Multimedia Operating Systems

- Disk Scheduling
- Admission Control
  - Deterministic Approach
  - Statistical Approach

The notion of "Real-Time"

A ***real-time process*** is a process which delivers the results of the processing in a given time-span.

⇒ The system must enforce externally-defined time constraints.

⇒ Speed and efficiency are not the main characteristics of a real-time system.

The playback of a video sequence is only acceptable when it is presented neither too quickly nor too slowly.

⇒ Timing and logical dependencies among different related tasks, processed at the same time, also must be considered.

Audio data sometimes must be synchronized with video data.

Real-Time Scheduling

- To fulfill the timing requirements of continuous media, the operating system must use real-time scheduling technique.

- The scheduler must consider the entire end-to-end data path.
  - The CPU is just one of the resources.
  - Other components include main memory, storage, I/O devices and networks.

Multimedia File Systems

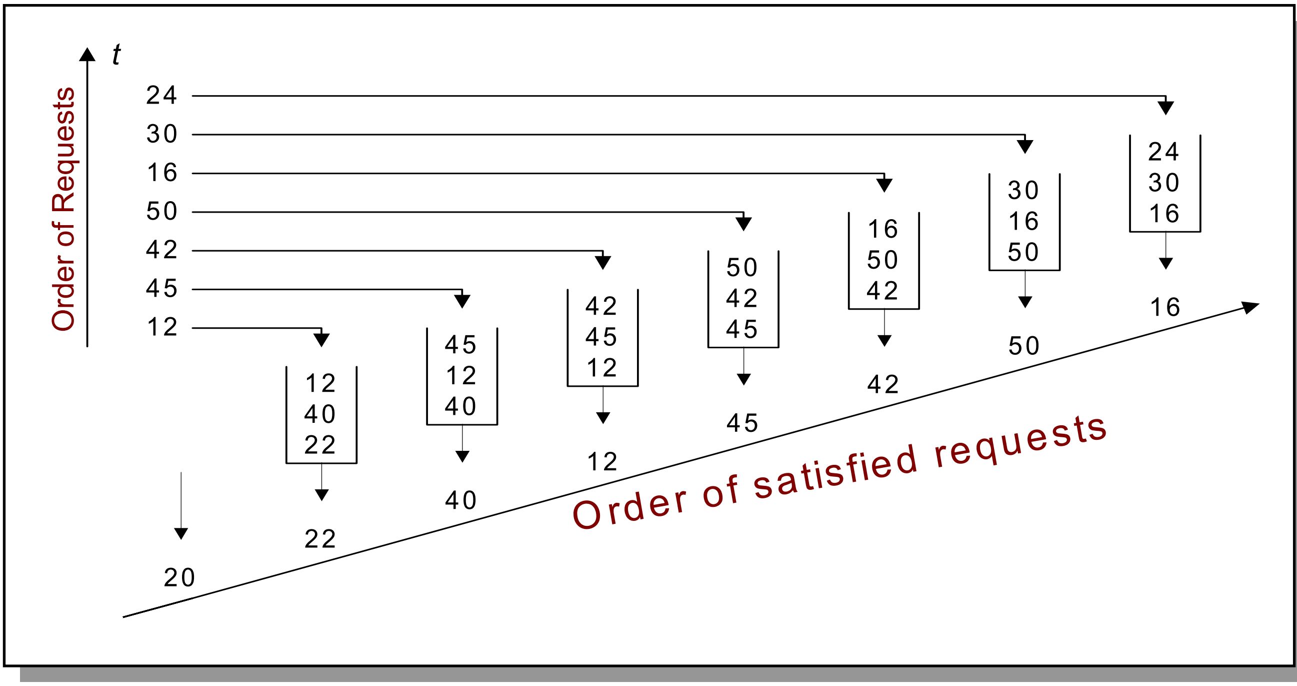
Conventional file systems:

- Provide no rate guarantees for data retrieval.
- Unsuitable for continuous media data.

Continuous media file systems:

- Guarantee that once a request is accepted, data are retrieved at the requested rate.

