Multimedia Storage Servers

Cyrus Shahabi

shahabi@usc.edu

Computer Science Department

University of Southern California

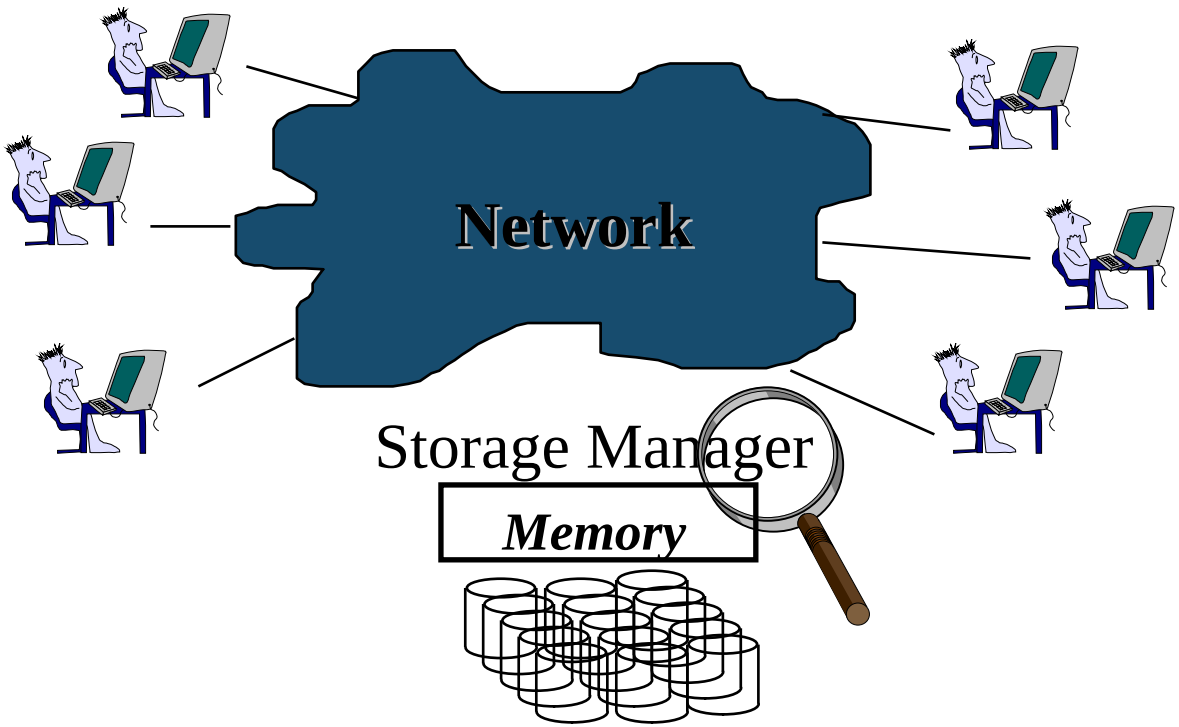
Los Angeles CA, 90089-0781

http://infolab.usc.edu

OUTLINE

- Introduction
- Continuous Media
- Magnetic Disk Drives
- Display of CM (single disk, multi-disks)
- Optimization Techniques
- Additional Issues
- Case Study (Yima)

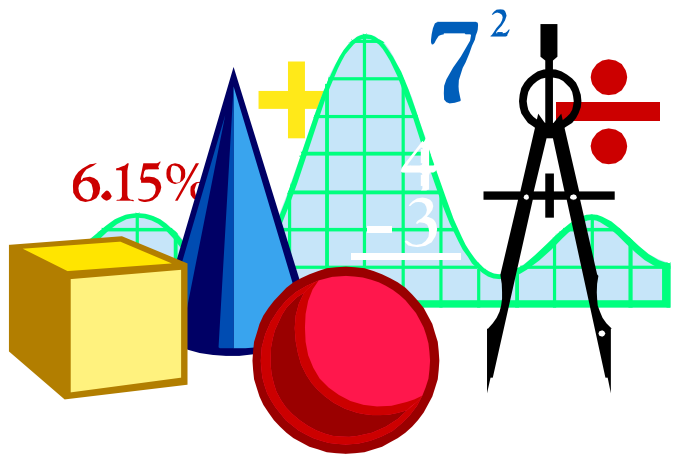
What is a Multimedia Server?



- Multiple streams of audio and video should be delivered to many users simultaneously

Some Applications

- Video-on-demand
- News-on-demand
- News-editing
- Movie-editing
- Interactive TV
- Digital libraries
- Distance Learning
- Medical databases
- NASA databases

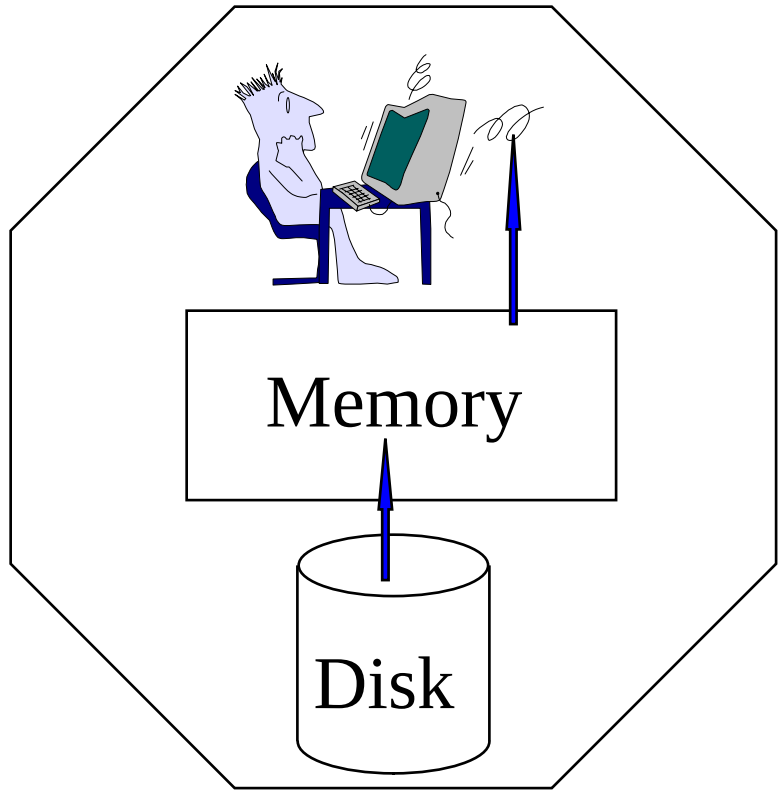


Challenge: Continuous Media

- CM object consists of a sequence of media quanta (e.g., audio samples or video frames), which convey meaning only when presented in time.
- Storage & Retrieval
  - Continuous display
  - High bandwidth requirement
  - Large size
- Communications
- End-user (display and interface)

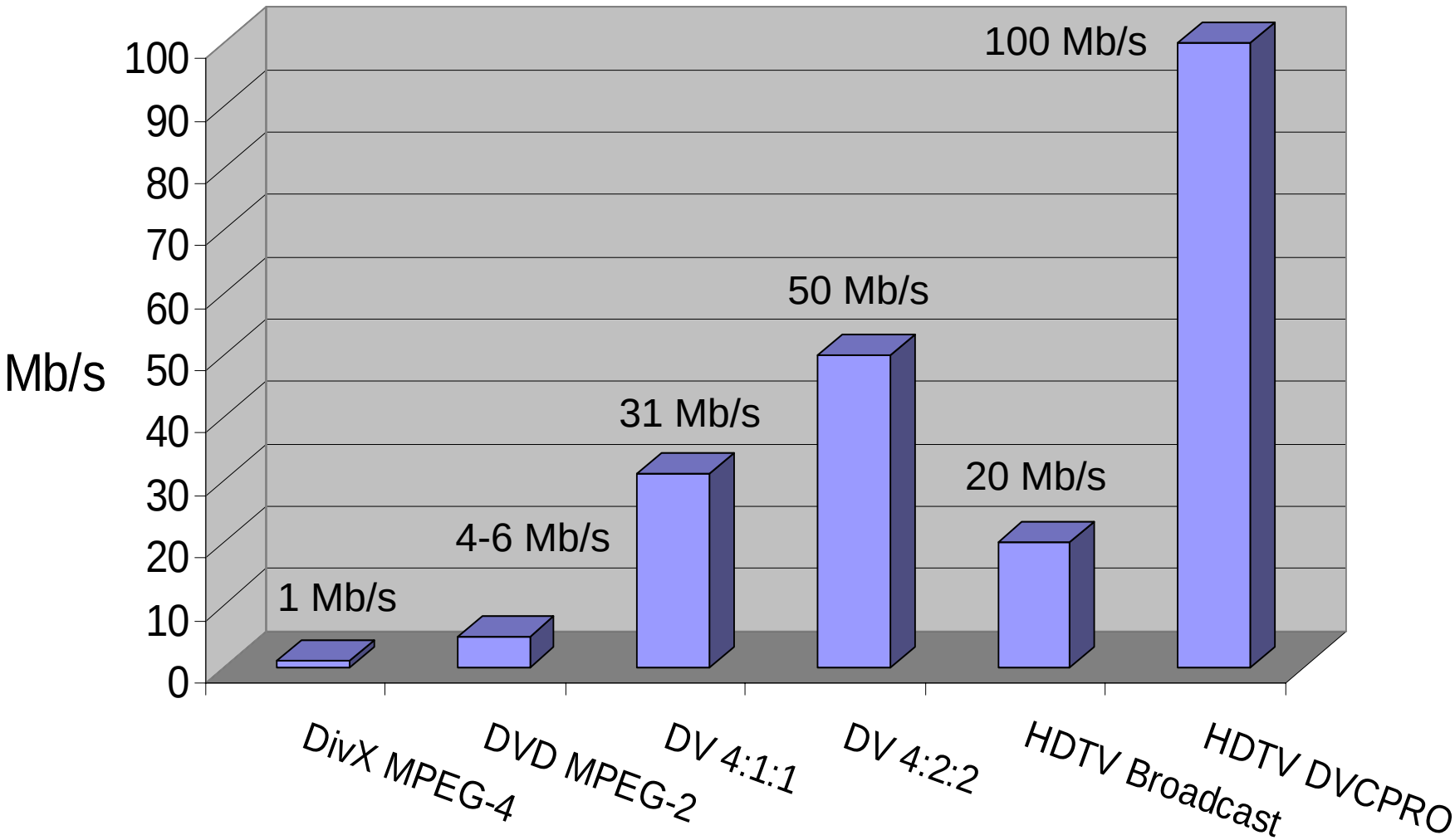
Challenge: Continuous Display

- Data should be transferred from the storage device to the memory (or display) at a pre-specified rate.
- Otherwise: frequent disruptions & delays, termed *hiccups*
- NTSC quality: 45Mb/s



Challenge: High Bandwidth

- Bandwidth requirements for different media types:



High Bandwidth & Large Size

	Access Time	Transfer Rate	Cost / Megabyte
Memory	1 ~ 5 ns	> 1 GB/s	~ $0.1
Disk	5 ~ 20 ms	< 40 MB/s	< $0.005
Optical	100 ~ 300 ms	< 5 MB/s	< $0.002
Tape	sec ~ min	< 10 MB/s	< $0.001

- HDTV quality ~ 1.4 Gb/s
  Uncompressed!
  Standard: SMPTE 292M
- 2-hr HDTV ~ 1260 GB

Streaming Media Servers

- Streaming media servers require a different "engine" than traditional databases because of:
  - Real-time retrieval and storage
  - Large media objects
- The performance metrics for streaming media servers are:
  - The number of simultaneous displays: *throughput N*
  - The amount of time that elapses until a display starts: *startup latency L*
  - The overall cost of the system: *cost per stream, C*

Media Types

- Examples of continuous media are:
  - Audio
  - Video
  - Haptics
- Continuous media are often compressed. There are many different compression algorithms, for example:
  - Motion Picture Experts Group: MPEG-1, MPEG-2, MPEG-4
  - Joint Photographic Expert Group: Motion-JPEG
  - Digital Video: DV, MiniDV
  - Microsoft Video 9, DivX, …
  - MP3: MPEG-1 layer 3 audio
  - Above codecs are based on discrete cosine transform (DCT)

Others:

