) \cdot \min\{N_j^i, L_j^i\}$. The total intra seek overhead in object i becomes $\sum_{j=1}^{l} T_{intra,j}^i$.

To calculate Δ, we first define P_j^i as the probability that the last accessed block of object i belongs to the layer j, assuming that $N^i > 0$. Then, $P_j^i = \frac{N_j^i - N_{j+1}^i}{N^i}$. If the last data block belongs to the layer j, the average value of Δ, δ_j^i, becomes $\sum_{k=1}^{j-1} L_k^i + \lfloor \frac{L_j^i}{N_j^i+1} N_j^i \rfloor$(Fig. 4). Hence, the expected value of Δ in object i becomes $E[\delta_j^i] = \sum_{j=1}^{l} \left(P_j^i \cdot \left(\lfloor \frac{L_j^i}{N_j^i+1} N_j^i + \sum_{k=1}^{j-1} L_k^i \rfloor \right)\right)$. Also, if the last accessed block of object i belongs to the layer j, the *distance A* becomes $d_A^{i,j} = \sum_{k=1}^{l} L_k^i - (\sum_{k=1}^{j-1} L_k^i + \lfloor \frac{L_j^i}{N_j^i+1} N_j^i \rfloor) = \sum_{k=j+1}^{l} L_k^i + (L_j^i - \lfloor \frac{L_j^i}{N_j^i+1} N_j^i \rfloor) \approx \sum_{k=j+1}^{l} L_k^i + \lfloor \frac{L_j^i}{N_j^i+1} \rfloor$, in average. The expected value of *distance A* in object i, $E[d_A^{i,j}]$, can be computed as $\sum_{j=1}^{l} P_j^i \cdot (\lfloor \frac{L_j^i}{N_j^i+1} \rfloor + \sum_{k=j+1}^{l} L_k^i)$. Hence, the expected inter seek overhead in object i can be represented as $T(\lceil E[d_A^{i,j}] \rceil)$. When $N^i = 0$ and $\Delta = 0$, we can regard the first cylinder of the object i as the *last-read cylinder* of the object. Hence, the *distance A* is the size of object i, L_o^i, and the inter seek overhead becomes $T(L_o^i)$. Combining those two cases, we can represent the inter seek overhead in object i, T_{inter}^i as in Eq. 1.

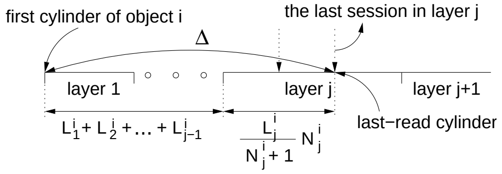

Figure 4: The value of Δ in object i in the Progressive placement scheme

$$T_{inter}^i = \begin{cases} T(\lceil E[d_A^{i,j}] \rceil) & \text{if } N^i > 0 \\ T(L_o^i) & \text{if } N^i = 0 \end{cases} \tag{1}$$

Finally, the sum of all intra seek overheads and inter seek overheads in each object in a round can be represented as $\sum_{i=w}^{z} \sum_{j=1}^{l} T_{intra,j}^i + \sum_{i=w}^{z-1} T_{inter}^i$. Since the distance of the return seek is approximated as the sum of the distance from the first cylinder of object w to the last cylinder of object $z-1$, $\sum_{i=w}^{z-1} L_o^i$, and Δ in object z, the average return seek overhead can be calculated as $T\left(\left\lceil \sum_{i=w}^{z-1} L_o^i + E[\delta_j^i] \right\rceil\right)$. Finally, the overall seek overhead of Progressive scheme in a round can be modeled as in Eq. 2.

$$T_{overhead} = \sum_{i=w}^{z} \sum_{j=1}^{l} T_{intra,j}^i + \sum_{i=w}^{z-1} T_{inter}^i + T\left(\left\lceil \sum_{i=w}^{z-1} L_o^i + E[\delta_j^i] \right\rceil\right) \tag{2}$$

3.3 Interleaved Placement Scheme

In Interleaved placement scheme, data blocks in the same segment are clustered together. The intra seek distance mainly depends on the interval between two adjacent sessions. Average interval between two adjacent sessions accessing the same