Disk Scheduling: FCFS



Advantages:

- Intrinsically fair
- Simple

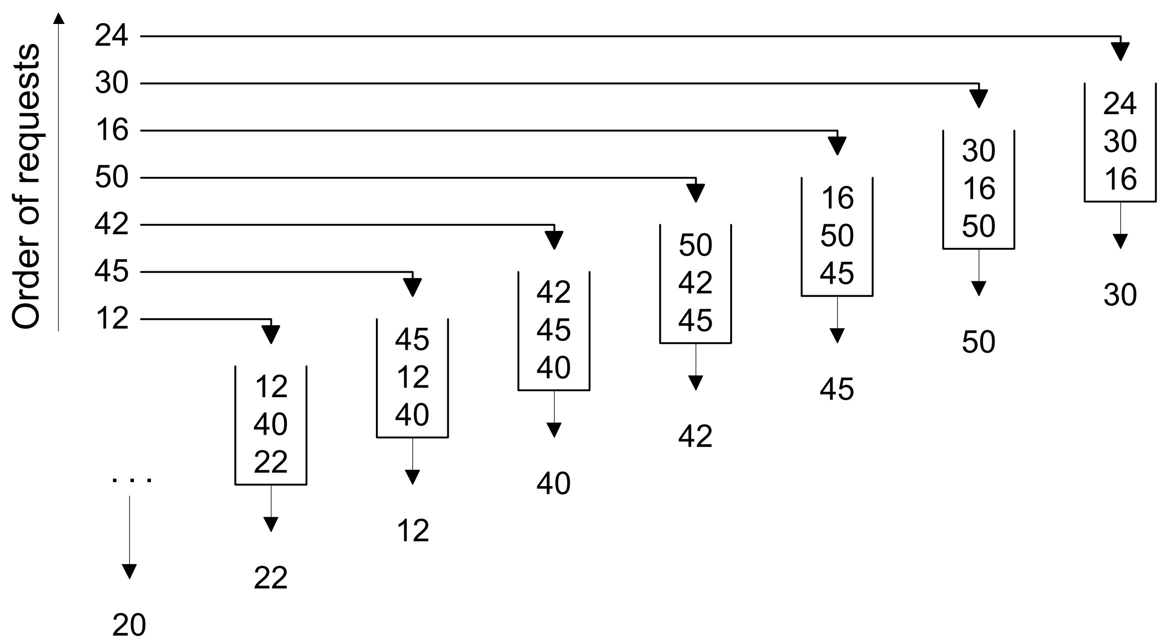
Disadvantages:

- Not optimal with respect to head movement.

  $\Rightarrow$ *high average seek time.*

Disk Scheduling: Shortest-Seek-Time First (SSTF)

SSTF selects among all requests the one with the minimum seek time from the current head position.



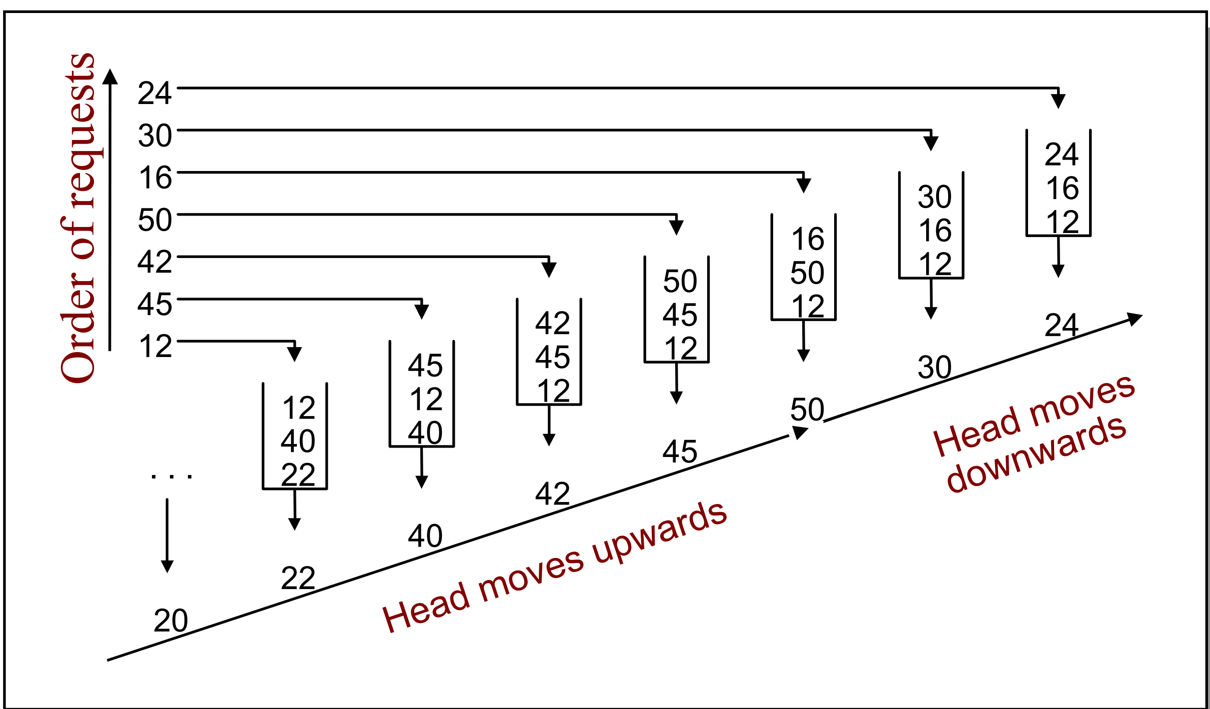
Advantage: optimal in terms of seek time.

