CS 414 – Multimedia Systems Design
Lecture 27 – Media Server (Part 1)

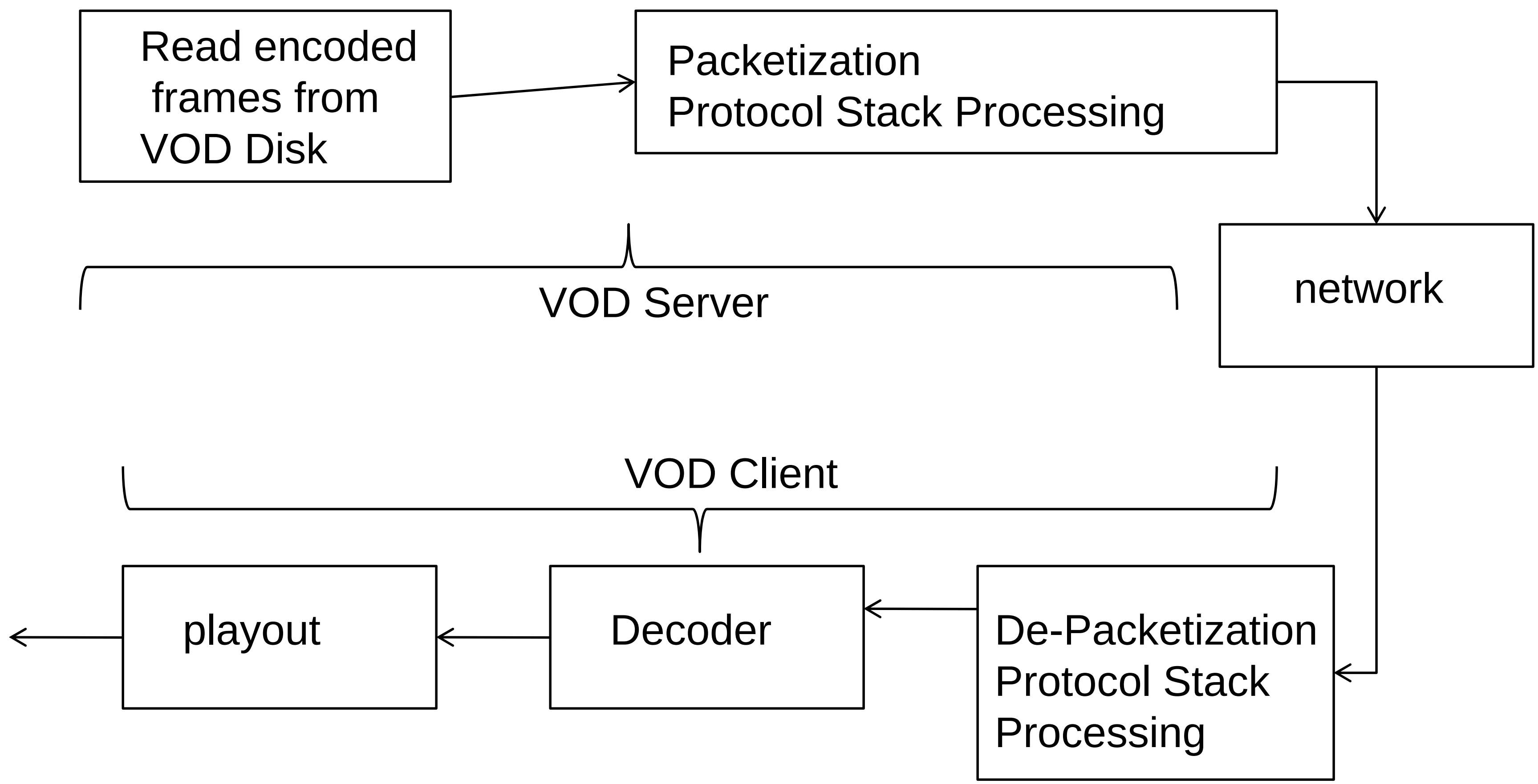
Klara Nahrstedt

Spring 2008

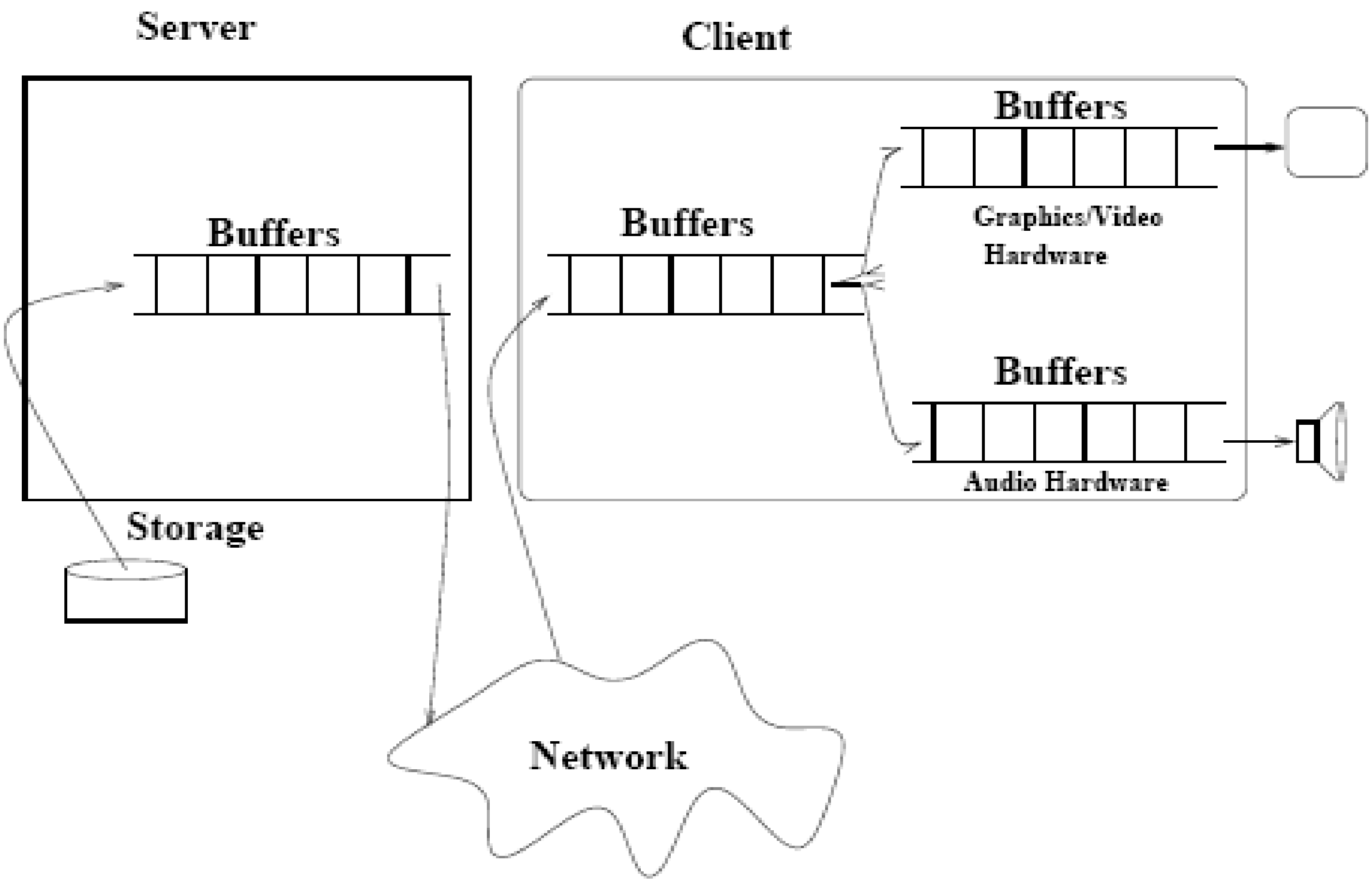
Administrative

- MP3 is out – deadline April 4
- Discussion Section – Monday, March 31

Buffering Strategies in Client-Server Systems



Client/Server Video-on-Demand System



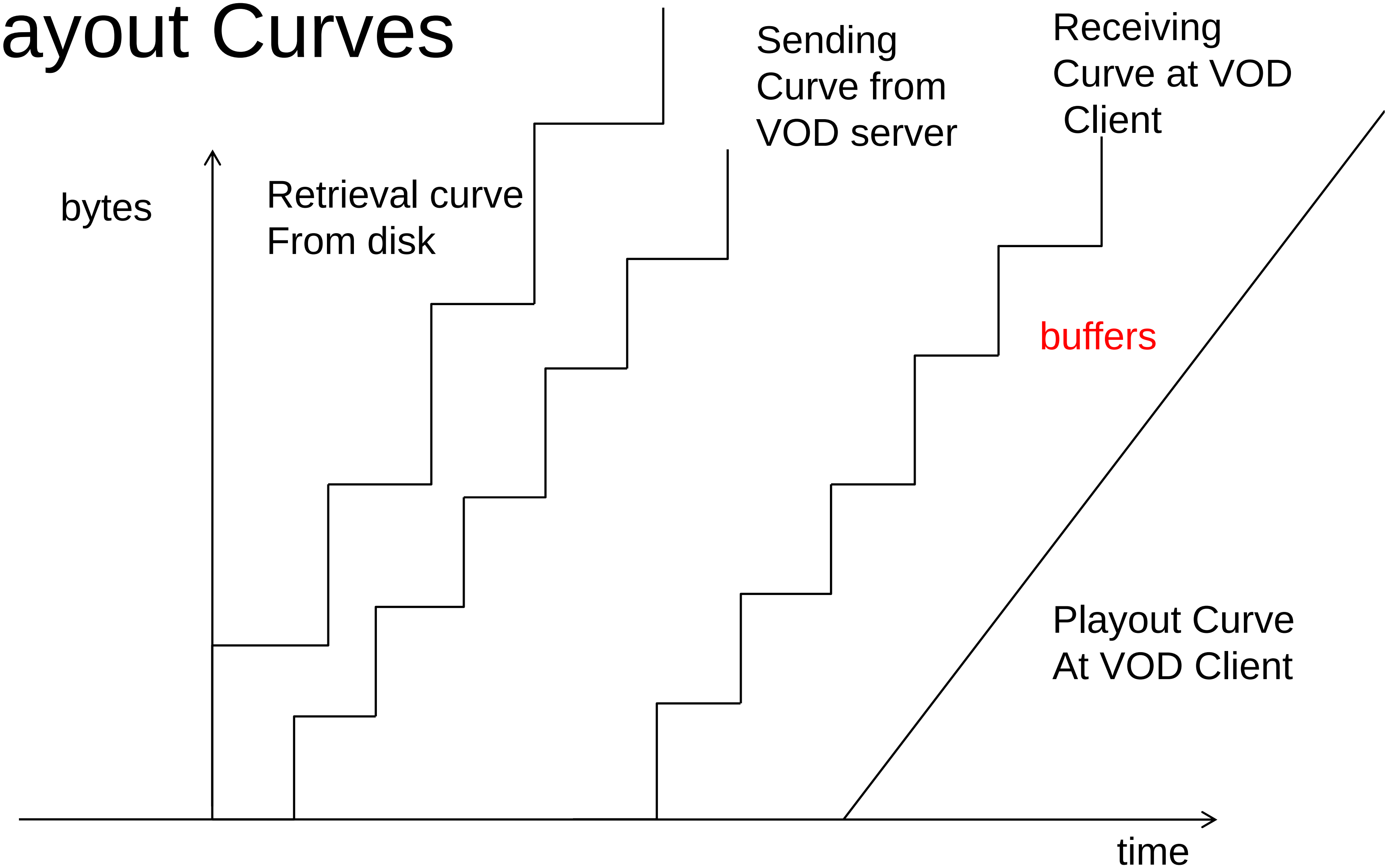
Servers Classification

- Media Servers
  - Push Servers – file servers with streaming model
  - Servers push data towards users
- Traditional File Servers
  - Pull Servers - FTP servers
  - Users pull data in one block at a time by repeatedly calling read to get one block after another
  - RPC Calls

Media Server Requirements

- Real-time storage and retrieval
  - Media quanta must be presented using the same timing sequence with which they were captured
- High-Data Transfer Rate and Large Storage Space
  - HDTV quality: 1280x720 pixels/frame; 24 bits/pixel -> 81 Mbytes per second
  - NTCS quality: 640x480 pixels/frame; 24 bits/pixel ->27MBytes per seconds

