Distributed Video Systems

Chapter 5

Issues in Video Storage and Retrieval

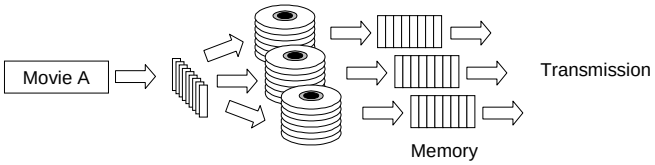
Part 3 - Multi-disk Video Server

Jack Yiu-bun Lee
Department of Information Engineering
The Chinese University of Hong Kong

Contents

5.1 Placement Policy

- Round-Robin
  - Common for push-based servers

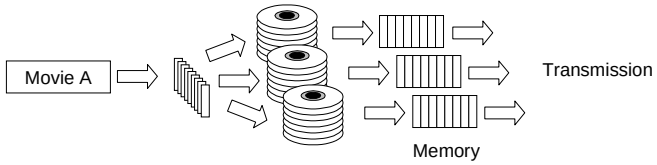


D_0	D_1	D_2
b_0	b_1	b_2
b_3	b_4	b_5
b_6	b_7	b_8
b_9	b_{10}	b_{11}
b_{12}	b_{13}	b_{14}

Stripe unit *i* of a video is placed at disk *i*%*d*, where d is the number of disks in the system.

5.1 Placement Policy

- Randomized
  - Employed in some pull-based servers

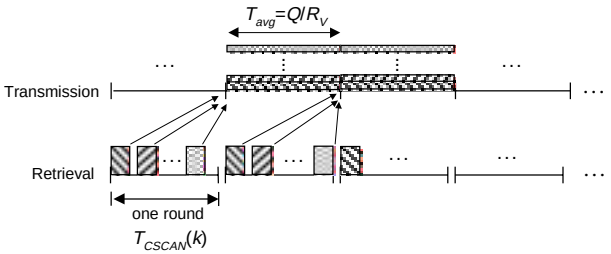


D_0	D_1	D_2
b_1	b_0	b_2
b_3	b_5	b_4
b_8	b_7	b_6
b_9	b_{10}	b_{11}
b_{13}	b_{14}	b_{12}

The order of stripe units in a parity group is permuted randomly.

5.2 Push-Based Designs

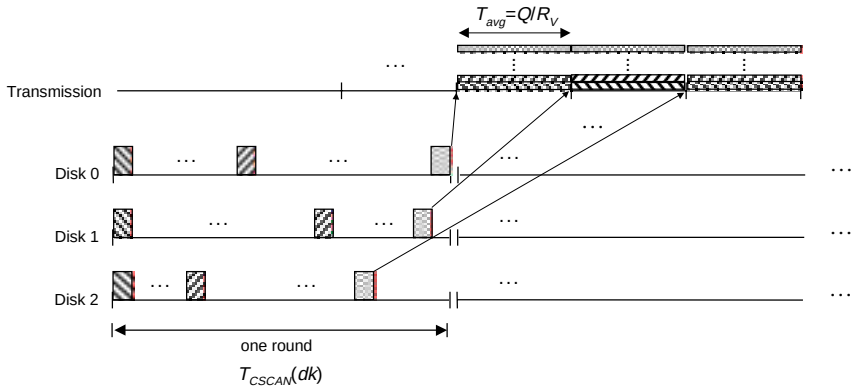
- Single-Disk Case
  - SCAN with *k* requests served per round



For continuity: $T_{CSCAN}(k) \le T_{avg}$

5.2 Push-Based Designs

- Disk-Striping Case
  - Concurrent Schedule (*d*=3 Disks)
    - SCAN with *dk* requests served per round



5.2 Push-Based Designs

- Disk-Striping Case
  - Concurrent Schedule (d=3 Disks)
    - Performance Gain
      - Higher throughput due to concurrent retrievals;
      - Average load balanced across all disks;
      - Lower seek-time overhead due to more (dk) requests served per SCAN round;