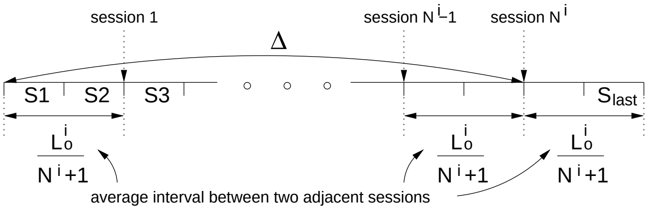

Figure 5: The value of Δ in object *i* in the Interleaved placement scheme

object depends both on the size of the file and the number of concurrent sessions. To compute the worst case seek time behavior, we assume that the access position of individual playbacks are evenly distributed. Since seek time profile is convex function, given a total seek distance, total seek time is maximized when the all seeks are of the same distance. The average intra seek distance in object *i* can be calculated simply as $\lfloor \frac{L_o^i}{N^i+1} \rfloor$. The number of seeks in a round using Interleaved placement scheme is $\min\{N^i, L_o^i\}$. Therefore, the total intra seek overhead during reading data blocks of object *i*, T_{intra}^i, can be calculated as $T_{intra}^i = T\left((\lfloor \frac{L_o^i}{\min\{N^i+1, L_o^i\}} \rfloor\right) \cdot \min\{N^i, L_o^i\}$. The value of Δ in object *i* can be calculated as $\lfloor \frac{L_o^i}{N^i+1} \cdot N^i \rfloor$(Fig. 5).

Hence, the *distance A* for inter seek in object *i* becomes $\lfloor L_o^i - \frac{L_o^i}{N^i+1} \cdot N^i \rfloor$ and the inter seek overhead, T_{inter}^i, can be calculated as $T_{inter}^i = T\left(\lfloor L_o^i - \frac{L_o^i}{N^i+1} \cdot N^i \rfloor\right)$. Therefore, the total seek overhead in object *i*, sum of the intra seek overhead and inter seek overhead, is $T_{intra}^i + T_{inter}^i$. The return seek distance of Interleaved placement scheme is the sum of the value of Δ in object z and the sum of object sizes from object w to object $z-1$. Hence, the return seek overhead of Interleaved placement scheme is $T\left(\sum_{i=w}^{z-1} L_o^i + \lfloor \frac{L_o^z}{N^z+1} \cdot N^z \rfloor\right)$. Putting all together, the total seek overhead of Interleaved placement scheme can be modeled as in Eq. 3.

$$T_{overhead} = \sum_{i=w}^{z} T_{intra}^i + \sum_{i=w}^{z-1} T_{inter}^i + T\left(\sum_{i=w}^{z-1} L_o^i + \lfloor \frac{L_o^z}{N^z+1} \cdot N^z \rfloor\right) \quad (3)$$

3.4 Harmonic Placement Scheme

In Harmonic placement schemes, the layers are partitioned into two groups, namely, *lower* and *upper* layer. We apply different placement scheme for each of lower layers and upper layers. This is primarily to exploit the access frequency and access pattern of the respective layers. Interleaved placement scheme is used for placing the information in lower layers and Progressive scheme is used in placing the information in upper layers. Seek behavior for lower and upper layer accesses will be similar to the seek behavior of the Interleaved and progressive placement scheme. Let *low* and L_{lower}^i be the number of lower layers and the number of cylinders that data blocks belonging to the lower layers in object *i* are stored, respectively. Then $L_{lower}^i = \sum_{j=1}^{low} L_j^i$.

The intra seek overhead that occurs in lower layer of object *i*, $T_{intra,low}^i$, can be calculated, in a similar fashion to that of Interleaved placement scheme, as $T_{intra,low}^i = T\left(\lfloor \frac{L_{lower}^i}{\min\{N^i+1, L_{lower}^i\}} \rfloor\right) \cdot \min\{N^i, L_{lower}^i\}$. Also, the intra seek overhead that occurs in upper layer *j* of object *i*, $T_{intra,j}$, can be calculated, in a similar fashion to that of Progressive scheme, as $T_{intra,j}^i = T\left(\lfloor \frac{L_j^i}{\min\{N_j^i+1, L_j^i\}} \rfloor\right) \cdot \min\{N_j^i, L_j^i\}$. The total intra seek overheads that occur in all upper layers of object *i* becomes $\sum_{j=low+1}^{l} T_{intra,j}^i$.