VOD Retrieval Transmission and Playout Curves



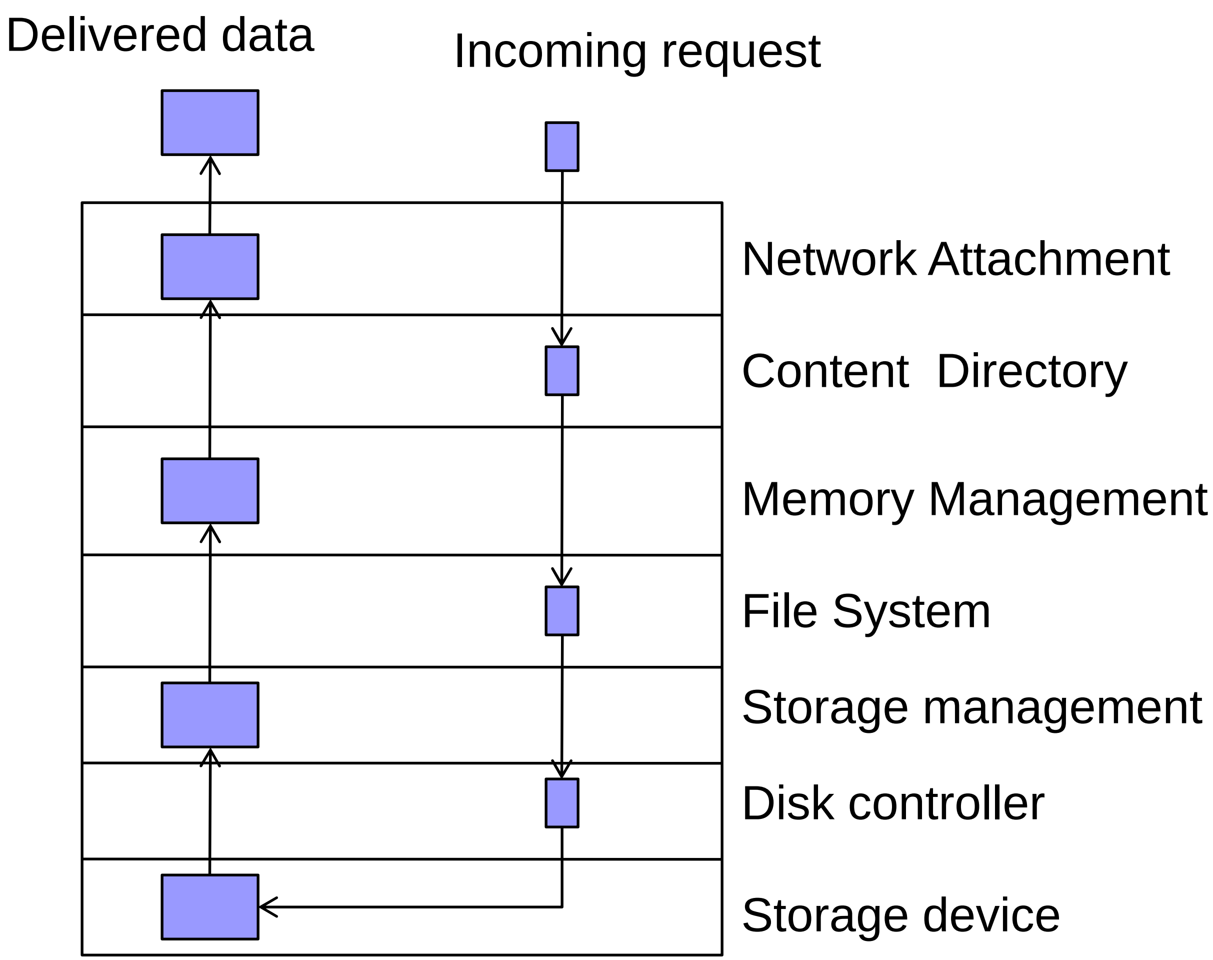
Playback

- Single Stream Playback
  - Possible approach – buffer the whole stream
    - Problem:??
  - Possible approach – prefetch just short video part
    - Problem:
      - Prevent starvation
      - Minimize buffer space requirement
      - Minimize initiation latency
- Multiple Streams Playback
  - Possible approach – dedicate a disk to each stream
    - Problem: ??
  - Possible approach – multiple streams per disk
    - Problems: ??