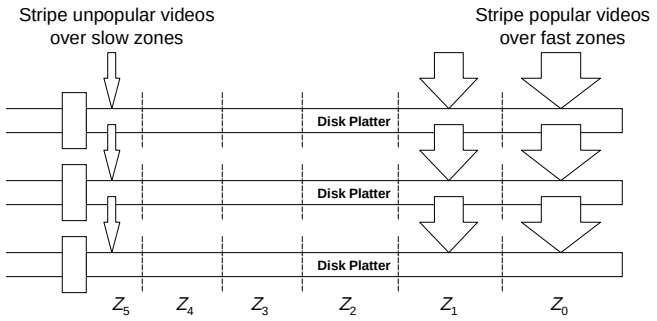
For continuity: $T_{CSCAN}(dk) \leq dT_{avg}$

But in general: $T_{CSCAN}(dk) \leq dT_{CSCAN}(k)$

Hence it may be possible to serve more requests under the disk-array case than the single-disk case.

5.2 Push-Based Designs

- Disk-Striping Case
  - Concurrent Schedule (d=3 Disks)
    - Performance Gain
      - Adapting to Disk Zoning

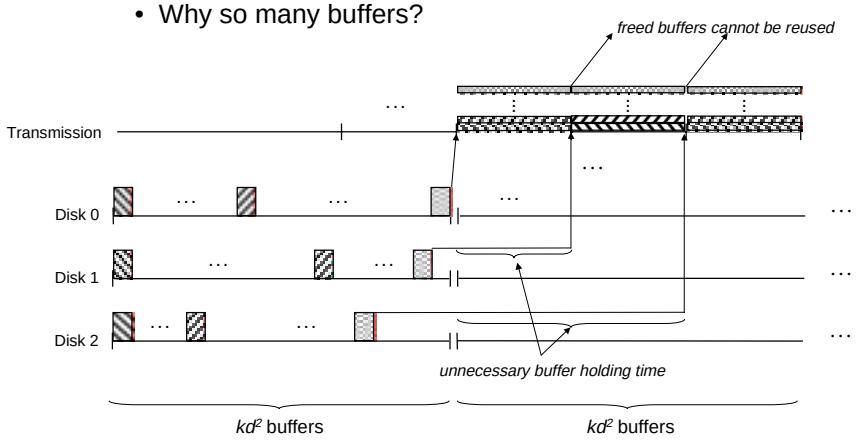


5.2 Push-Based Designs

- Disk-Striping Case
  - Concurrent Schedule (d=3 Disks)
    - Buffer Requirement
      - Double-buffering for SCAN;
      - dk requests served in a round in a disk;
      - total buffers = $d \times d \times k = 2d^2k$ buffers.
    - Scalability
      - 64KB stripe units, 20 requests per round;
      - 1 disk - Required buffer per disk = 2.5MB;
      - 8 disks - Required buffer per disk = 20MB!
    - Problem
      - Scalability is sub-linear because buffer requirement per disk increases for more disks.
      - But why?

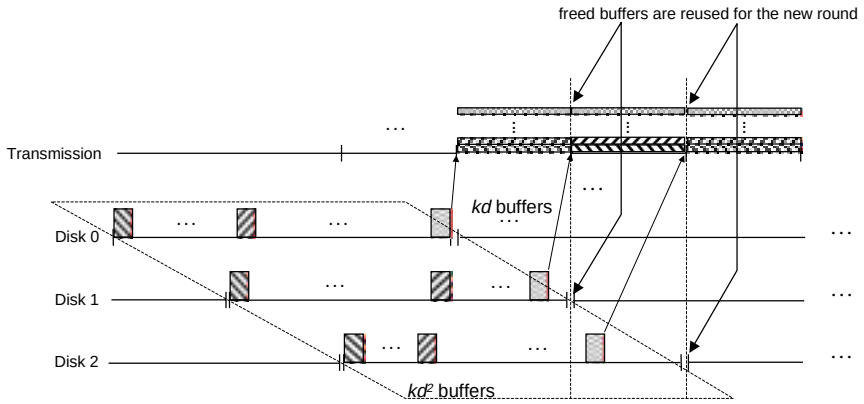
5.2 Push-Based Designs

- Disk-Striping Case
  - Concurrent Schedule (d=3 Disks)
    - Why so many buffers?