Assume that $N^i > 0$. Then, the last data block read in a round in an object belongs to either lower layers or upper layers. When it belongs to the lower layers, Δ of an object will be similar to that of the Interleaved placement scheme. If it belongs to the upper layers, Δ can be obtained in similar fashion to the Progressive scheme. The average value of Δ of object *i* of Harmonic scheme becomes $E[\delta_j^i] = \sum_{j=1}^{low} P_j^i \cdot \lfloor \frac{L_{lower}^i}{N^i+1} \cdot N^i \rfloor + \sum_{j=low+1}^{l} P_j^i \cdot \lfloor \frac{L_j^i}{N_j^i+1} N_j^i + \sum_{k=1}^{j-1} L_k^i \rfloor$. When the *last-read cylinder* belongs to the lower layer, the *distance A* of object *i* for inter seek becomes $d_A^{i,j} = L_o^i - \sum_{j=1}^{low} \lfloor \frac{L_{lower}^i}{N^i+1} \cdot N^i \rfloor \approx \sum_{j=1}^{low} (\lfloor L_o^i - \frac{L_{lower}^i}{N^i+1} \cdot N^i \rfloor)$. When the *last-read cylinder* of object *i* belongs to the upper layer, it becomes $d_A^{i,j} = L_o^i - \sum_{j=low+1}^{l} \lfloor \frac{L_j^i}{N_j^i+1} N_j^i + \sum_{k=1}^{j-1} L_k^i \rfloor \approx \sum_{j=low+1}^{l} (\lfloor \frac{L_j^i}{N_j^i+1} + \sum_{k=j+1}^{l} L_k^i \rfloor)$. Hence, the expected length of *distance A* for inter seek in object *i* becomes $E[d_A^{i,j}] = \sum_{j=1}^{low} P_j^i \cdot (\lfloor L_o^i - \frac{L_{lower}^i}{N^i+1} \cdot N^i \rfloor) + \sum_{j=low+1}^{l} P_j^i \cdot (\lfloor \frac{L_j^i}{N_j^i+1} + \sum_{k=j+1}^{l} L_k^i \rfloor)$. When $N^i = 0$, $\Delta = 0$ and *distance A* becomes L_o^i. Therefore, the inter seek overhead in object *i* of Harmonic scheme, T_{inter}^i, can be calculated as in Eq. 4.

$$T_{inter}^i = \begin{cases} T\left(E[d_A^{i,j}]\right) & \text{if } N^i > 0 \\ T(L_o^i) & \text{if } N^i = 0 \end{cases} \quad (4)$$

Hence, the total seek overhead in object *i*, sum of the intra seek overhead and inter seek overhead, is $T_{intra,low}^i + \sum_{j=low+1}^{l} T_{intra,j}^i + T_{inter}^i$, and summing up all of them of each object, we can get the total seek overhead in a round except the return seek overhead as in $\sum_{i=w}^{z} \left(T_{intra,low}^i + \sum_{j=low+1}^{l} T_{intra,j}^i\right) + \sum_{i=w}^{z-1} T_{inter}^i$.

The return seek distance of Harmonic scheme is the sum of the value of Δ of object z and the sum of object sizes from object w to object $z-1$. Hence, the return seek overhead of Harmonic scheme is formulated as $T\left(\left\lceil \sum_{i=w}^{z-1} L_o^i + E[\delta_j^z] \right\rceil\right)$.

Putting all together, the total seek overhead of Harmonic scheme in a round can be modeled as in Eq. 5.

$$\begin{aligned} T_{overhead} &= \sum_{i=w}^{z} \left(T_{intra,low}^i + \sum_{j=low+1}^{l} T_{intra,j}^i\right) \\ &+ \sum_{i=w}^{z-1} T_{inter}^i + T\left(\left\lceil \sum_{i=w}^{z-1} L_o^i + E[\delta_j^z] \right\rceil\right) \end{aligned} \quad (5)$$

4. PERFORMANCE ANALYSIS

4.1 Experiment Setup

Our experiment mainly focus on two issues. First, we like to verify the accuracy of the analytical models developed in this work. Second, more importantly, we examine the performance of the individual placement schemes.