Support for Continuous Media

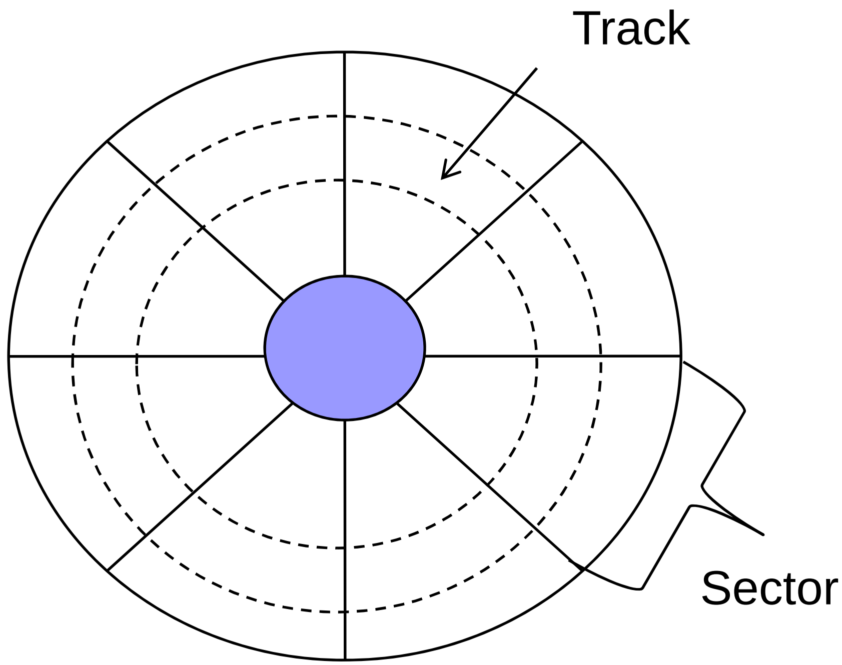
- Proper management of multimedia disk storage
  - Optimal placement of data blocks on disk
  - Usage of multiple disks
  - Role of tertiary storage
- Admission control
- Special disk scheduling algorithms and sufficient buffers to avoid jitter

Media Server Architecture



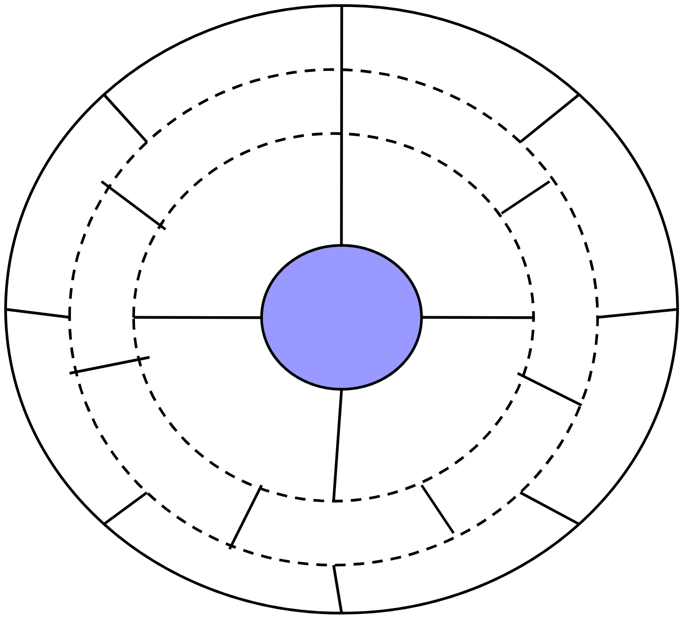
Disk Layout

Traditional Random Access
Disk Layout



Advantage: Easy mapping of location
Information to head movement and disk
rotation
Problem: loss of storage space

