CS 414 – Multimedia Systems Design

Lecture 26 – Media Server (Part 1)

Klara Nahrstedt

Spring 2009

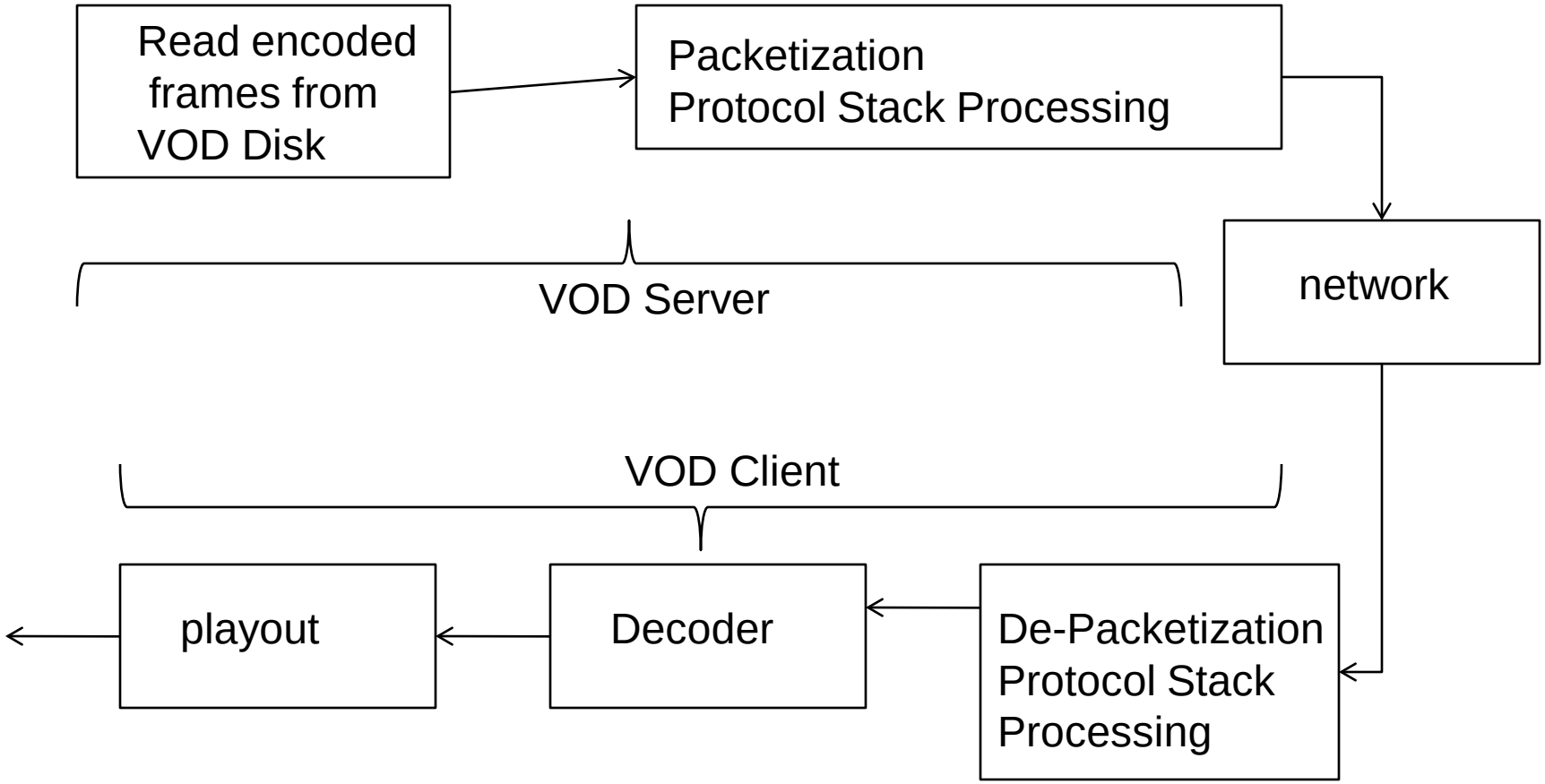
Administrative

- MP3 is out – deadline April 6