The network bandwidth availability trace is obtained via network simulator, NS. Our network topology has dumbbell setting. In our network setting, a single server is servicing multimedia streaming clients using RAP(Rate Adaptive Protocol)[8]. The server and the client is connected via one link. To simulate the actual network environment, we introduce ten ftp sessions which share the link. There are ten ftp server nodes and ten ftp client nodes. Each server and client pair forms a single session. In our simulation, we use four different bottleneck link bandwidths: 2.6 Mbits/sec, 7.7 Mbits/sec, 15.4 Mbits/sec and 30 Mbits/sec. Each of the bottleneck link bandwidths are chosen to represent typical subscriber line bandwidth capacity: ISDN(128 Kbits/sec), or different speed DSL(or Cable Modem) subscriptions(384 Kbits/sec, 768 Kbits/sec and 1.5 Mbits/sec). And the layer size increases exponentially as the layer index increases[4]. Hence, when we use Harmonic placement scheme, a server that provides streaming services to clients whose subscriber line bandwidth is 128 Kbps may store multimedia objects putting only layer 1 into the lower layers set and other layers into the upper layers set. In this case, the Harmonic placement scheme becomes exactly the same as the Progressive placement scheme. A server that serves clients whose subscriber line bandwidth is 384 Kbps may put layer 1 and layer 2 into the lower layers set and layer 3 and layer 4 into the upper layers set. A server that serves clients whose subscriber line bandwidth is 768 Kbps may put layer 1, layer 2 and layer 3 into the lower layers set and only layer 4 into the upper layers set. Finally, a server that serves clients whose subscriber line bandwidth is 1.5 Mbps may store multimedia objects putting all layers into the lower layers set. In this case, the Harmonic placement scheme becomes exactly the same as the Interleaving placement scheme. In this way, using Harmonic placement scheme, four different kinds of servers that store multimedia objects in four different ways provide streaming service to four groups of clients whose subscriber line bandwidth is different each other. Since large scale ISPs (Internet Service Providers) that provides streaming service to clients operate more than tens of homogeneous streaming servers in general, they can use Harmonic placement scheme only through reforming the homogeneous server farm into the heterogeneous server farm without additional hardware cost.

We can verify the effectiveness of three placement strategies under different client bandwidth capability. Using the bandwidth availability trace, we generate layer access patterns for individual clients and subsequently a sequence of I/O requests on the streaming server. With this I/O trace, we examine the disk utilization. Disk utilization is obtained via two different ways: analytical model and physical experiment. With a given I/O sequence, we can compute the disk overhead and disk transfer time using the models developed in this work. We also measure the behavior of the disk subsystem.

4.2 Experiment Results

Fig. 6 shows the results of analytical models and physical experiments under three placement schemes. There are thirty concurrent sessions accessing ten video objects. They are uniformly distributed. In physical experiment, we found that hard disk drive with Progressive placement is saturated under thirty concurrent sessions (1.5 Mbits/sec subscriber line capacity). Subsequent experiment is based upon the workload generated by thirty sessions.[1]

As can be seen in Fig. 6, the disk operation efficiency obtained from the analytical models are very close to the results obtained from physical experiments. Among three schemes, the Harmonic placement scheme shows the best performance in all subscriber line capacities. When the subscriber line bandwidth is relatively small, the Progressive placement scheme shows better performance than the Interleaved placement scheme. Since, in the Progressive placement scheme, data blocks in the same layer are stored contiguously, the average seek distance is smaller than that in the Interleaved placement scheme when the subscriber line bandwidth is 128 Kbps, where only layer 1 data blocks are read, or 384 Kbps, where only layer 1 and layer 2 data blocks are read. On the other hands, when the subscriber line bandwidth is relatively large, the Progressive placement scheme generates more number of seeks than Interleaved placement scheme and the distance of the individual seeks increases as well. The Interleaved placement scheme stores data blocks of all layers in the same segment contiguously. Therefore, as the subscriber line bandwidth becomes larger, the Interleaved placement scheme shows better performance than the Progressive placement scheme. The Harmonic placement scheme contiguously stores data blocks that are supposed to be read contiguously in each subscriber line bandwidth. Hence, the Harmonic placement scheme always outperforms the other schemes.