Zoned Disk (ZBR – Zone
Bit Recording)

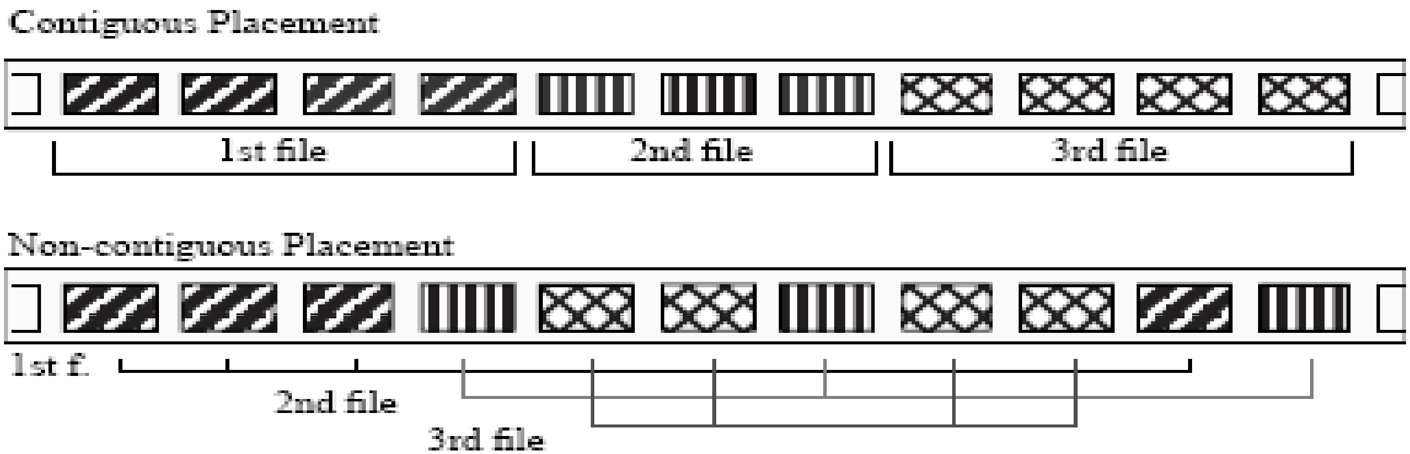


Advantage: Sector size same
Rotation speed constant; efficient
Usage of space

Storage Management

- Storage access time to read/write disk block is determined by 3 components
  - Seek Time
    - Time required for the movement of read/write head
  - Rotational Time (Latency Time)
    - Time during which transfer cannot proceed until the right block or sector rotates under read/write head
  - Data Transfer Time
    - Time needed for data to copy from disk into main memory