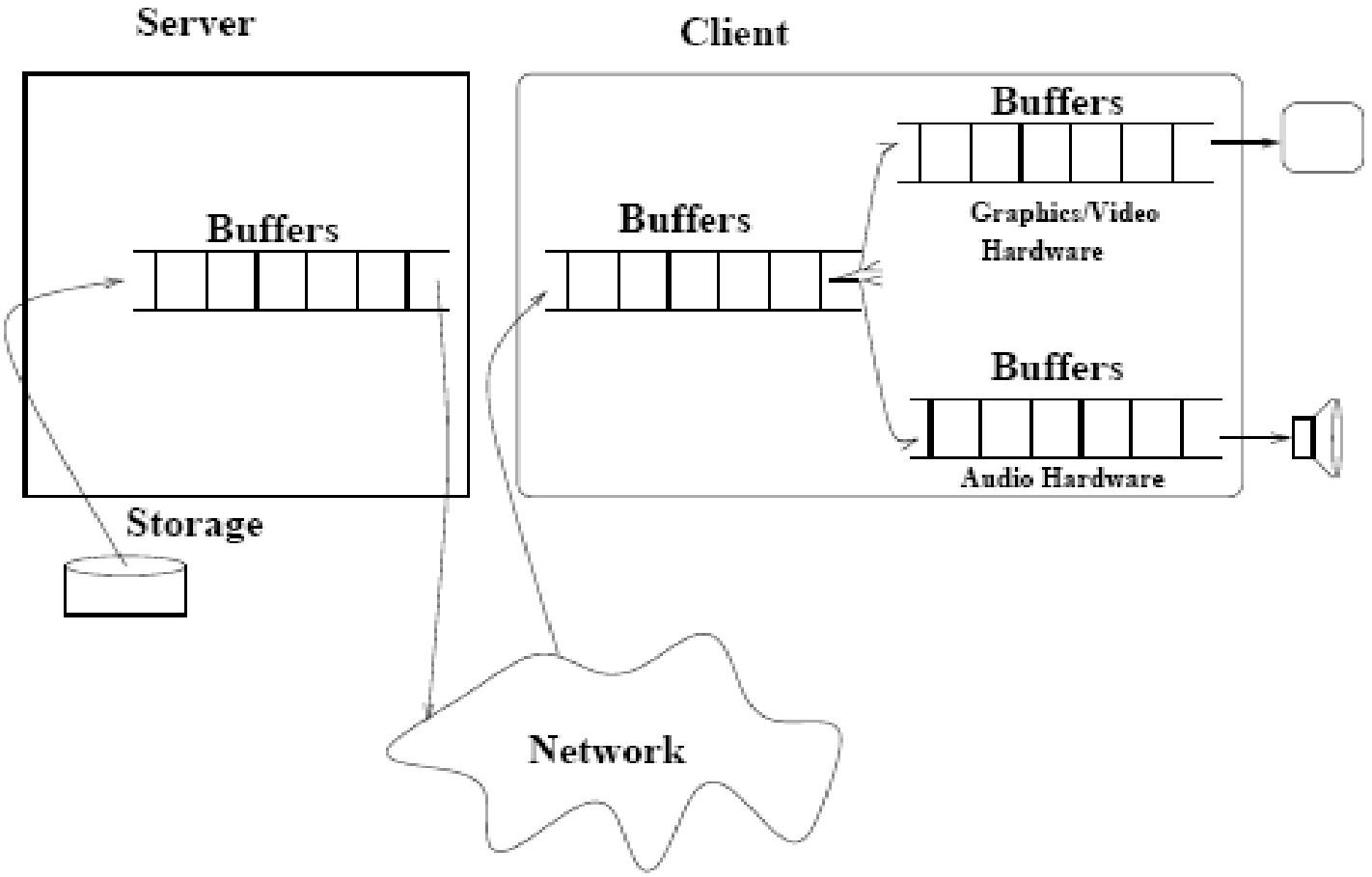
Outline

- Memory Management Issues for Multimedia Systems
- Introduction to Media Servers
  - Disk Layout
  - Disk/Storage Management
    - Multimedia Data Placement Strategies on Disk