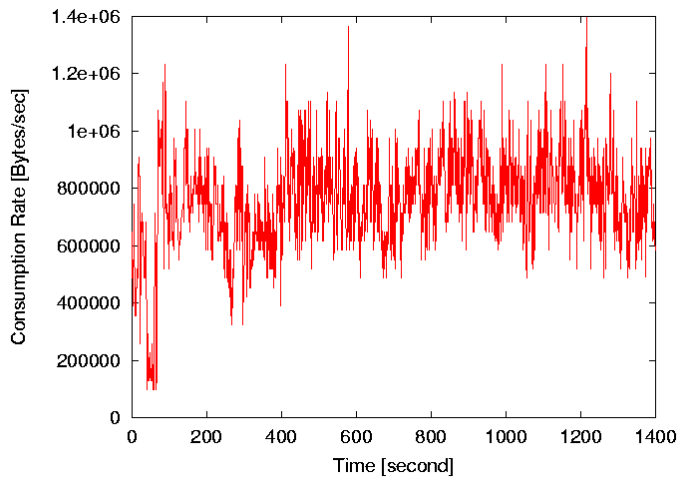
- Wavelet-based codecs
- Lossless compression

Compression

- MPEG-1 180:1 reduction in both size and bandwidth requirement (SMPTE 259M, NTSC 270 Mb/s is reduced to 1.5 Mb/s).
- MPEG-2 30:1 to 60:1 reduction.
  (NTSC ~ 4, DVD ~ 8, HDTV ~ 20 Mb/s)
- Problem: loose information
  (cannot be tolerated by some applications: medical, NASA)

Media Characteristics

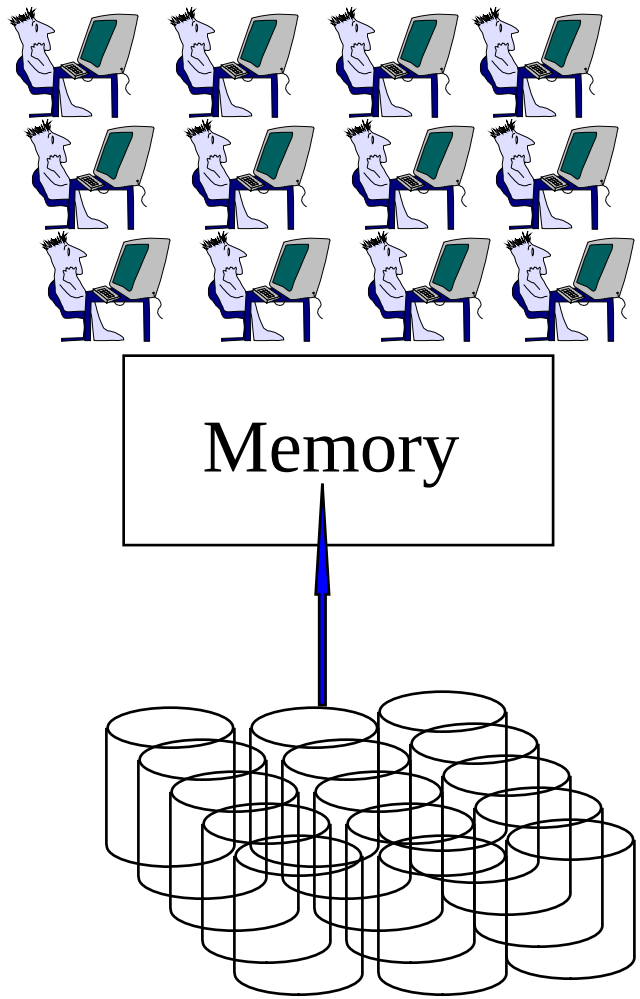
- Data requires a specific bandwidt
  - Constant bitrate (CBR) CM
  - Variable bitrate (VBR) CM



- Easier case: CBR
  - Data is partitioned into equi-sized blocks which represent a certain display time of the media
  - E.g.: 176,400 bytes represent 1 second of playtime for CD audio (44,100 samples per second, stereo, 16-bits per sample)

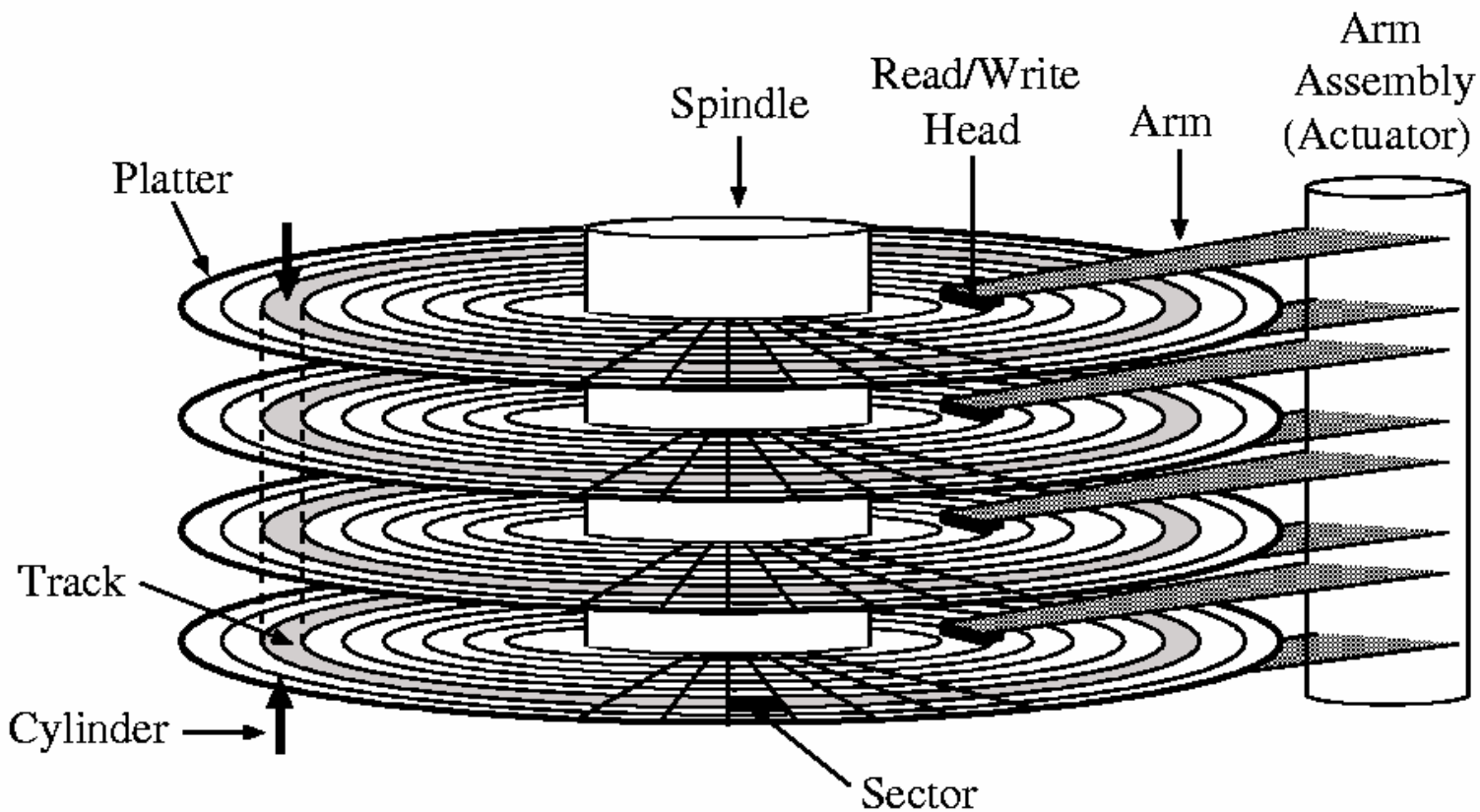
Assumed Hardware Platform

- Multiple magnetic disk drives:
  - Not too expensive (as compared to RAM)
  - Not too slow (as compared to tape)
  - Not too small (as compared to CD-ROM)
  - And it's already everywhere!



Magnetic Disk Drives

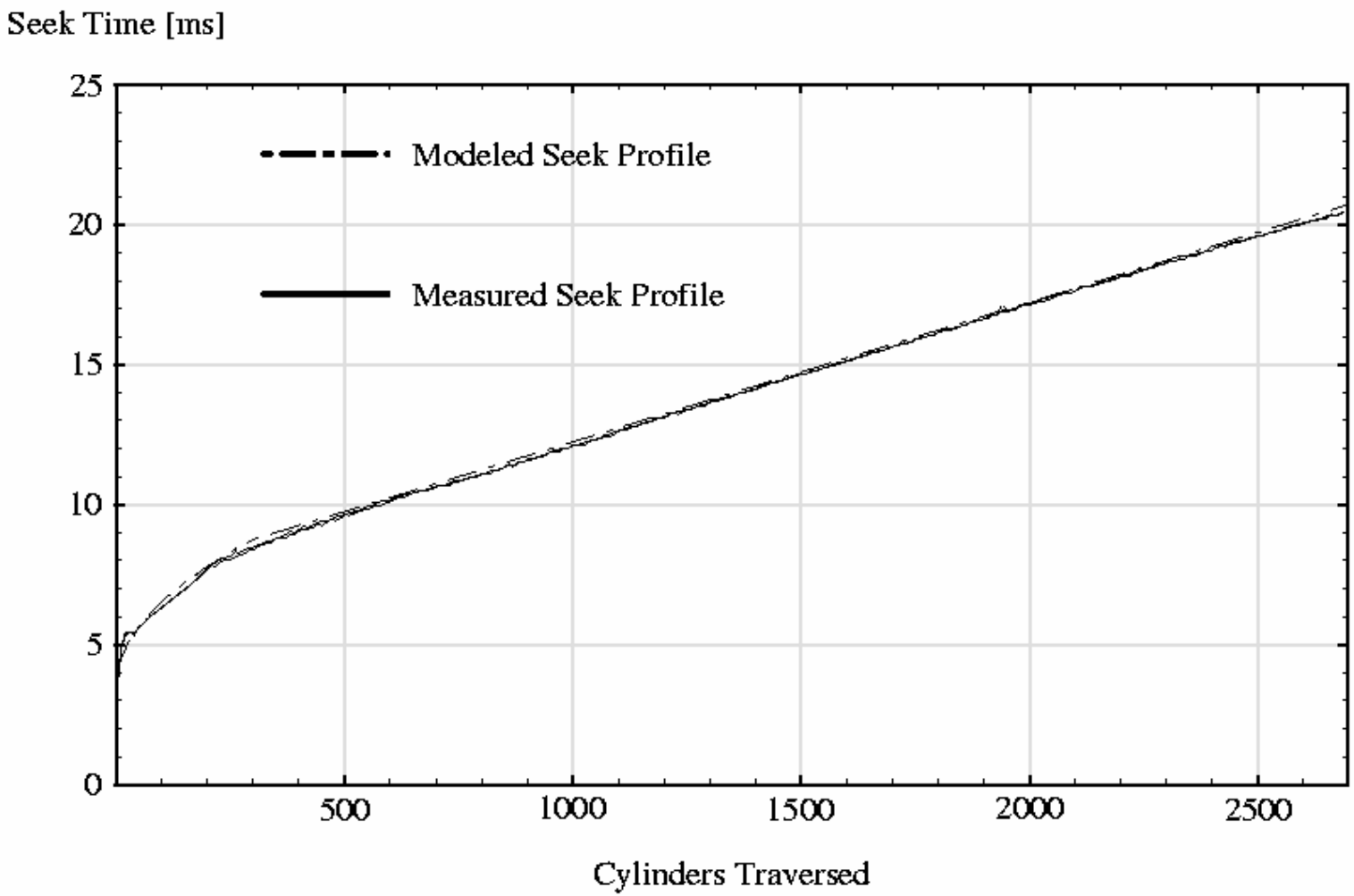
- An electro-mechanical random access storage device
- Magnetic head(s) read and write data from/to the disk



Disk Device Comparison

Model	ST31200WD	ST32171WD	ST34501WD
Series	Hawk 1LP	Barracuda 4LP	Cheetah 4LP
Manufacturer	Seagate Technology™, Inc.		
Capacity C	1.006 GB	2.061 GB	4.339 GB
Avg. transfer rate R_D	3.47 MB/s	7.96 MB/s	12.97 MB/s
Spindle speed	5,400 rpm	7,200 rpm	10,033 rpm
Avg. rotational latency	5.56 msec	4.17 msec	2.99 msec
Worst case seek time	21 msec	19 msec	16 msec
Surfaces	9	5	8
Cylinders $\#cyl$	2697	5177	6582
Number of Zones Z	23	11	7
Sector size	512 bytes	512 bytes	512 bytes
Sectors per Track ST	59 - 106	119 - 186	131 - 195
Sector ratio $\frac{ST_{z_0}}{ST_{z_{N-1}}}$	$\frac{106}{59} = 1.8$	$\frac{186}{119} = 1.56$	$\frac{195}{131} = 1.49$
Introduction year	1990	1993	1996

Disk Seek Characteristic



Disk Seek Time Model

$$T_{Seek} = \begin{cases} c_1 + (c_2 \times \sqrt{d}) \\ c_3 + (c_4 \times d) \end{cases}$$

If d < z cylinders

If d >= z cylinders

$$T_{AvgRotLatency} = \frac{1}{2} \times \frac{60\,\text{sec}}{rpm}$$

Metric	Disk Model: Hawk 1LP ST31200WD	Disk Model: Barracuda 4LP ST32171WD	Disk Model: Cheetah 4LP ST34501WD	Units
Seek constant c_1	$3.5 + 5.56^a$	$3.0 + 4.17^a$	$1.5 + 2.99^a$	msec
Seek constant c_2	0.303068	0.232702	0.155134	msec
Seek constant c_3	$7.2535 + 5.56^a$	$7.2814 + 4.17^a$	$4.2458 + 2.99^a$	msec
Seek constant c_4	0.004986	0.002364	0.001740	msec
Switch-over point z	300	600	600	cylinders
Total size *#cyl*	2697	5177	6578	cylinders

[a] Average rotational latency based on the spindle speed: 5,400 rpm, 7,200 rpm, and 10,033 rpm, respectively.