Placement of MM Data Blocks on Single Disk

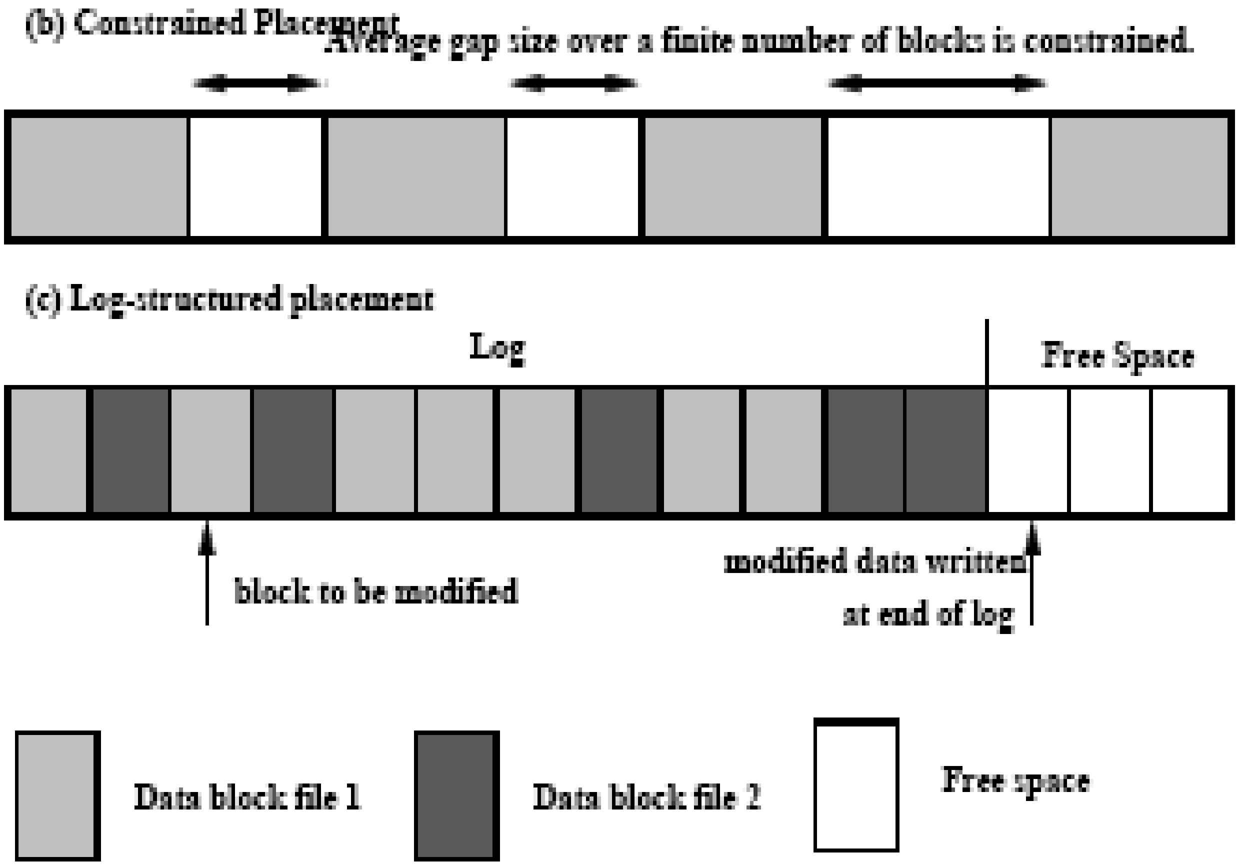


Continuous Placement	Scattered Placement
Simple to implement, but subject to fragmentation	Avoids fragmentation
Enormous copying overhead during insert/delete to maintain continuity	Avoid copying overhead
When reading file, only one seek required to position the disk head at the start of data	When reading file, seek operation incurs for each block , hence **intrafile seek**

Intra-file Seek Time

- Intra-file seek – can be avoided in scattered layout if the amount read from a stream always evenly divides block
- Solution: select sufficient large block and read one block in each round
  - If more than one block is required to prevent starvation prior to next read, deal with intra-file seek
- Solution: constrained placement or log-structure placement

Scattered Non-continuous Placement



Constrained Placement

- Approach: separation between successive file blocks is bounded
  - Bound on separation – not enforced for each pair of successive blocks, but only on average over finite sequence of blocks
  - Attractive for small block sizes
  - Implementation – expensive
- For constrained latency to yield full benefit, scheduling algorithm must retrieve immediately all blocks for a given stream before switching to another stream

Log-Structure Placement

- This approach writes modified blocks sequentially in a large contiguous space, instead of requiring seek for each block in stream when writing (recording)
  - Reduction of disk seeks
  - Large performance improvements during recording, editing video and audio
- Problem: bad performance during playback
- Implementation: complex

Placement of Multiple MM Files on Single Disk

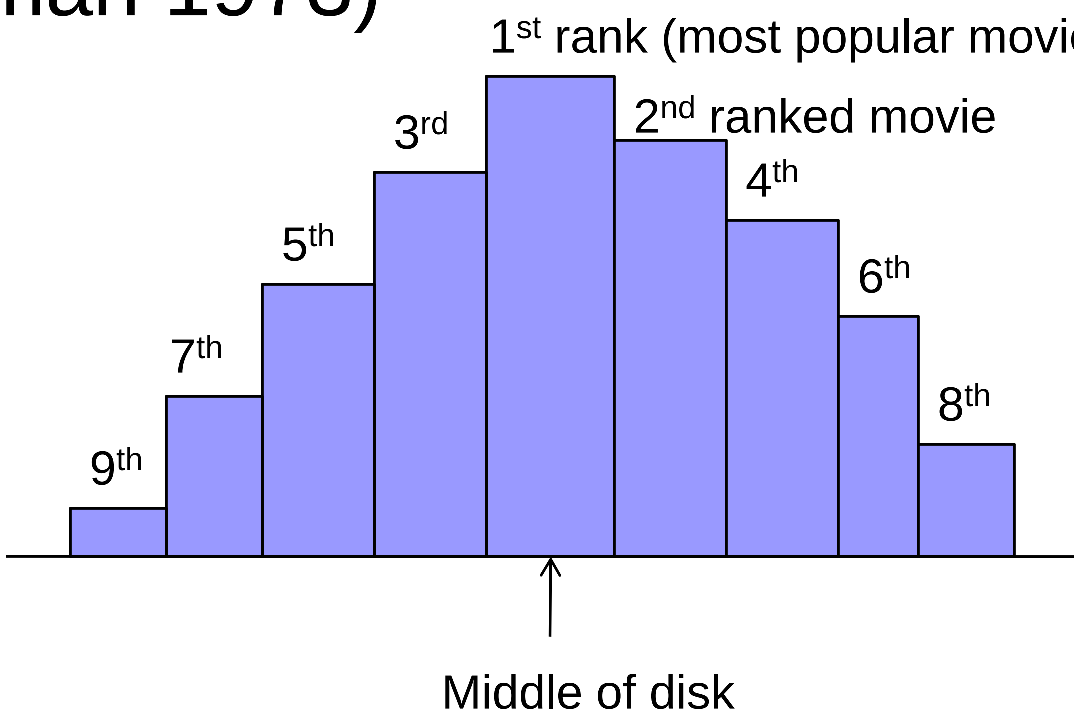
- Popularity concept among multimedia content - very important
- Take popularity into account when placing movies on disk
- Model of popularity distribution – Zipf's Law
  - Movies are k^{th} ranked
    - if their probability of customer usage is C/k,
      - C = normalization factor
    - Condition holds: C/1 + C/2 + … C/N = 1,
      - N is number of customers