Buffering Strategies in Client-Server Systems



Client/Server Video-on-Demand System



Memory Management

- Virtual memory versus real memory paradigm
- In VM – paging introduces unpredictable delays
- Multimedia timing requirements suggest that no paging is desired, however it might be difficult to do from user space
- Multimedia applications may want to pin pages into memory which include their time-sensitive code
- On-Chip Caching is desired
  - Intel Pentium Processor with MMX technology (SIMD Instruction Set) has on-chip cache of size 32KB for video processing only

Memory Reservation Concept

- Possible
  - Done in hard real-time applications
  - Done in some multimedia applications – pin multimedia applications
- Be careful when you pin only the application since if other services are paged, the application will wait anyway
- Possible architecture
  - Memory broker and memory controller
  - Memory broker pins a certain size of memory – global reserve – shared among RT processes

Reservation Concept

- *Admitted storage* – memory size from the global reserve that has been already allocated to accepted processes
- *Available storage* – free memory from global reserve that is available for new RT processes
- *GlobalReserve:= AdmittedStorage + AvailableStorage;*
- Admission control:
  - test: RequestedStorage ≤ AvailableStorage
    - If yes, then
      - AdmittedStorage:= AdmittedStorage + RequestedStorage
      - AvailableStorage: = AvailableStorage – RequestedStorage
    - Else reject request for RequestedStorage

Reservation Protocol

- Step 1: SRT (Soft Real-Time) Process declares '*RequestedStorage*' to Memory Broker
- Step 2: Admission Control performed in broker
- Step 3: If Admission positive, broker builds memory reserve of size *'RequestedStorage',* allocated to SRT process during time critical run
- Step 4: Memory Broker
  - provides reservation ID to SRT process,
  - creates entry in reservation table, and
  - informs memory controller

Reservation Protocol

- Step 5: SRT process uses its reservation ID during execution to get memory allocated from memory controller
- Step 6. Memory controller
  - checks the validity of the reservation, and if positive,
  - allocates shared memory to SRT from global reserve pool;
  - gives key of shared segment to application which then attaches the shared memory segment to its own address space

Reservation Protocol

- Step 7: After SRT finishes, it
  - Frees shared memory segment from its application space,
  - Sends an event to memory controller with key to shared memory
- Step 8: Memory controller
  - Frees shared memory
  - Increases available storage information for broker
    - AvailableStorage: = AvailableStorage + FreedStorage