Disk Service Time Model

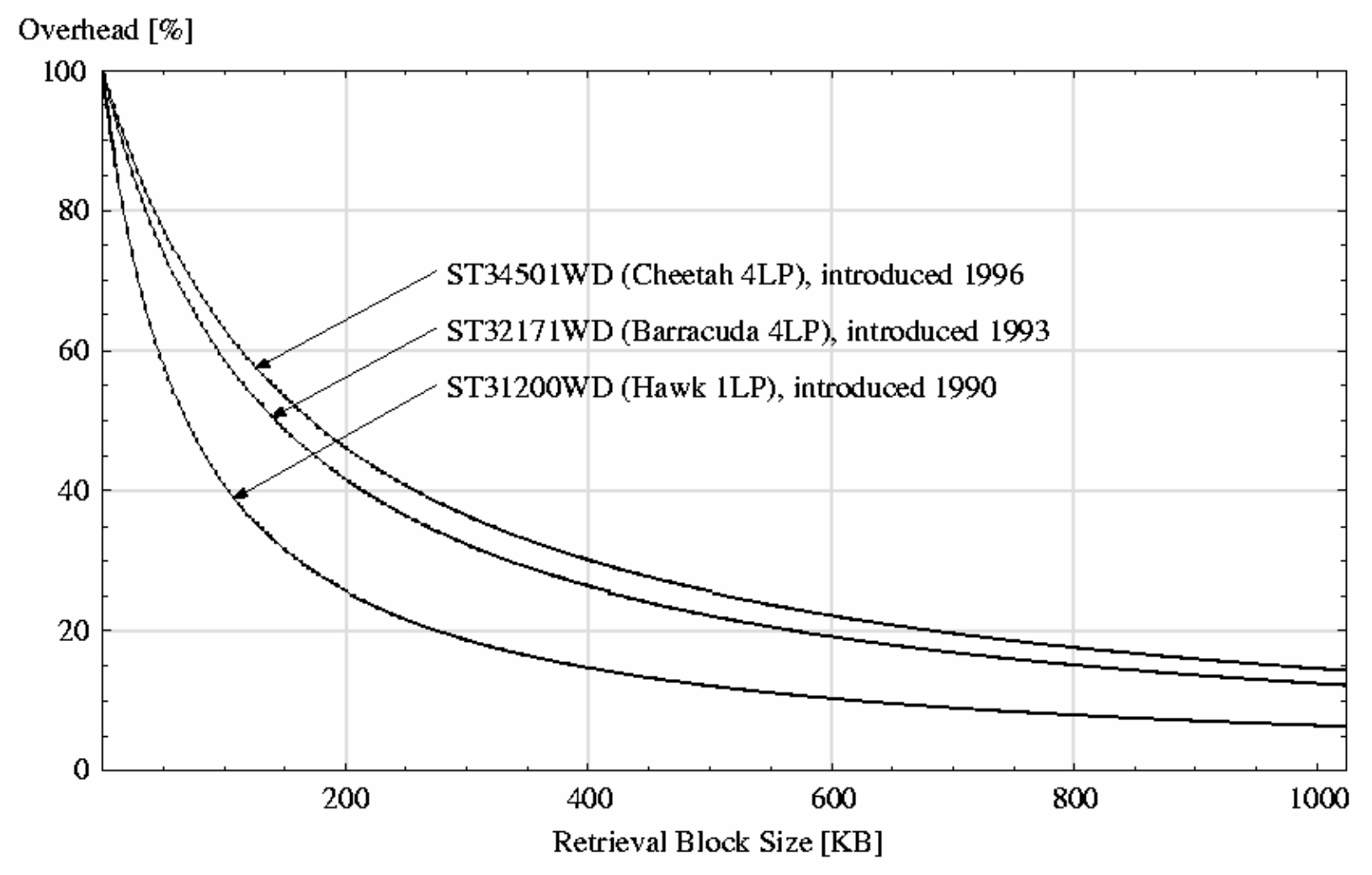
$$T_{Service} = T_{Transfer} + T_{AvgRotLatency} + T_{Seek}$$

$$BW_{Effective} = \frac{B}{T_{Service}}$$

$$T_{Transfer} = \frac{B}{BW_{Max}}$$

- $T_{Transfer}$: data transfer time [s]
- $T_{AvgRotLatency}$: average rotational latency [s]
- $T_{Service}$: service time [s]
- B: block size [MB]
- $BW_{Effective}$: effective bandwidth [MB/s]

Data Retrieval Overhead



Sample Calculations

- Assumptions:
  - T_{Seek} = 10 ms
  - BW_{Max} = 20 MB/s
  - Spindle speed: 10,000 rpm

$$BW_{Effective} = \frac{B}{\frac{B}{BW_{Max}} + \frac{30\,\text{sec}}{rpm} + T_{Seek}}$$

B	1 KB	10 KB	100 KB	1 MB	10 MB
$BW_{Effective}$	0.076 MB/s	0.74 MB/s	5.55 MB/s	15.87 MB/s	19.49 MB/s
	0.38%	3.7%	27.8%	79.4%	97.5%

Summary

- Average rotational latency depends on the spindle speed of the disk platters (rpm)
- Seek time is a non-linear function of the number of cylinders traversed
- Average rotational latency + seek time = overhead (wasteful)
- Average rotational latency and seek time reduce the maximum bandwidth of a disk drive to the effective bandwidth

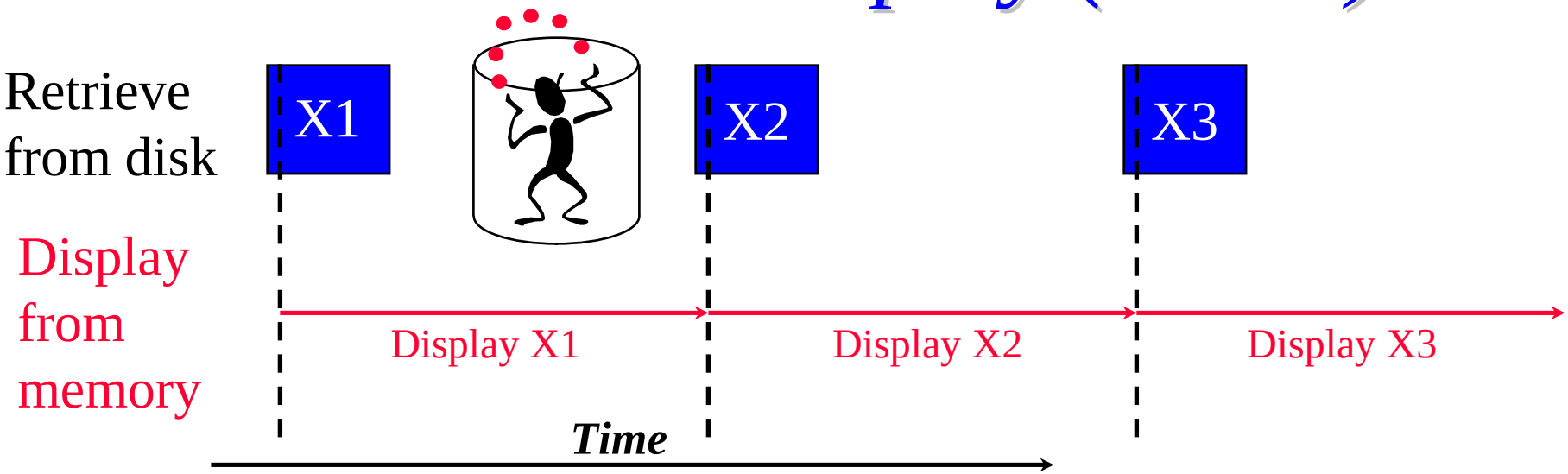
Streaming Concepts

- What is streaming?
  - Data resides on a remote server.
  - When a user initiates playback, the data is transmitted over the network and displayed on the user's machine (client).
    - *Store-and-display*: data is completely downloaded before playback starts.
    - *Streaming*: data is processed at the client side as soon as it is received.
- Advantages of streaming:
  - No waiting for downloads (not much, anyway)
  - No physical copies of the content (avoid copyright violations)
  - No storage requirements at the client side
  - Support of live events

Streaming Concepts (2)

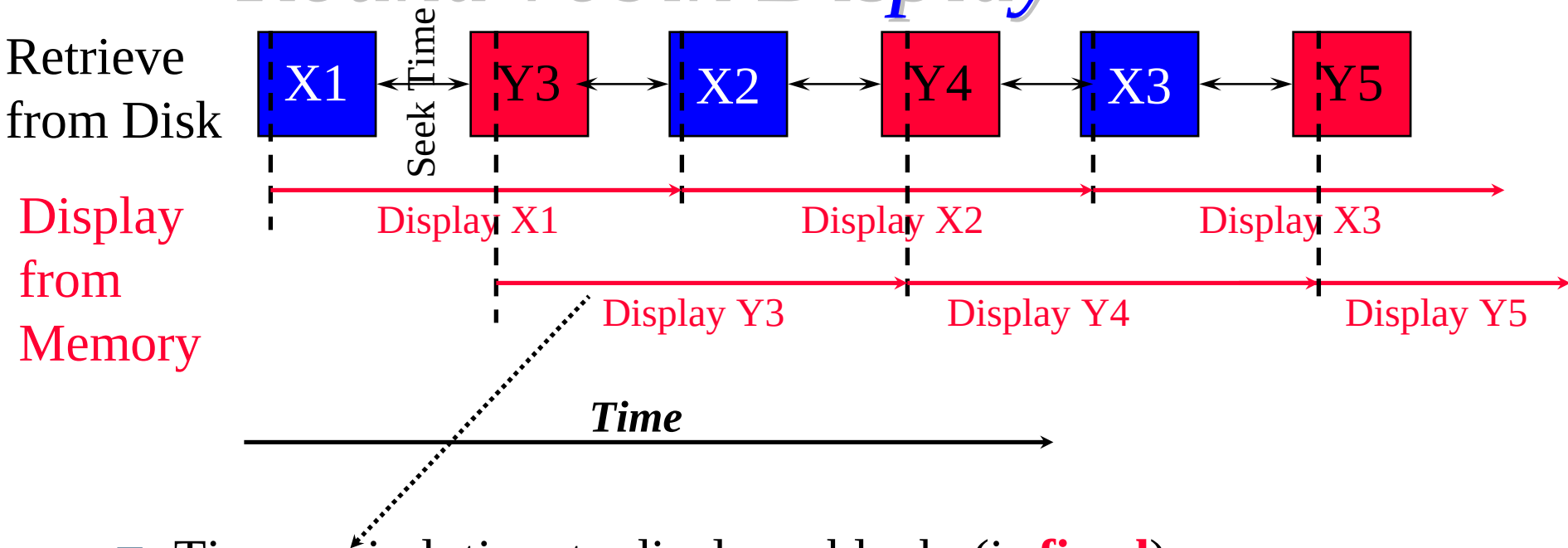
- Disadvantages of streaming:
  - It requires real-time guarantees from the network and the server
  - Lost or damaged packets (blocks) or missed deadlines may cause hiccups in the display

Continuous Display (1 disk)