5.2 Push-Based Designs

- Disk-Striping Case
  - Offset Schedule (*d*=3 Disks)



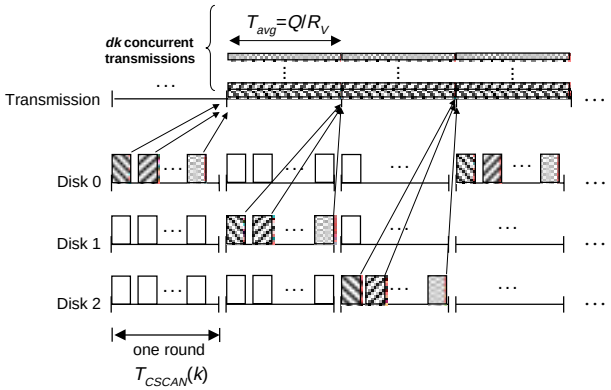
5.2 Push-Based Designs

- Disk-Striping Case
  - Offset Schedule (*d*=3 Disks)
    - Buffer Requirement
      - total buffers = (*d+1)dk* buffers.
    - Scalability
      - 64KB stripe units, 20 requests per round;
      - 1 disk - Required buffer per disk = 2.5MB;
      - 8 disks - Required buffer per disk = 11.25MB.
    - Better than concurrent schedule but still not linear!
    - Anything else we can do?
      - Reduce the number of requests served in a round.

5.2 Push-Based Designs

Jack Y.B. Lee

- Disk-Striping Case
  - Split Schedule (d=3 Disks)
    - Serves k requests per round but different groups of requests in subsequent rounds.



5.2 Push-Based Designs

Jack Y.B. Lee

- Disk-Striping Case
  - Split Schedule (d=3 Disks)
    - Buffer Requirement
      - total buffers = 2dk buffers.
    - Scalability
      - 64KB stripe units, 20 requests per round;
      - 1 disk - Required buffer per disk = 2.5MB;
      - d disks - Required buffer per disk = 2.5MB.
    - The buffer requirement per disk is finally fixed!
    - Hence the storage subsystem is scalable to a large number of disks.

5.2 Push-Based Designs

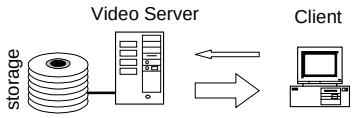
Jack Y.B. Lee

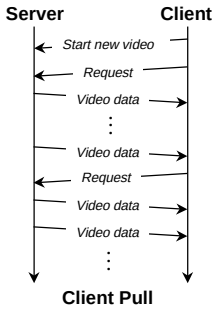
- Summary
  - Scheduling is simpler because
    - it is centralized at the server;
    - it is periodic and proceeds in rounds;
    - little randomness in the process;
  - Dimensioning is simpler because
    - the buffer requirement is well-defined;
    - continuity condition is simple;
      e.g. $T_{CSCAN}(k) \leq T_{avg}$
  - Most existing studies in the literature employs server-push models.

5.3 Pull-Based Designs

Jack Y.B. Lee

- System Model

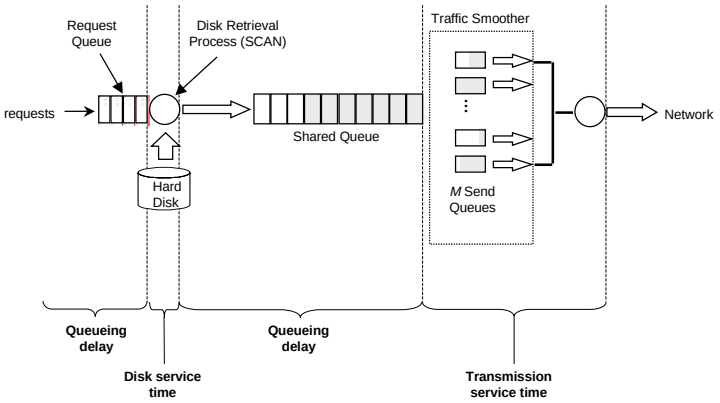




  - No centralized scheduler;
  - Retrieves and transmits data upon request;
  - More randomness in the process.