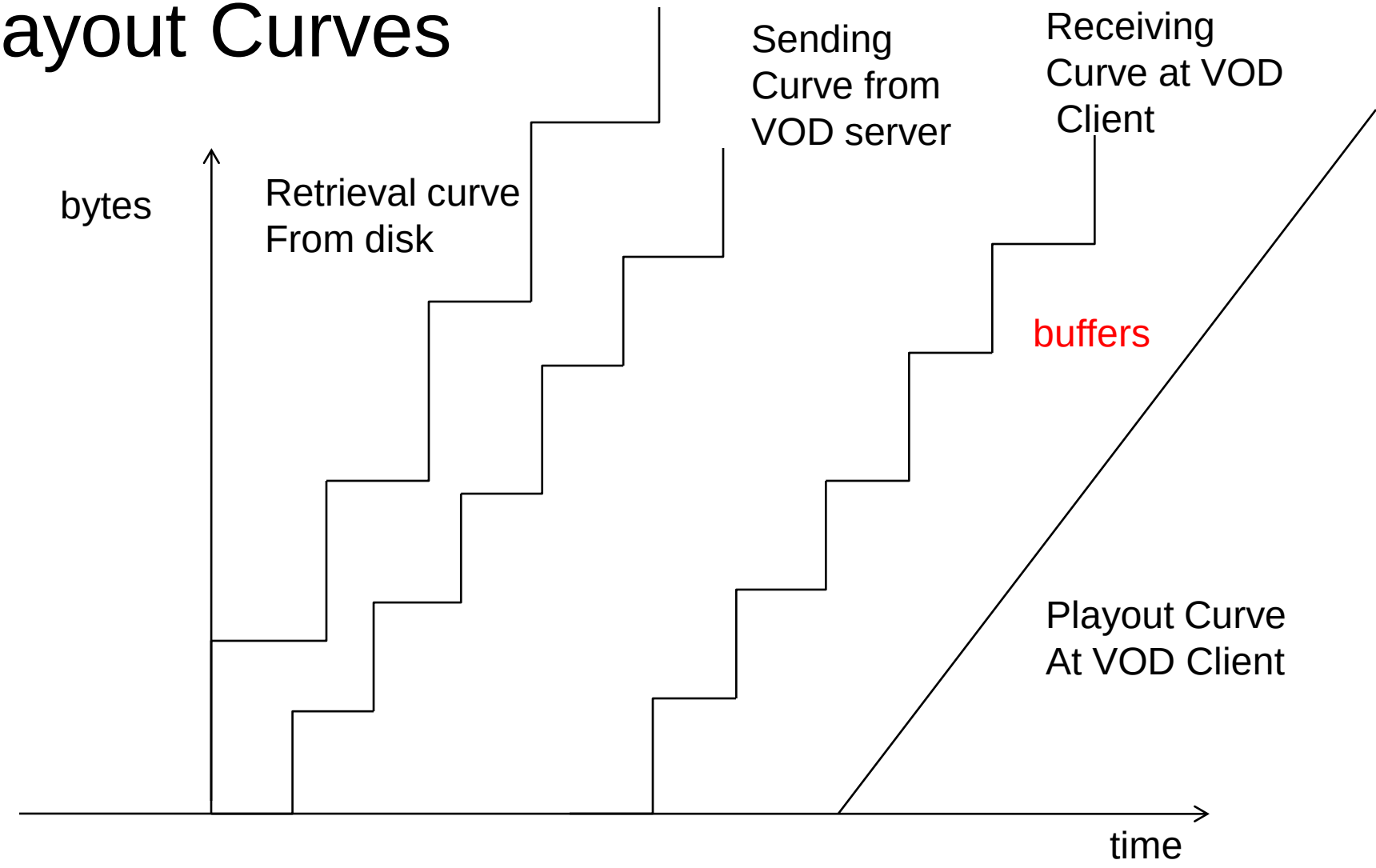
Servers Classification

- 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 (Remote Procedure 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