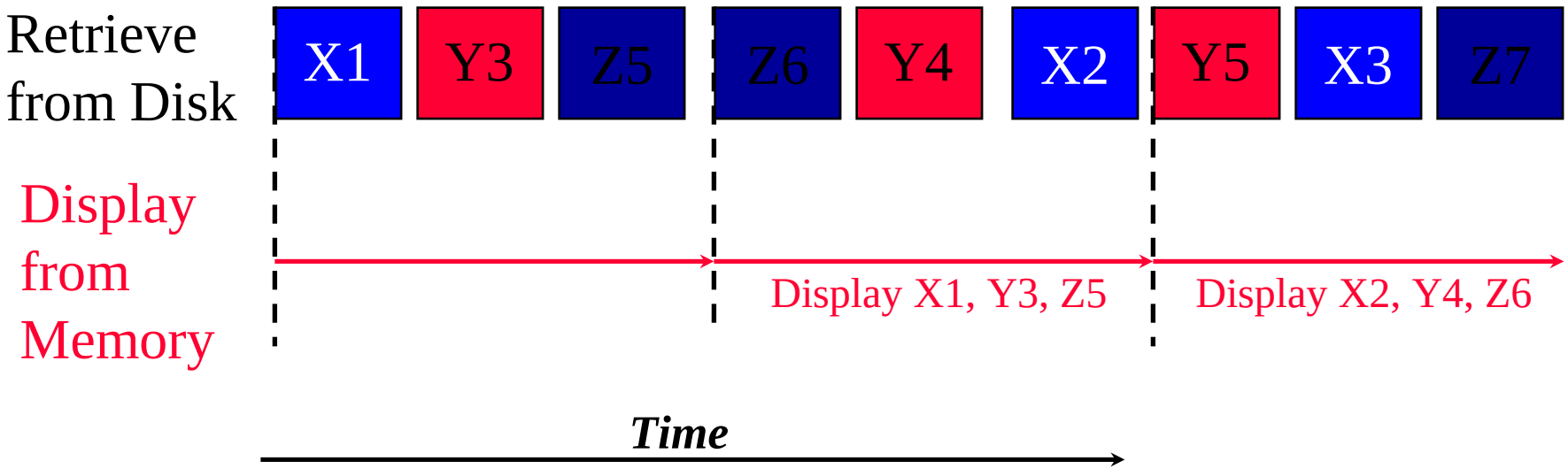
- Traditional production/consumption problem
- RC = Consumption Rate, MPEG-1 1.5 Mb/s
- RD = Production Rate,Seagate Barracuda 68 Mb/s
- For now: RC < RD
- Partition video X into *n* blocks: *X1*, *X2*, ..., *Xn* (to reduce the buffer requirement)

Round-robin Display



- Time period: time to display a block (is **fixed**)
- System Throughput (*N*): number of streams
- Assuming random assignment of the blocks:
  - Maximum seek time between block retrievals
  - Waste of disk bandwidth ==> lower throughput
  - Tp=?, N=?, Memory=?, max-latency=?

Cycle-based Display

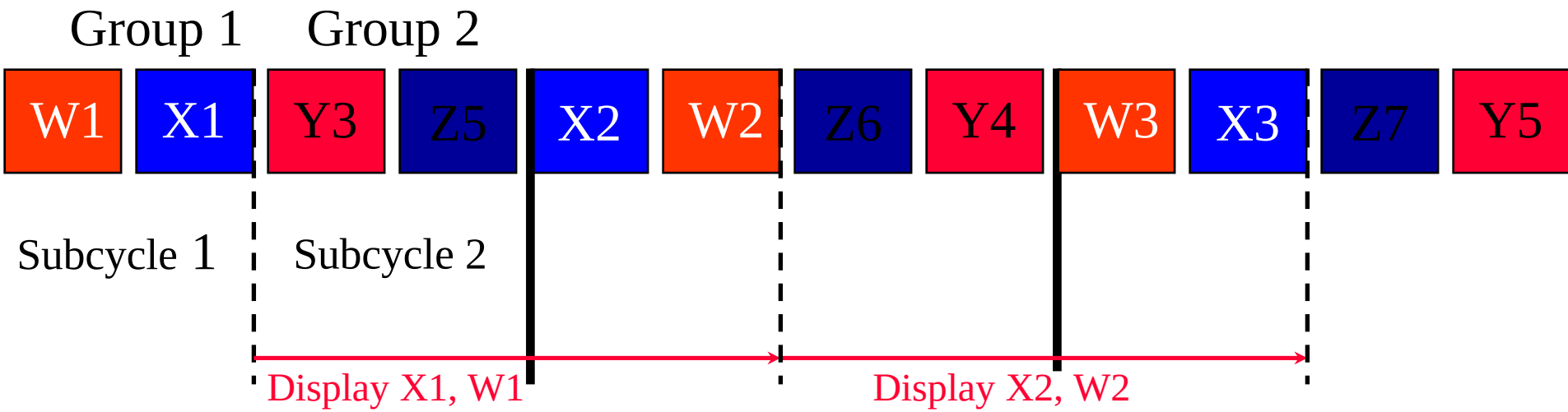


- Using disk scheduling techniques

  - Less seek time ==> Less disk bandwidth waste ==> Higher throughput

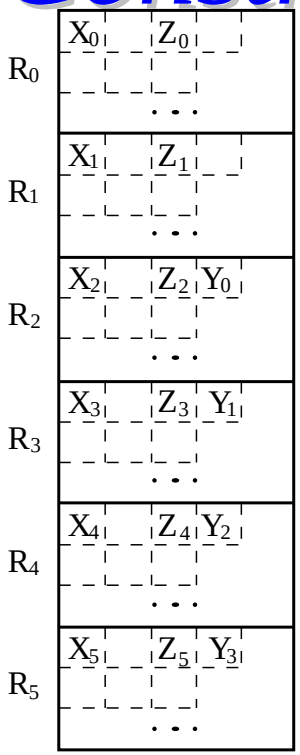
  - Larger buffer requirement
  - Tp=?, N=?, Memory=?, max-latency=?

Group Sweeping Schema (GSS)

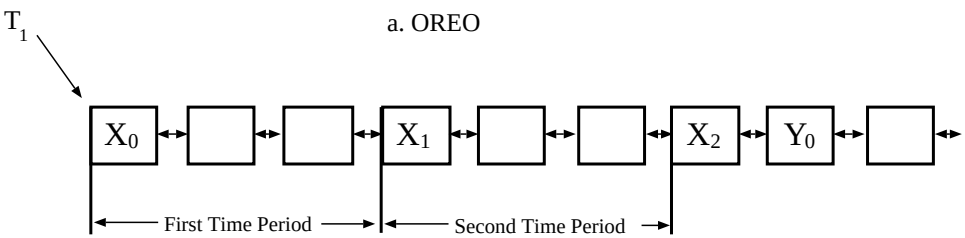


- Can shuffle order of blocks retrievals within a group
- Cannot shuffle the order of groups
- GSS when g=1 is cycle-based
- GSS when g=N is round-robin
- Optimal value of g can be determined to minimize memory buffer requirements
  - Tp=?, N=?, Memory=?, max-latency=?

Constrained Data Placement



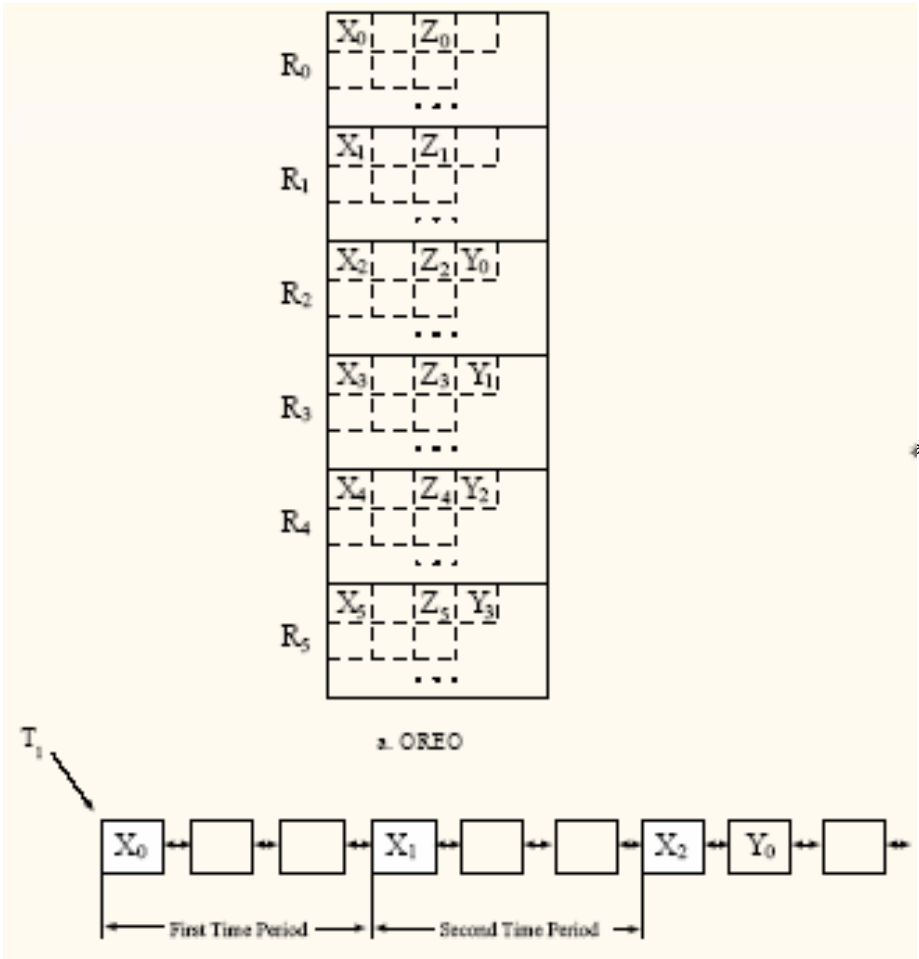
a. OREO



b. Time Period Schedule

- Partition the disk into *R* regions
- During each time period only blocks reside in the same region are retrieved
- Maximum seek time is reduced almost by a factor of *R*
- Introduce startup latency time
  - Tp=?, N=?, Memory=?, max-latency=?

Constrained Data Placement

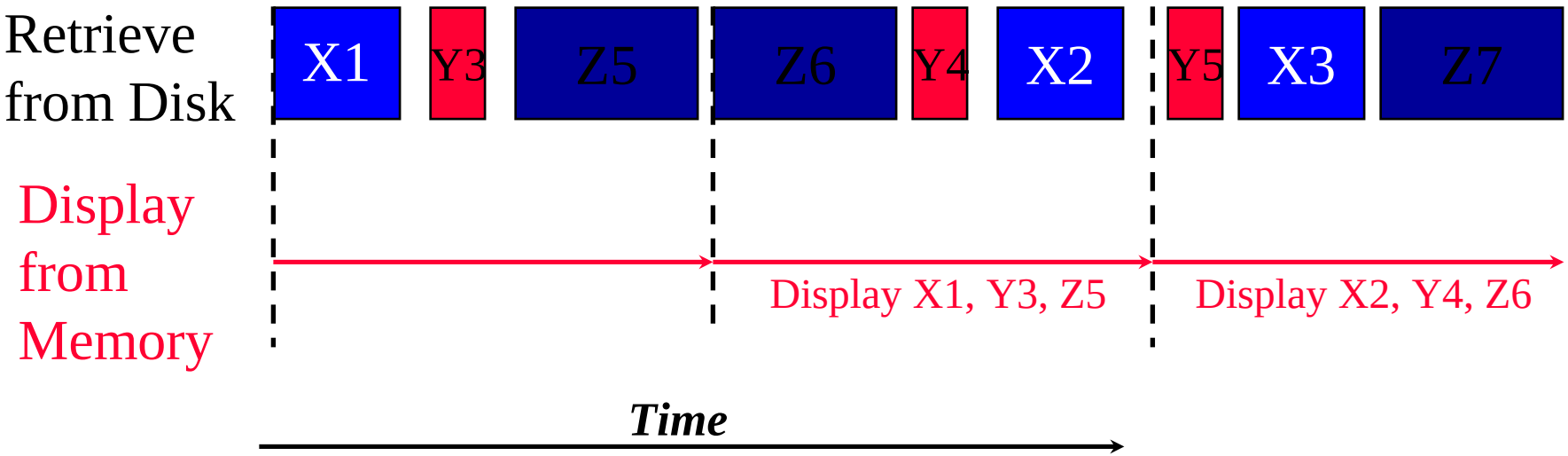


- Partition the disk into *R* regions.
- During each time period only blocks reside in the same region are retrieved.
- Maximum seek time is reduced almost by a factor of *R*.
- Introduce startup latency time
  - Tp=?, N=?, Memory=?, max-latency=?