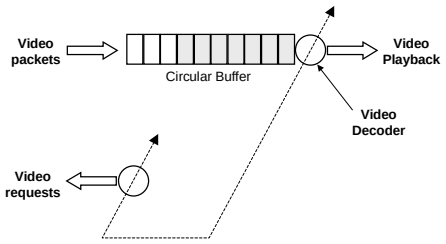
5.3 Pull-Based Designs

- Server Architecture



5.3 Pull-Based Designs

- Client Architecture
  - Data and request flows driven by the client:

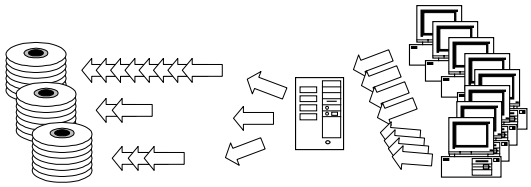


  - The circular buffer is prefetched with data before playback starts;
  - The prefetched data is used to absorb delay variations in the server, the network, etc.

5.3 Pull-Based Designs

Jack Y.B. Lee

- Challenges
  - System Dimensioning
    - Capacity planning
    - Server buffer requirement
    - Start-up delay
    - Server response time
    - Client buffer requirement
  - Load Balancing in Disk Array



5.3 Pull-Based Designs

Jack Y.B. Lee

- Load Balancing Tactics
  - Randomization
    - Random striping of video data
    - Statistically load balanced
    - Deterministic guarantees not possible
  - Forced Request Scheduling
    - Delay service of requests at server to ensure load balancing
    - Per-request scheduling
    - Extra scheduling delay incurred
  - Admission Scheduling
    - Control the time a client starts a new video session
    - Per-session scheduling
    - Periodicity of the request generation process assumed

5.4 Reliability Issues

Jack Y.B. Lee

- Reliable VoD Systems
  - Fault Containment
    - Ensure a single fault won't bring down the entire system.
    - Partial faults lead to partial failures.
    - E.g. partition and replication.
  - Fault Recovery
    - Able to restart service after a recovery process.
    - E.g. replication with fail-over, hierarchical storage.
  - Non-stop Service
    - Able to sustain existing services despite failures.
    - E.g. mirroring.

5.4 Reliability Issues

Jack Y.B. Lee

- Considerations in Applying RAID Schemes
  - Storage Overhead
    - Video storage is huge, so excessive overhead is economically undesirable.
  - Impact on I/O Performance
    - Video retrieval is I/O intensive so sacrificing too much I/O performance is undesirable. This may incur extra storage because more disks are required to meet the I/O demand.
  - Performance Degradation After Failure
    - Video systems require performance *guarantees*. Hence if disk performance degrades after failure, it will likely lead to service interruptions.
  - Hardware Requirement for Real-time Recovery
    - Complex erasure-correction codes (e.g. RS-Code) may not be economically feasible to implement for extremely high data rates.

5.4 Reliability Issues

Jack Y.B. Lee

- General Applicability of RAID Schemes
  - RAID-1 (Mirroring)
    - Non-stop service possible but expensive.
    - Suitable for I/O-bound systems with excessive storage.
  - RAID-2 (ECC)
    - Non-stop service possible but still expensive.
    - Unpopular with few commercial implementations.
  - RAID-3 (Bit-interleaved Parity)
    - Non-stop service possible, minimal storage overhead.
    - No performance degradation after a disk failure.
  - RAID-4 (Block-interleaved Parity)
    - Minimal storage overhead.
    - Performance degradation depends on disk placement policy and scheduling algorithm.

5.4 Reliability Issues

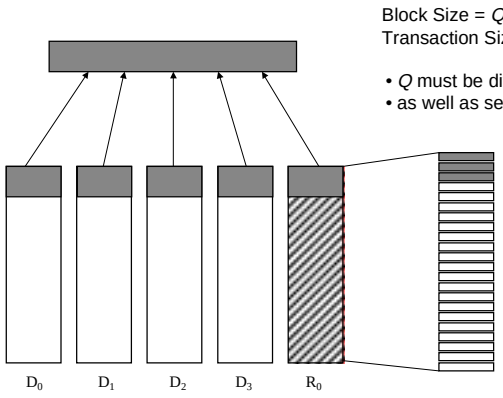
Jack Y.B. Lee

- General Applicability of RAID Schemes
  - RAID-5 (Block-interleaved Distributed Parity)
    - Minimal storage overhead.
    - Performance degradation depends on disk placement policy and scheduling algorithm.
    - Read performance gain over RAID-4 may not be usable in video applications.
  - RAID-6 (P+Q Redundancy)
    - Can tolerate double-disk failure.
    - Targeted for very large disk arrays.
    - Unpopular, few commercial implementations.
    - Application to video systems uncertain.