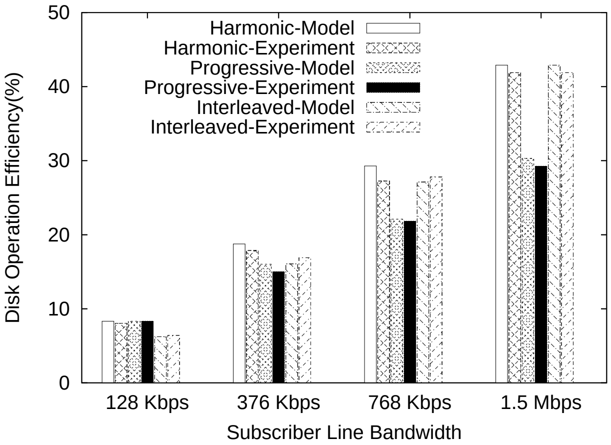

Figure 6: Disk operation efficiency of individual placement schemes: analytical model and physical experiment: 30 concurrent sessions accessing different objects with uniform probability

Fig. 7 shows the disk operation efficiency under varying number of subscribers. The number of sessions varies from 5 to 30. As we can see from all figures, the Harmonic placement scheme shows better disk operation efficiency regardless of the number of sessions. As the number of sessions increases, not only the disk operation efficiency of each scheme increases but also the difference of disk operation efficiency among each scheme increases. When the subscriber line bandwidth is 128 Kbps, the Progressive placement scheme outperforms Interleaved placement scheme and in other cases the Interleaved placement scheme outperforms the Progressive scheme.

Fig. 8 shows the maximum number of concurrent sessions in individual placement schemes. When the subscriber line bandwidth is small(128Kbits/sec), Harmonic and Progressive Placement scheme yield the best performance. This is because only fraction of the objects are accessed due to bandwidth limitation. On the other hand, when the subscriber line bandwidth is sufficient(1.5 Mbits/sec), Interleaved and harmonic placement yield the best performance. When the subscriber line bandwidth is large, relatively significant fraction of the file is accessed and therefore sequential playback on a file yields

---

[1] Actually, the disk operation efficiency of each scheme becomes larger when the number of sessions are larger than 30. We only shows a snapshot of efficiency spectrum in Fig. 6.

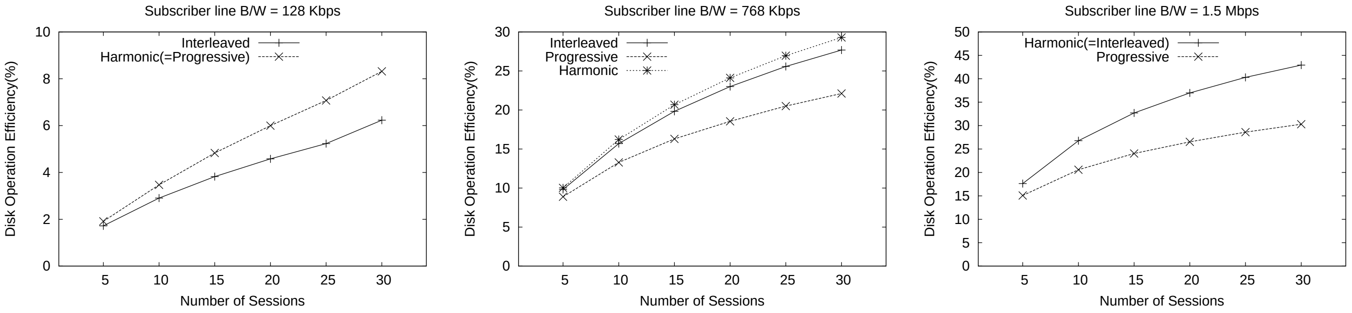

Figure 7: Disk operation efficiency of each placement scheme according to the number of simultaneous sessions

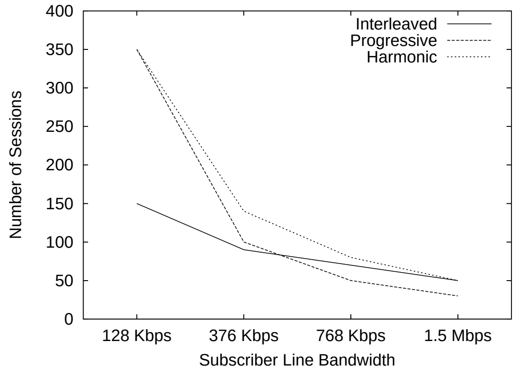

Figure 8: Maximum number of sessions

almost sequential disk access.