Hybrid

- For the blocks retrieved within a region, use GSS schema
- This is the most general approach
  - Tp=?, N=?, Memory=?, max-latency=?
- By varying R and g all the possible display techniques can be achieved
- Round-robin (R=1, g=N)
- Cycle-base (R=1, g=1)
- Constrained placement (R>0, g=1), ...
- A configuration planner calculates the optimal values of R & g for certain application.

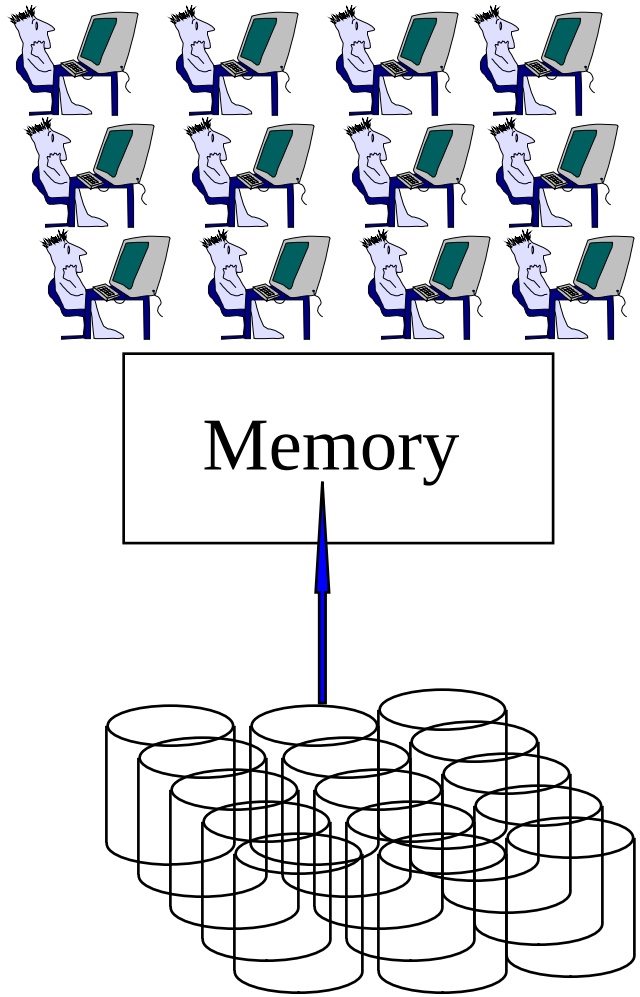
Display of Mix of Media



- Mix of media types: different Rc's: audio, video
- Rc(Y) < Rc(X) < Rc(Z)
- Different block sizes: *Rc(X)/B(X)=Rc(Y)/B(Y)= ...*
- Display time of a block (time period) is still fixed

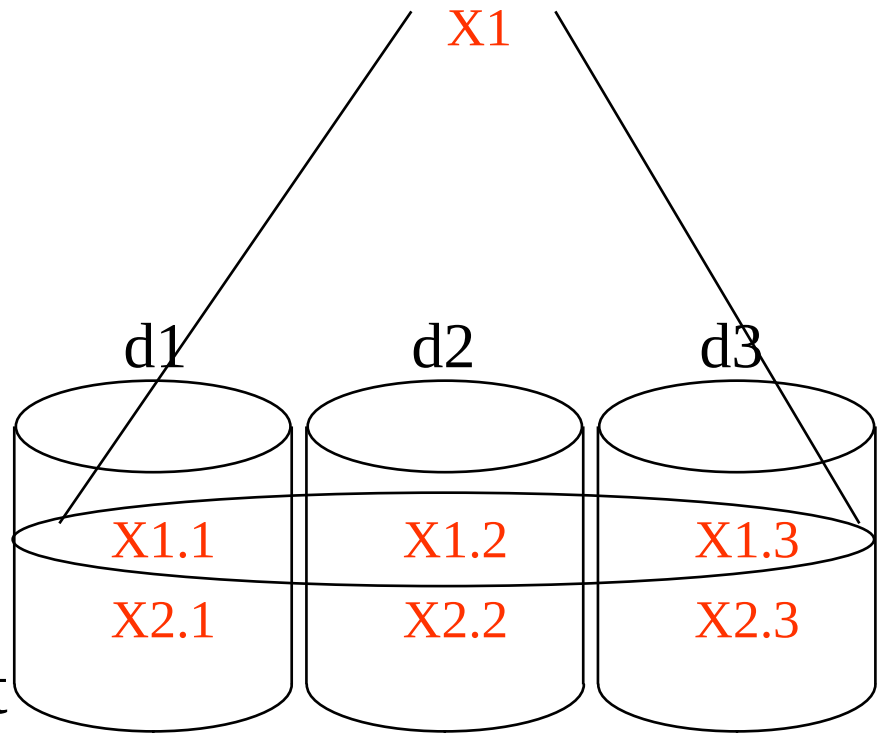
Multiple-disks

- Single disk: even in the best case with 0 seek time, 68/1.5=45 MPEG-1 streams
- Typical applications (MOD): 1000 streams
- Solution: aggregate bandwidth and storage space of multiple disk drives
- How to place a video?



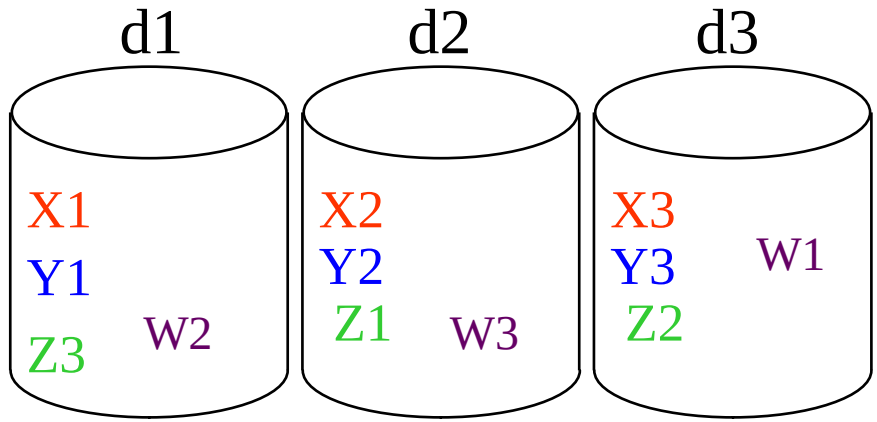
RAID Striping

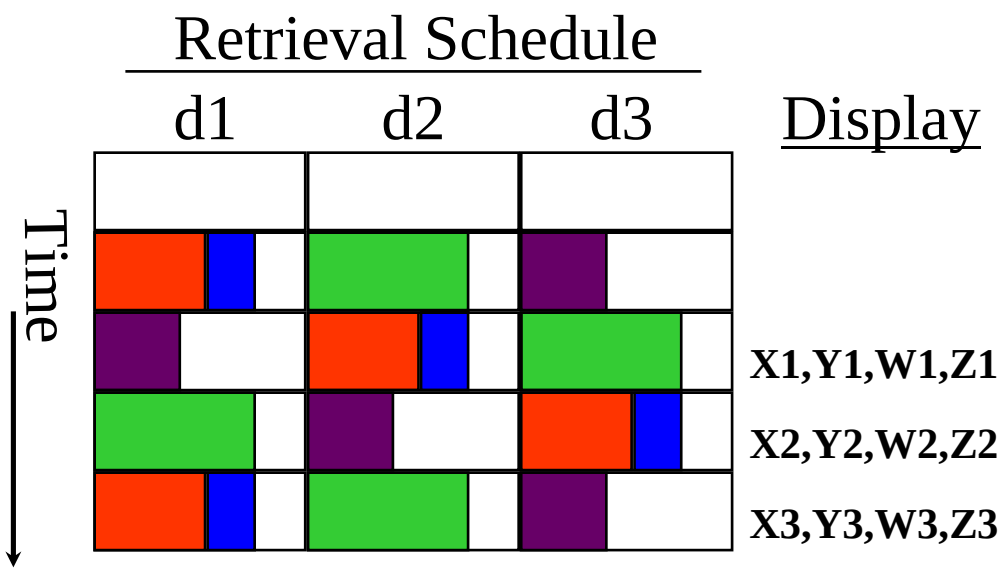
- All disks take part in transmission of a block
- Can be conceptualized as a single disk
- Even distribution of display load
- Efficient admission
- Is not scalable in throughput



Round-robin retrieval

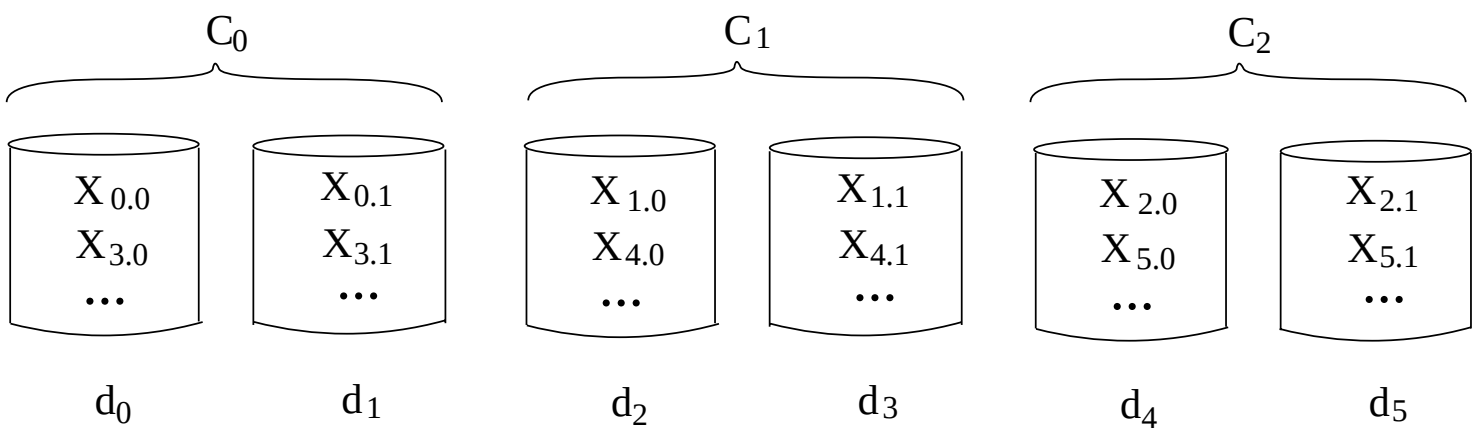
- Only a single disk takes part in transmission of each block
- Retrieval schedule
  - Round-robin retrieval of the blocks
- Even distribution of display load
- Efficient admission
- Not scalable in latency





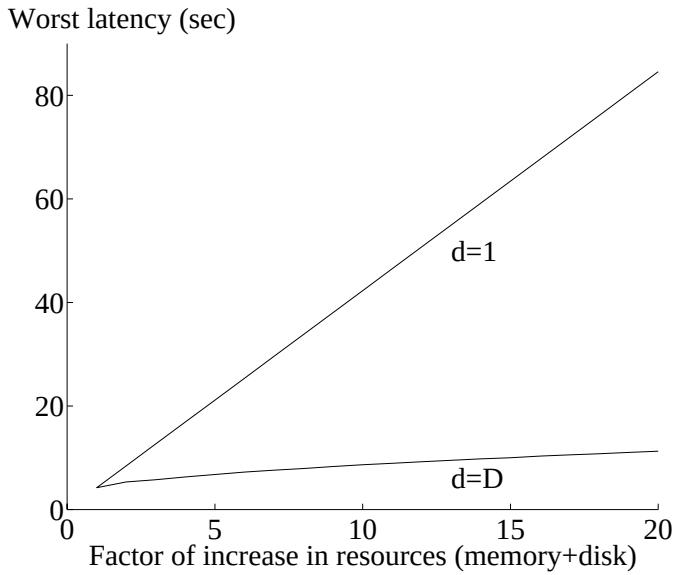
Hybrid Striping

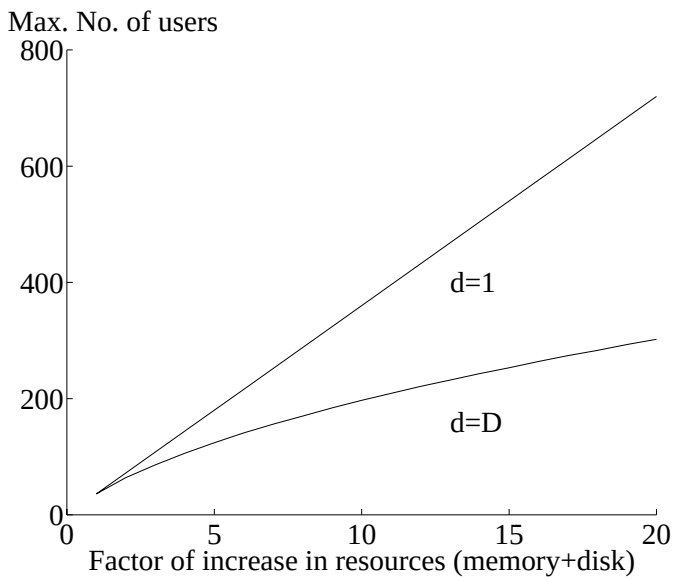
- Partition D disks into clusters of d disks
- Each block is declustered across the *d* disks that constitute a cluster (each cluster is a logical disk drive)
- RAID striping within a cluster
- Round-robin retrieval across the clusters
- RAID striping (d=D), Round-robin retrieval (d=1)