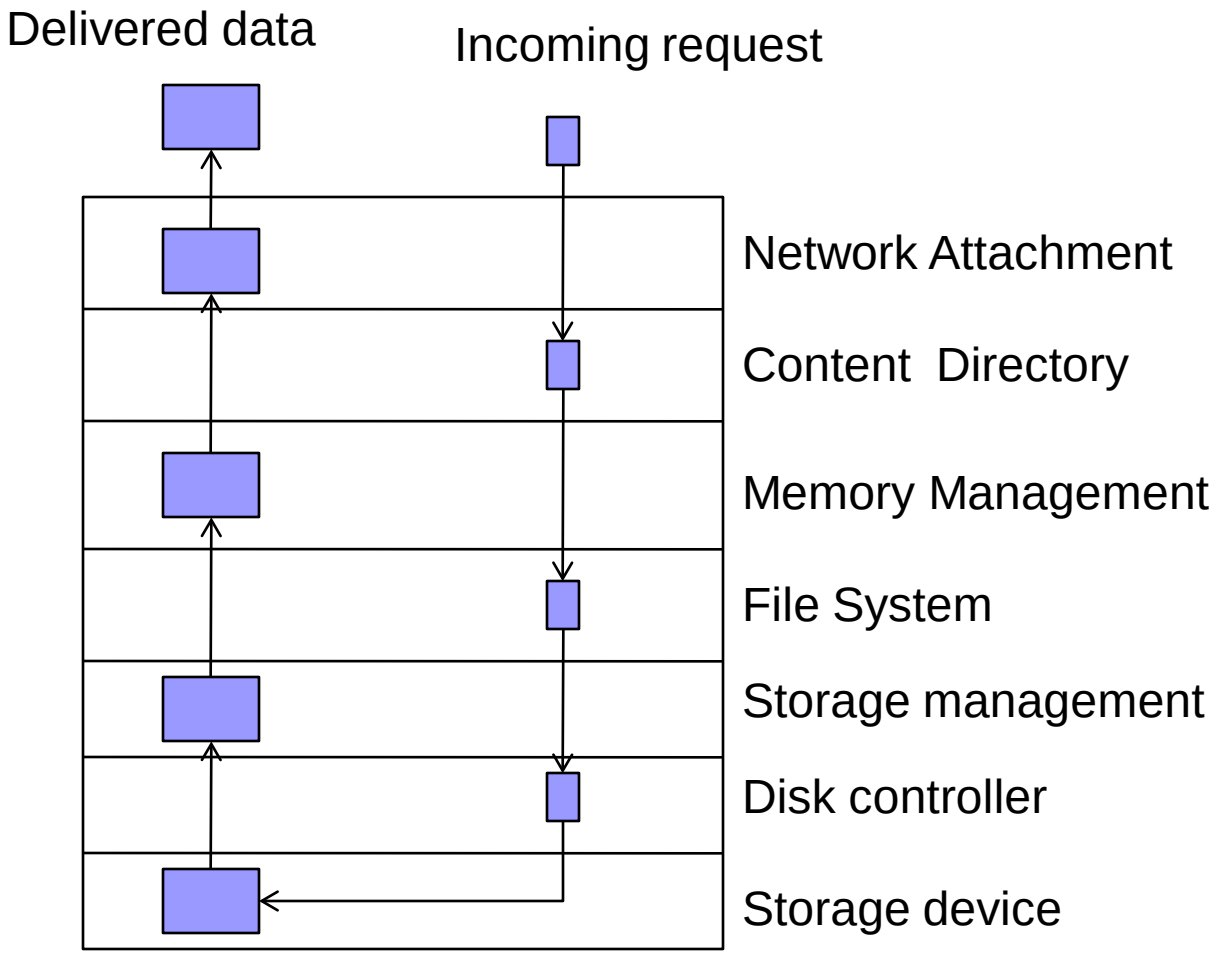
Playback

- 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

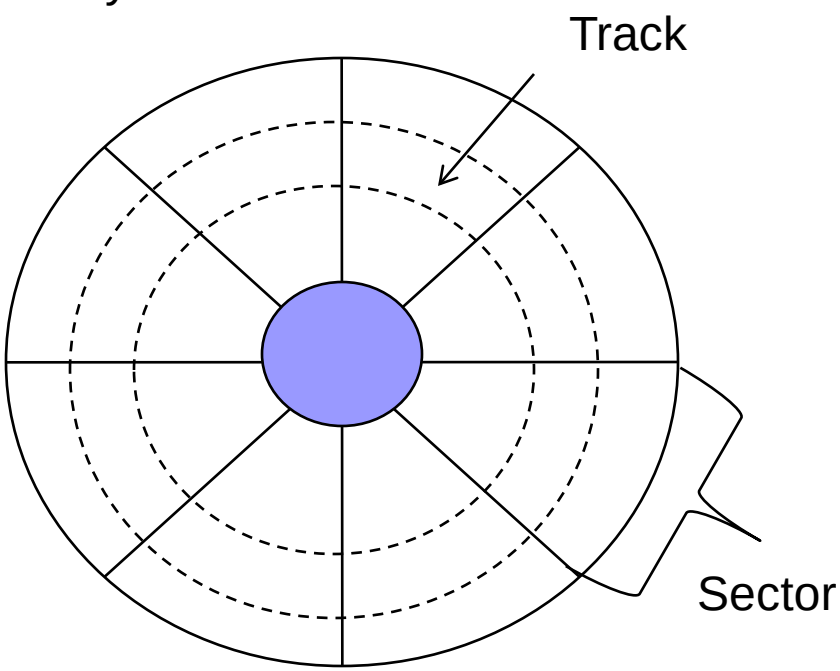
- Proper management of multimedia disk storage
  - Optimal placement of data blocks on disk
  - Usage of multiple disks
  - Role of tertiary storage
- Admission control
- Special disk scheduling algorithms and sufficient buffers to avoid jitter