It is worth noting that the differences among the individual placement strategies become more significant under low subscriber line bandwidth. In fact, this phenomenon confirmed by the analytical model as well as physical experiment put forth an important sign in placement strategy. When a server may have to service large number of low bandwidth sessions, special care needs to be taken to properly partition the set of layers and to use harmonic placement strategy.

The Interleaved placement scheme should be avoided especially when we are to provide low-quality video streaming service or music streaming service to clients.

5. CONCLUSION

In this work, we aim at filling up the chasm between the file system technology and scalable streaming technology. Our thesis in this paper is on if it is possible to support scalable streaming in more efficient fashion from file system's point of view. We propose novel file organization, "Harmonic Placement". Harmonic Placement clusters the frequently accessed layers together. We develop elaborate performance models for three file organization schemes which effectively capture the file system behavior. The accuracy of the model is confirmed via physical experiment. To examine the performance of the given placement schemes, we physically measure the disk utilization behavior and also compute the disk utilization using the analytical model developed in this work. The result obtained from the physical experiment and the analytical model lie within very close proximity. It is found that in all cases, Harmonic placement exhibits the best performance. The result of our work provides important guidance on streaming server planning and storage management.

6. REFERENCES

[1] A. Balk, M. Gerla, D. Maggiorini, and M. Sanadidi. Adaptive video streaming: pre-encoded mpeg-4 with bandwidth scaling. *Computer Networks*, 44:415–439, 2004.

[2] P. de Cuetos and K. W. Ross. Adaptive rate control for streaming stored fine-grained scalable video. In *Proceedings of the 12th international workshop on Network and operating systems support for digital audio and video*, pages 3–12. ACM Press, 2002.

[3] F. Fitzek, P. Seeling, and M. Reisslein. Video streaming in wireless internet. *Wireless Internet: Technologies and Applications Series: Electrical Engineering and Applied Signal Processing Series*, 2004.

[4] R. N. Inc. Helix producer user's guide, June 2002.

[5] W. Li. Overview of Fine Granularity Scalability in MPEG-4 Video Standard. *IEEE Trans. on Circuits and Systems for Video Technology*, 11(3), March 2001.

[6] J. Liu, B. Li, Y. T. Hou, and I. Chlamtac. On optimal layering and banwidth allocation for multisession video broadcasting. *IEEE Trans. on Wireless Communication*, 3(2):656–667, March 2004.

[7] H. M. Radha, M. van der Schaar, and Y. Chen. The mpeg-4 fine-grained scalable video coding method for multimedia streaming over ip. *IEEE Trans. on Multimedia*, 3(1):53–58, March 2001.

[8] R. Rejaie, M. Handely, and D. Estrin. Rap: An end-to-end rate-based congestion control mechanism for realtime streams in the internet. In *Proceedings of IEEE Infocom*, pages 1337–1345, New York, NY, USA, March 1999.

[9] R. Rejaie and A. Ortega. Pals: Peer-to-peer adaptive layered streaming. In *Proceedings of Network and Operating System Support for Digital Audio and Video*, Monterey, CA. USA, June 2003.

[10] S. ryong Kang, Y. Zhang, M. Dai, and D. Loguinov. Multi-layer active queue management and congestion control for scalable video streaming. In *Proceedings of the 24th International Conference on Distributed Computing Systems(ICDCS'04)*, 2004.

[11] D. Saparilla and K. W. Ross. Optimal streaming of layered video. In *Proceedings of IEEE INFOCOM*, 2000.

[12] D. Wu, Y. T. Hou, and Y.-Q. Zhang. Scalable video coding and transport over broad-band wireless networks. *Proceedings of IEEE*, 89(1):6–15, Jan 2001.