Scalability

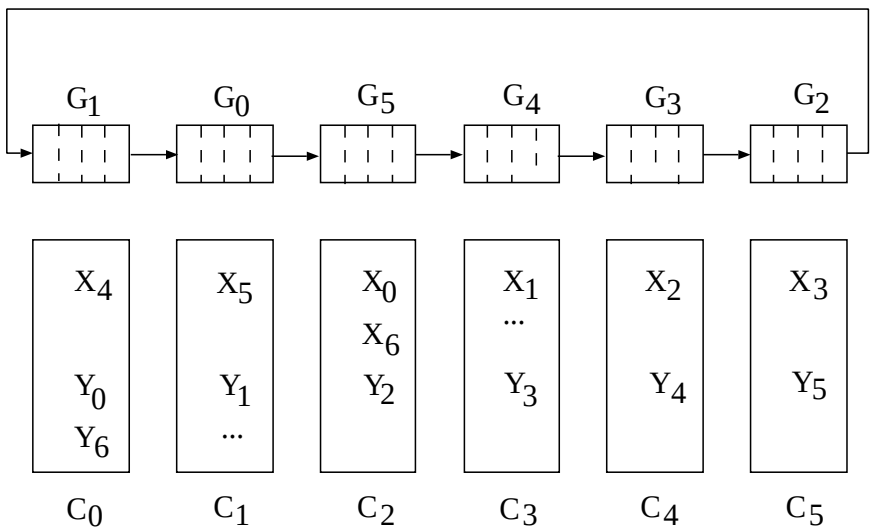
- By varying the value of d different throughput and latency time can be achieved
- For an application, the maximum throughput and the tolerable latency time can be given as inputs to a planner
- The output will then be the optimal value of d for that application
- Note that hybrid always outperforms RAID striping and round-robin retrieval

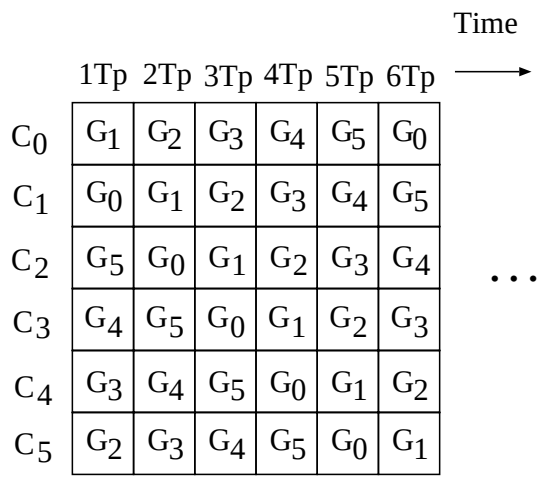




Latency & Throughput *(Hybrid Striping)*

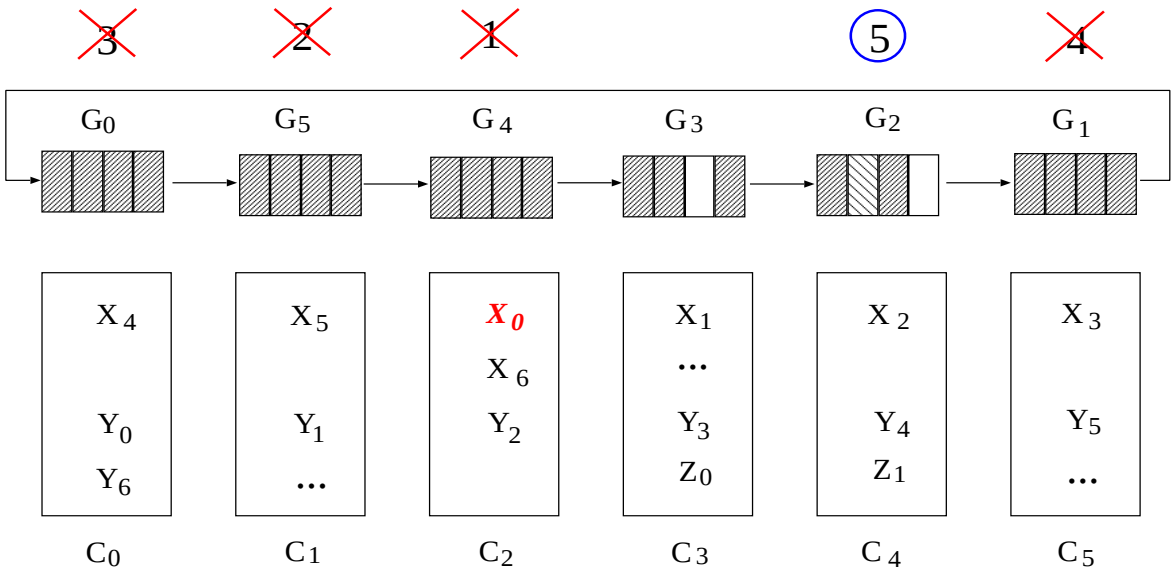
- Conceptualize a set of slots supported by a cluster in a *Tp* as a *group*
- A request maps onto one group and groups visit clusters in a round-robin manner
- During a time period, each occupied slot of a group retrieves a block that resides in the cluster that is being visited by that group





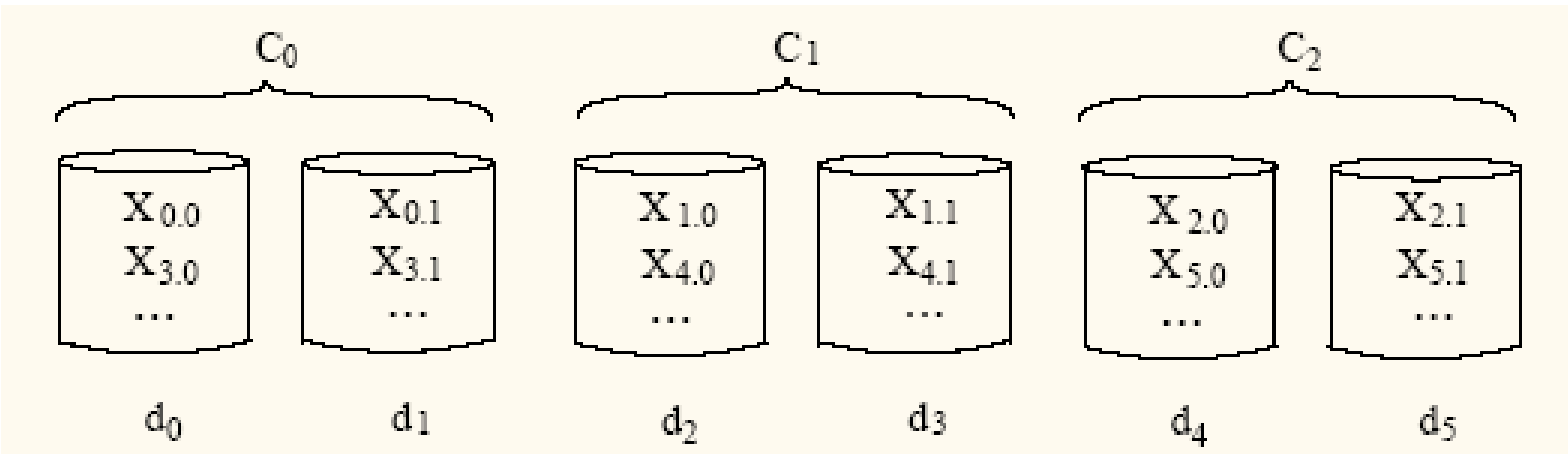
Admission Control

- When a request arrives, search an empty slot from the group currently accessing the cluster which has the first block
- *failure*: look up a group and find no empty slot
- *success*: find a group with empty slot and assign the request



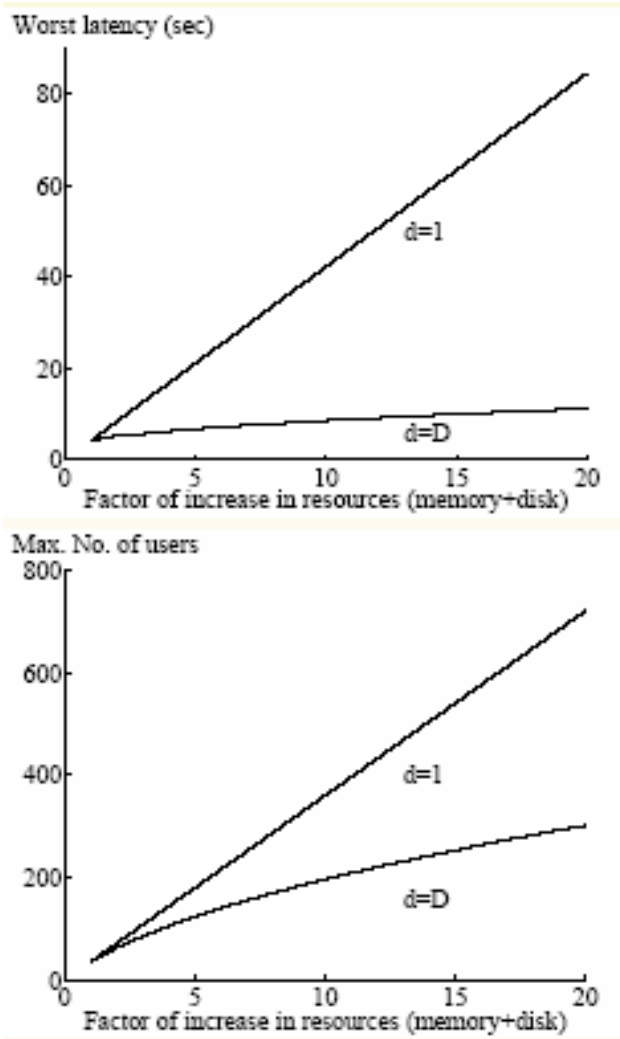
Hybrid Striping

- Partition D disks into clusters of d disks.
- Each block is declustered across the *d* disks that constitute a cluster (each cluster is a logical disk drive).
- RAID striping within a cluster.
- Round-robin retrieval across the clusters.
- RAID striping (d=D), Round-robin retrieval (d=1).