Example

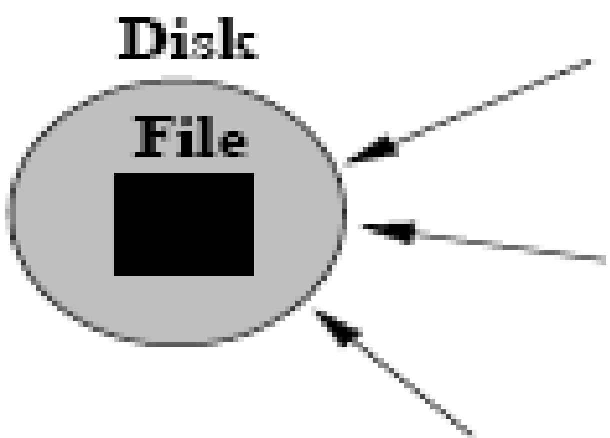
- Assume N = 5 movies
- Problem: what is the probability that the next customer picks 3^{rd} ranked movie?
- Solution:
  - Solve C from the equation
    - $C/1 + C/2 + C/3 + C/4 + C/5 = 1$
  - $C = 0.437$
  - Probability to pick 3^{rd} ranked movie is $C/3 = 0.437/3 = 0.1456$

Placement Algorithm for Multiple Files on Single Disk

- Organ-Pipe Algorithms (Grossman and Silverman 1973)

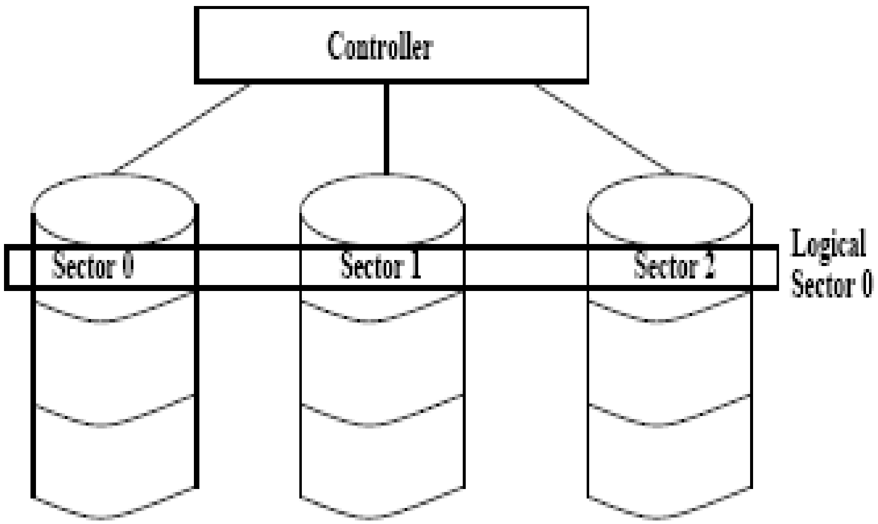


Need for Multiple Disks Solutions for Media Server



- Limitation of Single Disk: Disk Throughput
- Approach: 1 Maintain multiple copies of the same file on different disks
  - Very expensive
- Approach 2: Scatter multimedia file across multiple disks