Media Server Architecture



Disk Layout

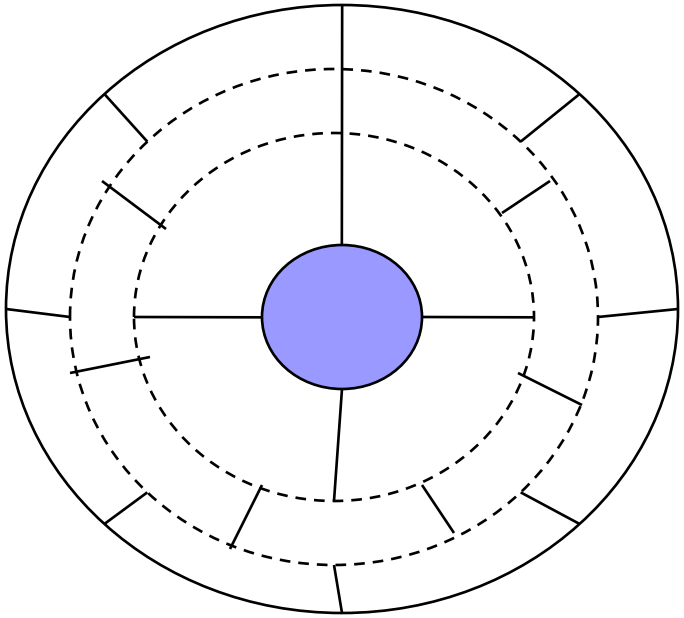
Traditional Random Access Disk Layout



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

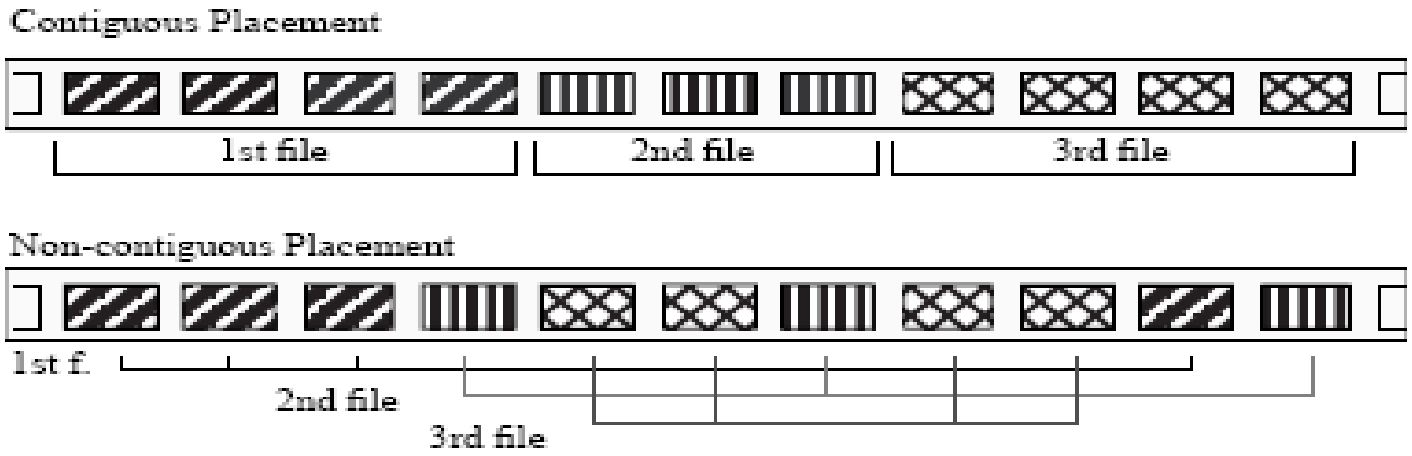
ZBR Disk Design

- Stores more sectors per track on outer tracks than on inner tracks
- Is called also Zone Constant Angular Velocity (ZCAV)
- **Place most popular movies in disk outer zones** and aggregate “hot” movies together to take advantage of high bandwidth in outer zones
- ZBR’s drive controller varies rate at which it reads and writes – faster on outer tracks
- Products that use ZBR
  - HD DVD-RW
  - Most hard drives since 1990s

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

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