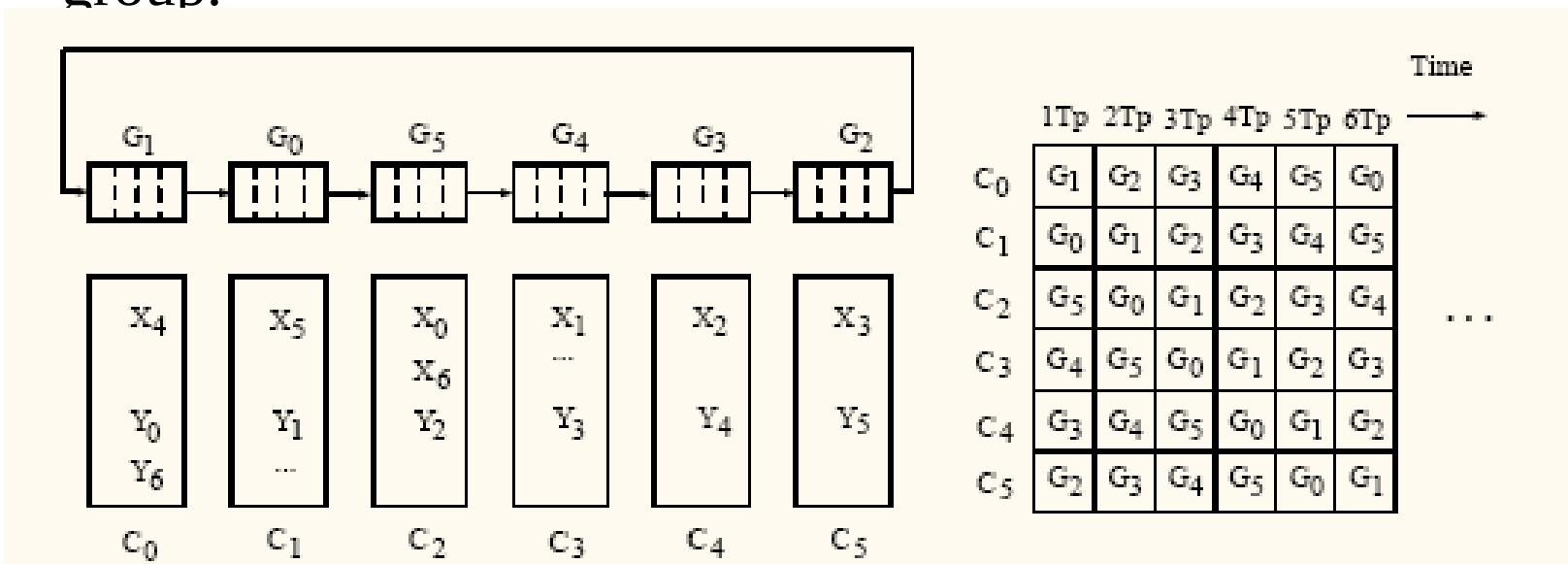
Scalability

- By varying the value of d different throughput and latency time can be achieved.
- For an application, the maximum throughput and the tolerable latency time can be given as inputs to a planner.
- The output will then be the optimal value of d for that application.
- Note that hybrid always outperforms RAID striping and round-robin retrieval.



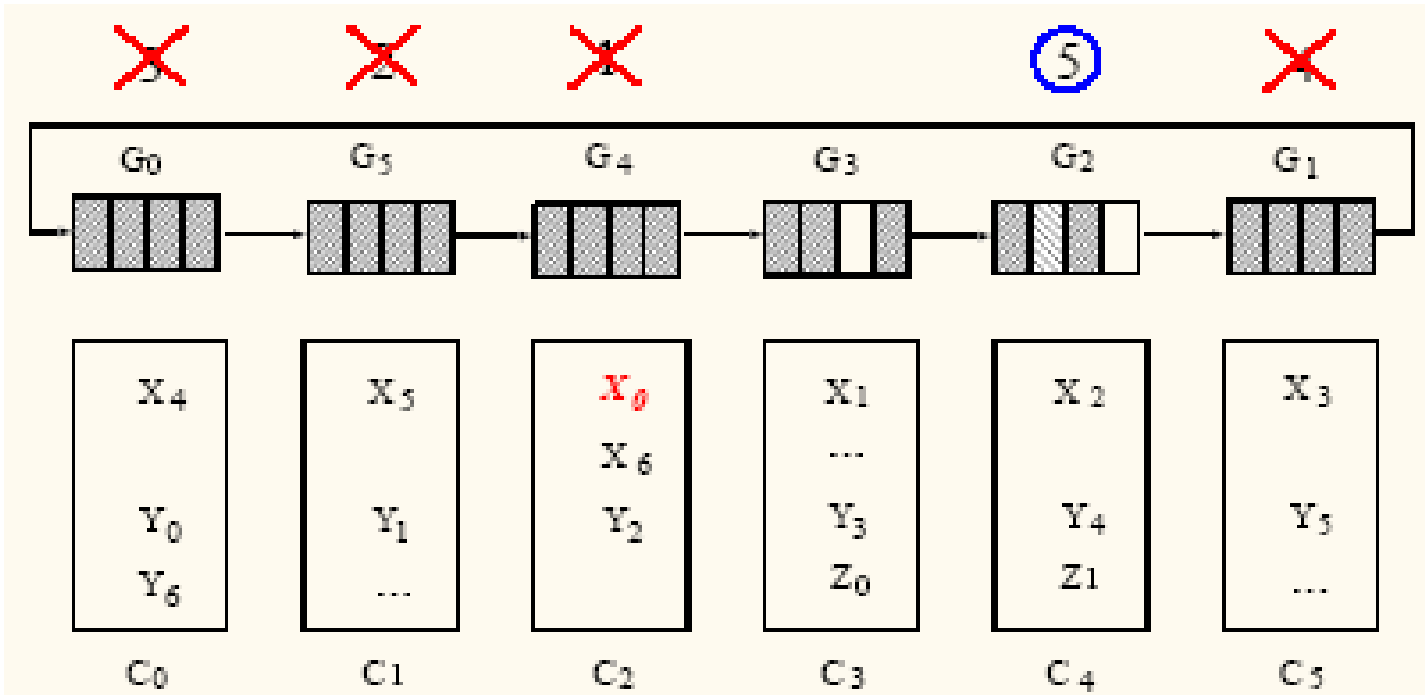
Latency & Throughput *(Hybrid Striping)*

- Conceptualize a set of slots supported by a cluster in a *Tp* as a *group*.
- A request maps onto one group and groups visit clusters in a round-robin manner.
- During a time period, each occupied slot of a group retrieves a block that resides in the cluster that is being visited by that group.



Admission Control

- When a request arrives, search an empty slot from the group currently accessing the cluster which has the first block.
- *failure*: look up a group and find no empty slot.
- *success*: find a group with empty slot and assign the request.



Throughput & Startup Latency

- The number of slots (N) in a time period defines the maximum number of simultaneous displays (*throughput*) supported by a cluster
- N and Tp could be configured using various techniques such as GSS
- The throughput of a system with C clusters is:

$$N \times C$$

- If a request experiences i failures before success, the average *startup latency* is:

$$L = \begin{cases} 0.5 \times T_P (i = 0) \\ i \times T_P (i \neq 0) \end{cases}$$

- How to determine the value of i ?

Throughput & Startup Latency

- The probability of a failure is a function of system load
- Develop a probabilistic approach to determine the expected latency of a request
- *p(k)* : probability that there are *k* active requests in the system (using a queuing model)
- $p_f(i,k)$: probability that a request has *i* failures before a success when there are *k* active requests in the system
- The expected latency is:

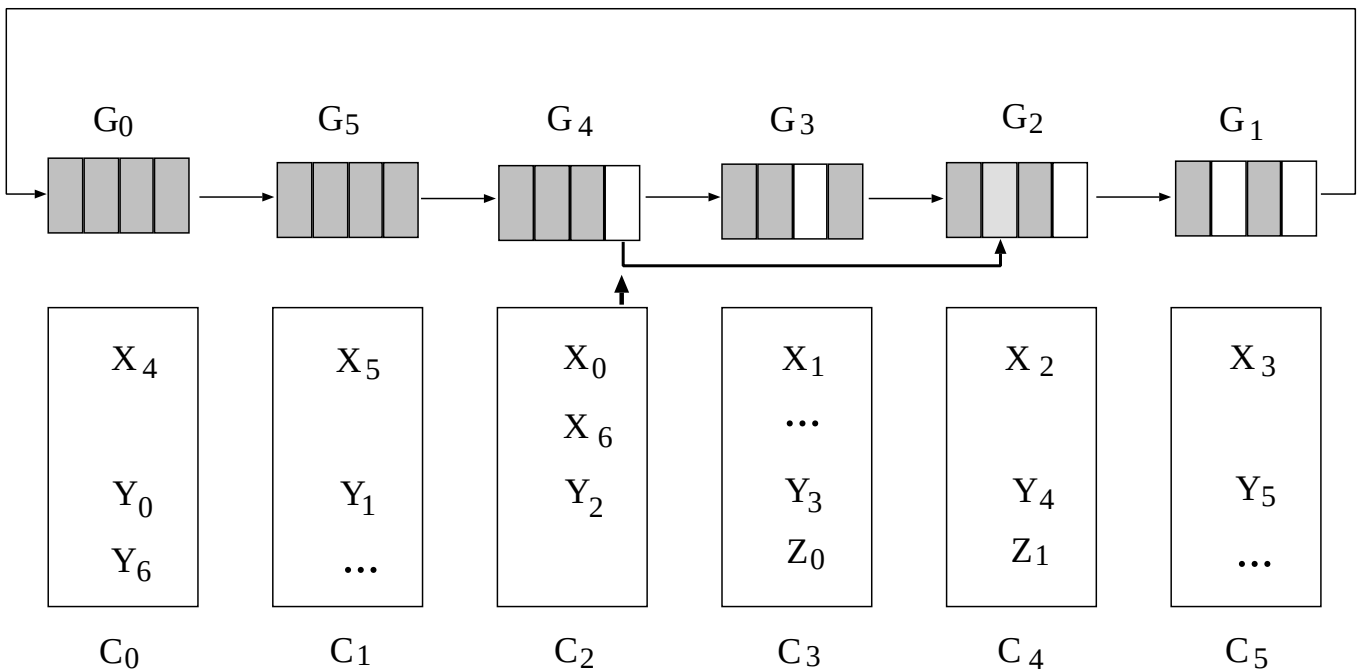
$$E[L] = \sum_{k=0}^{m-1} p(k)\, p_f(0,k) 0.5 T_p + \sum_{k=0}^{m-1} \sum_{i=1}^{\lfloor k/N \rfloor} p(k)\, p_f(i,k)\, i T_p$$

Optimization Techniques

- Two techniques to minimize startup latency
  - Request Migration
  - Object Replication
- Three techniques to maximize throughput
  - Batching requests
  - Piggybacking
  - Buffer sharing

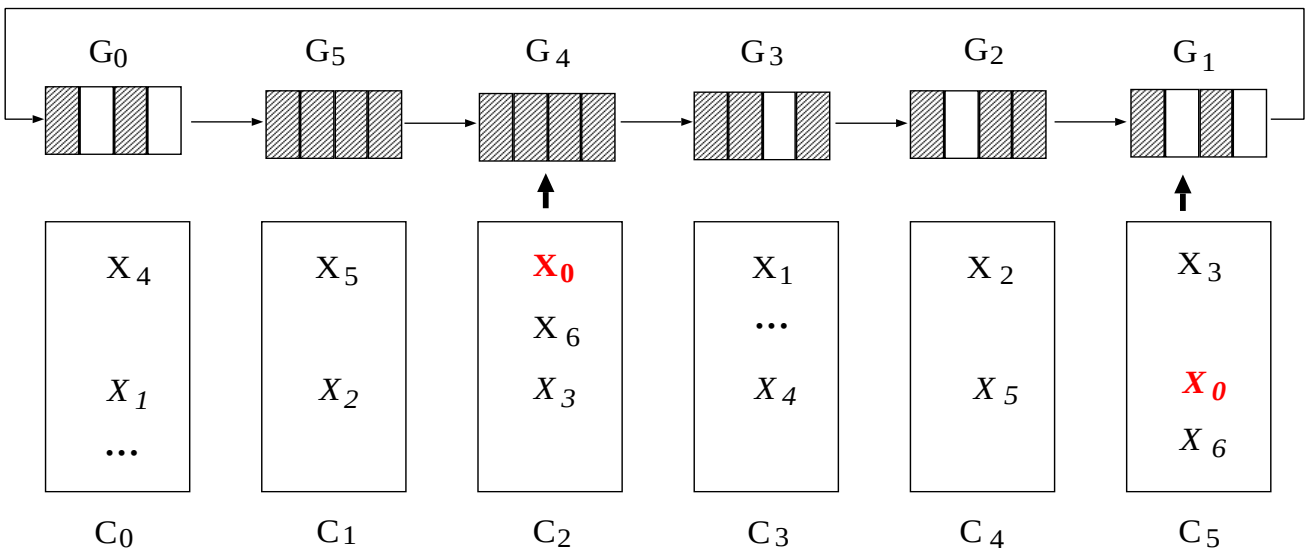
Request Migration

- Migrating a request from a busy group to a group with more idle slots reduces the possible latency of future requests