Approach: Data Striping



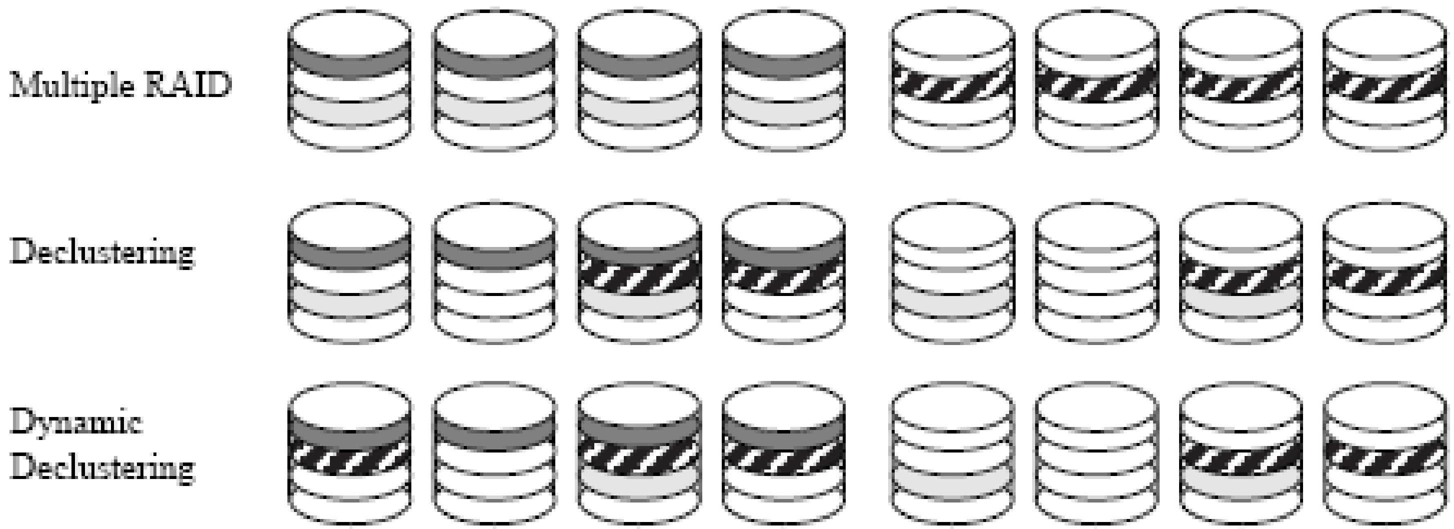
- RAID (Redundant Arrays of Inexpensive Disks)
  - Addresses both performance and security
  - (0-6) RAID levels – different approach at combining performance enhancements with security/fault-tolerance enhancements
- Disks spindle synchronously
  - Operate in lock-step parallel mode
- Striping improves BW, but does not improve seek or rotational delay

Data Striping – Group Creation

Multiple RAID: Creation of Subgroups of disks into independent logical disk arrays; limits # of disks per file

Declustering: Groups are not made up of complete disks ; # of disks for any stripe is fixed and of same size, but disks on which stripe is located differs

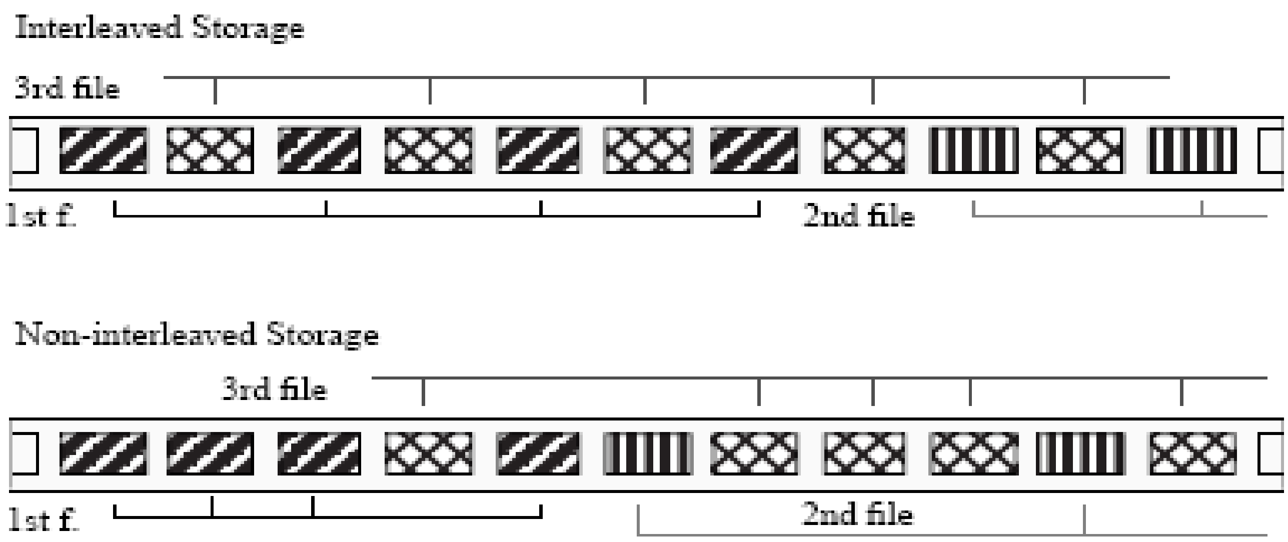
Dynamic Declustering: Non - static strip allocation to disks



Storage Management

- Disk access – slow and costly
- Reduce disk access
  - Use block caches (anticipate future reads or writes)
  - Reduce disk arm motion
    - Blocks accesses in sequence (continuously) , place together on one cylinder
    - Interleaved vs non-interleaved storage

Data Interleaving



Data Interleaving On single disk
(consecutive blocks are placed on The same cylinder But in interleaved way

Round k	Disk 1	Disk 2	Disk 3
1	File A, block 1	File B, block 1	File C, block 1
2	File C, block 2	File A, block 2	File B, block 2
3	File B, block 3	File C, block 3	File A, block 3
4	File A, block 4	File B, block 4	File C, block 4

Data Interleaving On Multiple Disks
(Disks are not Synchronized)

Conclusion

- Disk Layout and Number of disks play important role for media servers
- Data block placement and file placement are crucial for real-time retrieval on media servers