5.4 Reliability Issues

Jack Y.B. Lee

- Impact on I/O Performance
  - Fine-grained Striping Schemes (RAID-3)



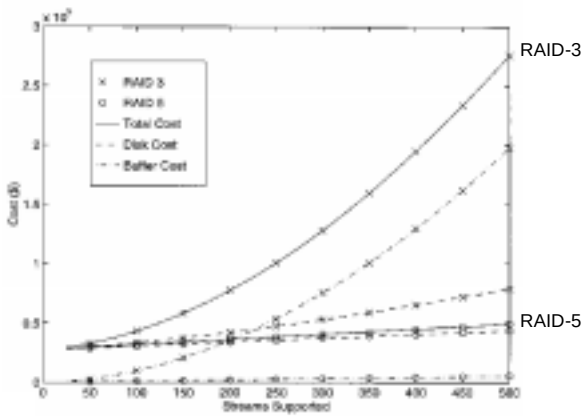
5.4 Reliability Issues

Jack Y.B. Lee

- Performance Comparisons
  - Fine-grained Striping Schemes (RAID-3)
    - Block size varies depending on number of disks in the array, and sector size of the disks.
    - To maintain I/O performance, we must maintain a relatively constant transaction size *q* at each disk.
    - This implies larger block size for more disks.
    - This incurs more buffer requirement at server and client.
  - Coarse-grained Striping Schemes (RAID-5)
    - Block size depends on sector size only and independent of the number of disks in the array.
    - I/O performance can be maintained without increasing buffer requirement.
    - Striping must be done in such a way that performance degradation do not occur after a disk failure.

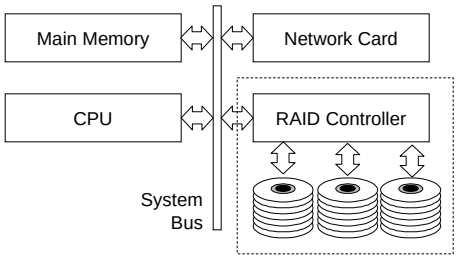
5.4 Reliability Issues

- Performance Comparisons
  - Includes disk and memory cost: (Barnett & Anido 1998)



5.4 Reliability Issues

- Performance Degradation After Failure
  - RAID-3 : None
  - RAID-5
    - Conventional RAID-5 Disk Array

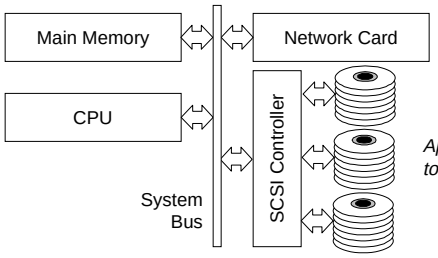


      - No control on data placement;
      - Fragmentation can lead to performance degradation after a disk failure.

5.4 Reliability Issues

Jack Y.B. Lee

- Performance Degradation After Failure
  - RAID-5
    - Software-Based RAID-5 Disk Array

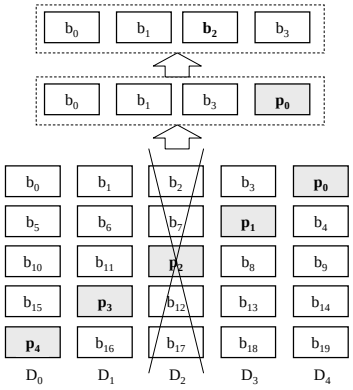


      - Application-level striping, software controllable data placement;
      - Software-based erasure correction (fast CPU needed);
      - Fragmentation does not lead to performance degradation after a disk failure.