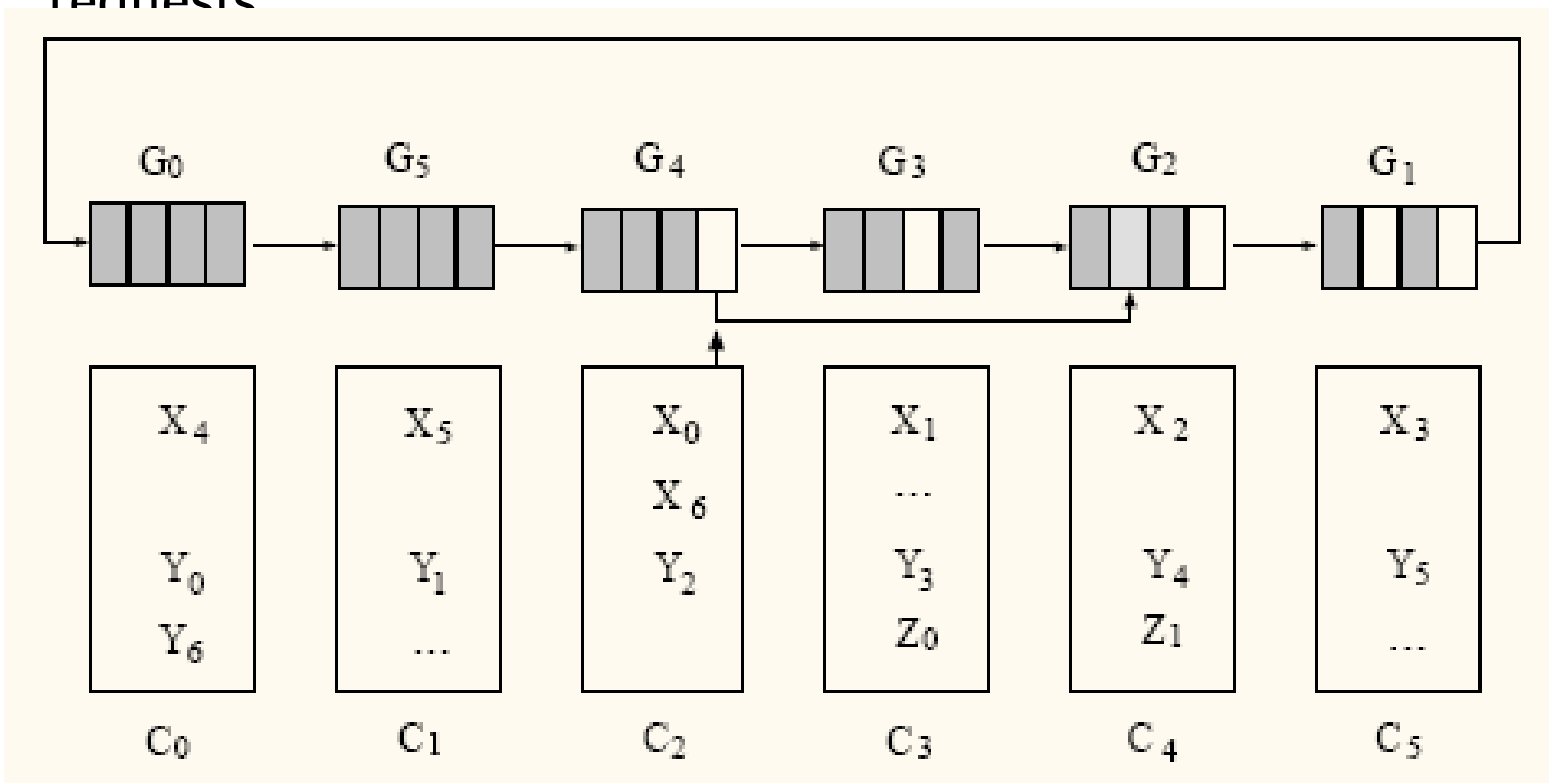
Object Replication

- With multiple copies of an object, simultaneous checking for an empty slot reduces the worst case startup latency and the average latency



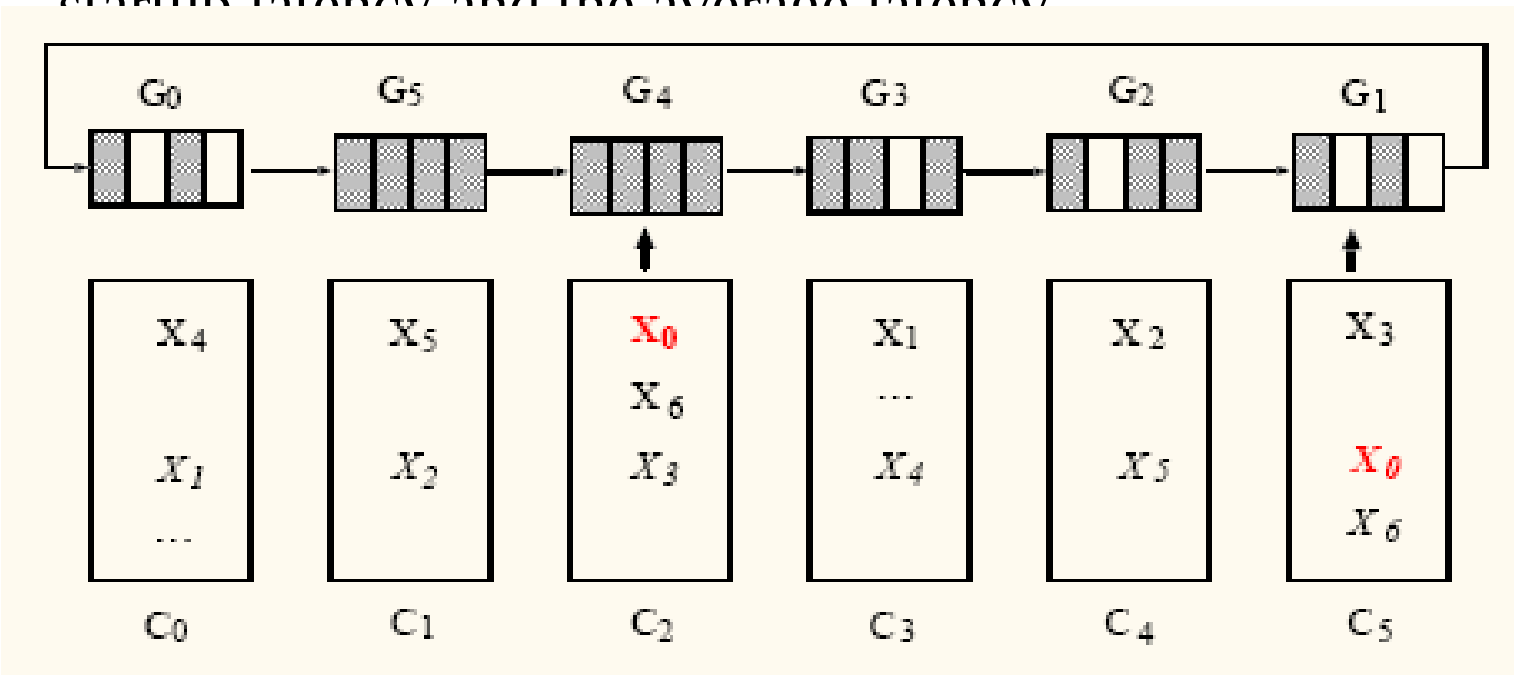
Request Migration

- Migrating a request from a busy group to a group with more idle slots reduces the possible latency of future requests



Object Replication

- With multiple copies of an object, simultaneous checking for an empty slot reduces the worst case startup latency and the average latency



Maximize Throughput

- Objective: support more requests than the maximum throughput of the system!

- Observation: for most of the applications (e.g., MOD) many requests for the same stream (e.g., new released movies) arrive close to each other
- Solution: Share streams among multiple requests

- How?

Stream Sharing

- **Batching requests:** introduce a startup delay and multiplex single stream to support all the requests for the same stream
- **Piggybacking:** if two (or more) streams are close enough, speed-up one and slow down the other so they meet and then share streams
- **Buffer sharing:** keep blocks of the first request in memory so that the second request can be served from memory (rather than disk)

Additional Issues (1)

- VBR support
  - Approach: approximate bandwidth with segments of constant bitrate and buffers
- Pause and resume
  - Stop a stream at a specific point and resume from the same point
- Fast forward and fast rewind (speed up stream)
  - Send more data during each time period
  - Drop some frames or blocks at the server side
  - Have separate “trick” files and switch between them

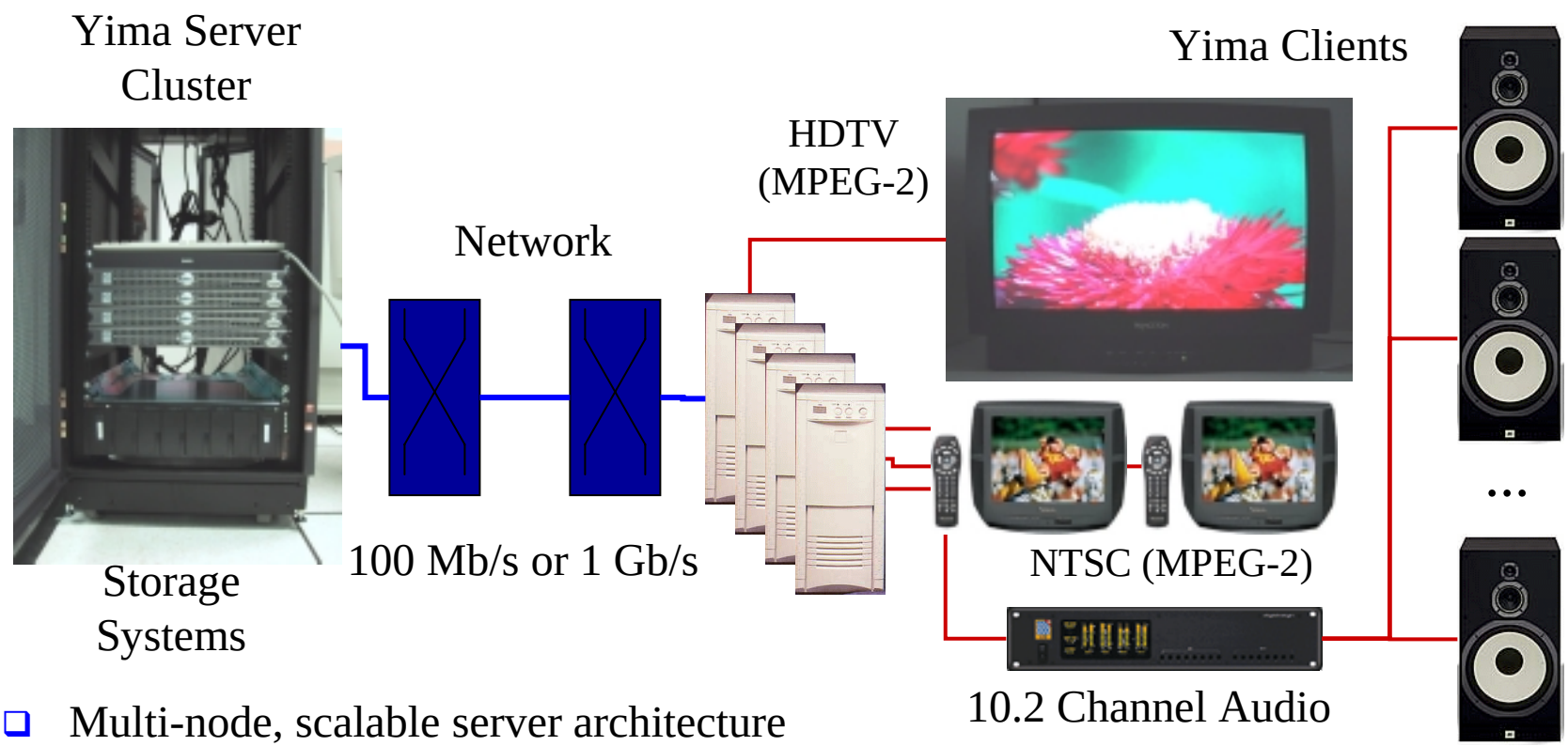
Additional Issues (2)

- Synchronization between multiple streams
- Audio/video synchronization is very sensitive
  - Synchronization must be addressed at different levels
    - Server
    - Network
    - Client
- Random Data Placement
- Deadline driven scheduling approach
  - Probabilistic model to hiccup probability

Additional Issues (3)

- Multi-zone disk drives (variable transfer rate)
  - E.g., Track-pairing
  - E.g., FIXB (elevator allocation of blocks over zones)
- Heterogeneity
  - Different media types
  - Different disk types
- Fault tolerance: what kind of failures are acceptable?
  - Replication vs. parity-based
  - OnLine reorganization

Yima *Multichannel Streaming System*



- Multi-node, scalable server architecture
- Media format independent: supports DVD MPEG-2 (8 Mb/s), HD MPEG-2 (20 Mb/s), MPEG-4 (800 Kb/s), 10.2 channel audio, etc.
- Standard transmission protocols: RTP, RTSP
- Selective retransmission of lost media packets for improved playback quality
- Synchronization across multiple media streams