5.4 Reliability Issues

Jack Y.B. Lee

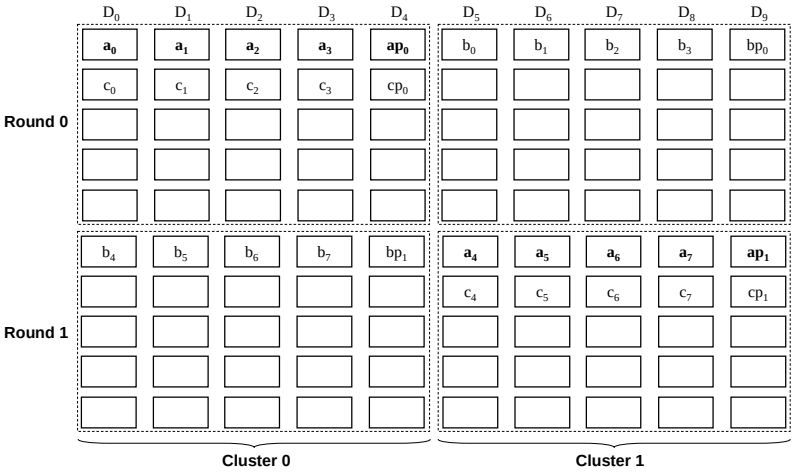
- Performance Degradation After Failure
  - RAID-5
    - Software-Based RAID-5 Disk Array
      - Buffer requirement is proportional to the parity group size.



5.5 Streaming RAID

Jack Y.B. Lee

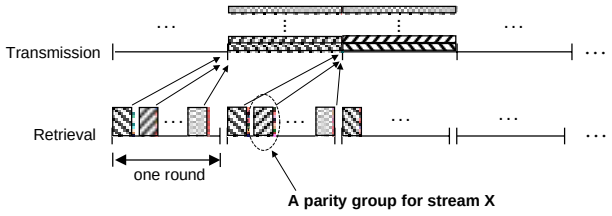
- Architecture
  - Multiple parity-group clusters with parity disk.



5.5 Streaming RAID

Jack Y.B. Lee

- Architecture
  - Scheduler

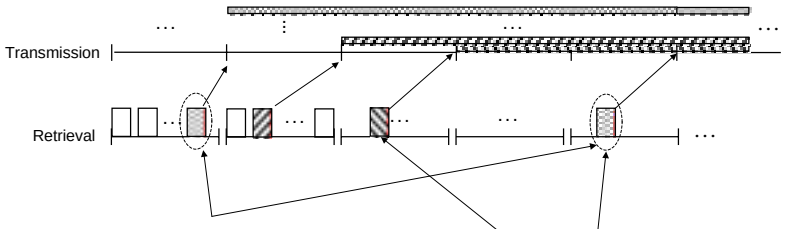


  - Buffer Requirement
    - Proportional to *d* and parity group size.

5.6 Staggered-Group Scheme

Jack Y.B. Lee

- Architecture
  - Scheduler

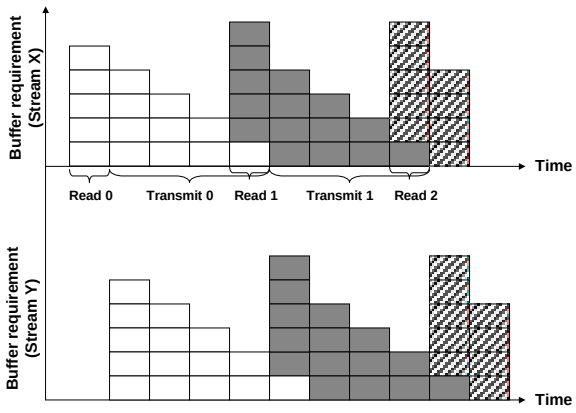


- **One parity group retrieved per *d* rounds for stream X.**
- **Transmission of the parity group spread across *d* rounds.**
- **Up to *n*/*d* streams are served in a disk round.**
- **Retrievals are staggered for stream X and Y.**

5.6 Staggered-Group Scheme

Jack Y.B. Lee

- Architecture
  - Buffer Requirement



Overall buffer requirement is reduced by half. Is there any tradeoff?

References

Jack Y.B. Lee

- The materials in this chapter are based on:
  - A.N.Mourad,
    "Issues in the design of a storage server for video-on-demand," *ACM Multimedia Systems,* vol.4:70-86, 1996.
  - S.A.Barnett, *et al.*,
    "Performability of disk-array-based video servers,"
    ACM Multimedia Systems, vol.6:60-74, 1998.
  - S.Berson, *et al.*,
    "Fault tolerant design of multimedia servers,"
    Proc. International Conference on Management of Data (SIGMOD), 1995.