Disadvantage: request targets in the middle of the disk are preferred over those in the innermost and outermost disk areas.

⇒ *Starvation can occur.*

Disk Scheduling: SCAN

- Like SSTF, SCAN orders requests to minimize seek time.
- In contrast to SSTF, SCAN takes the direction of the current disk movement into account.
  - It first serves all requests in one direction until it does not have any request in this direction anymore.
  - The head movement is then reversed and service is continued.



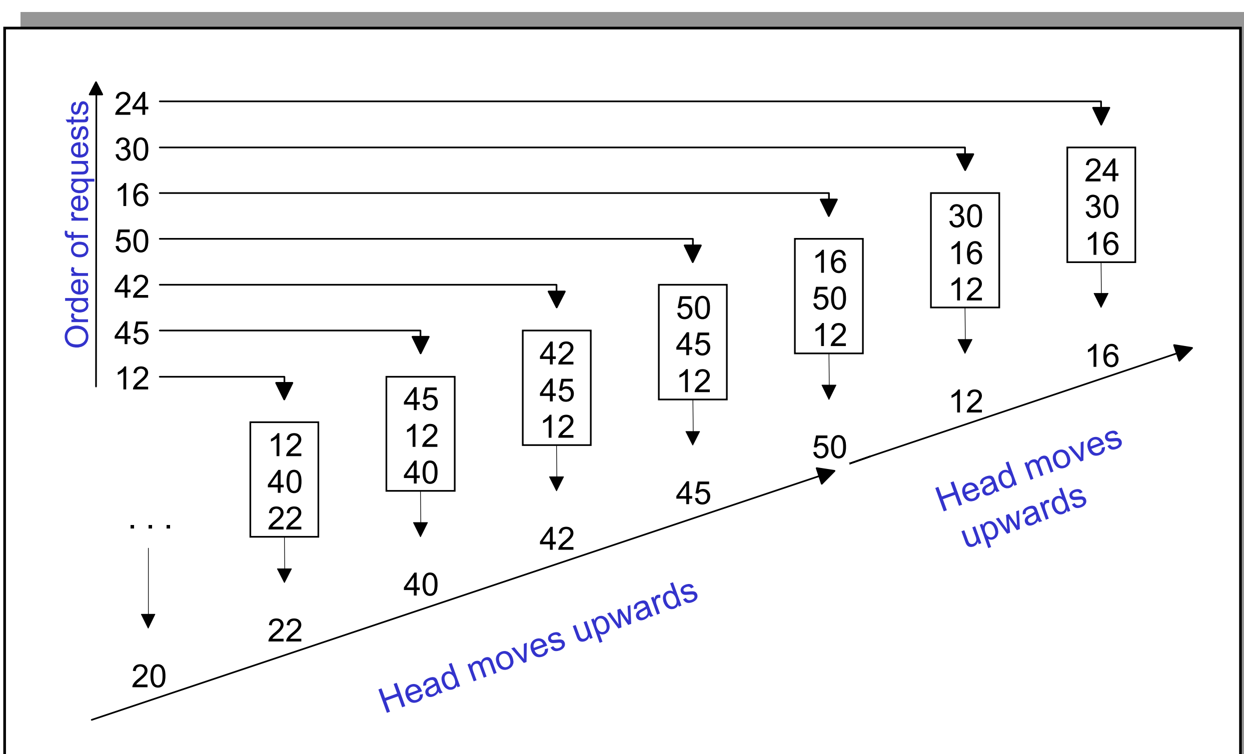
Note: *Middle tracks still get a better service then edge tracks.*

Disk Scheduling: C-SCAN

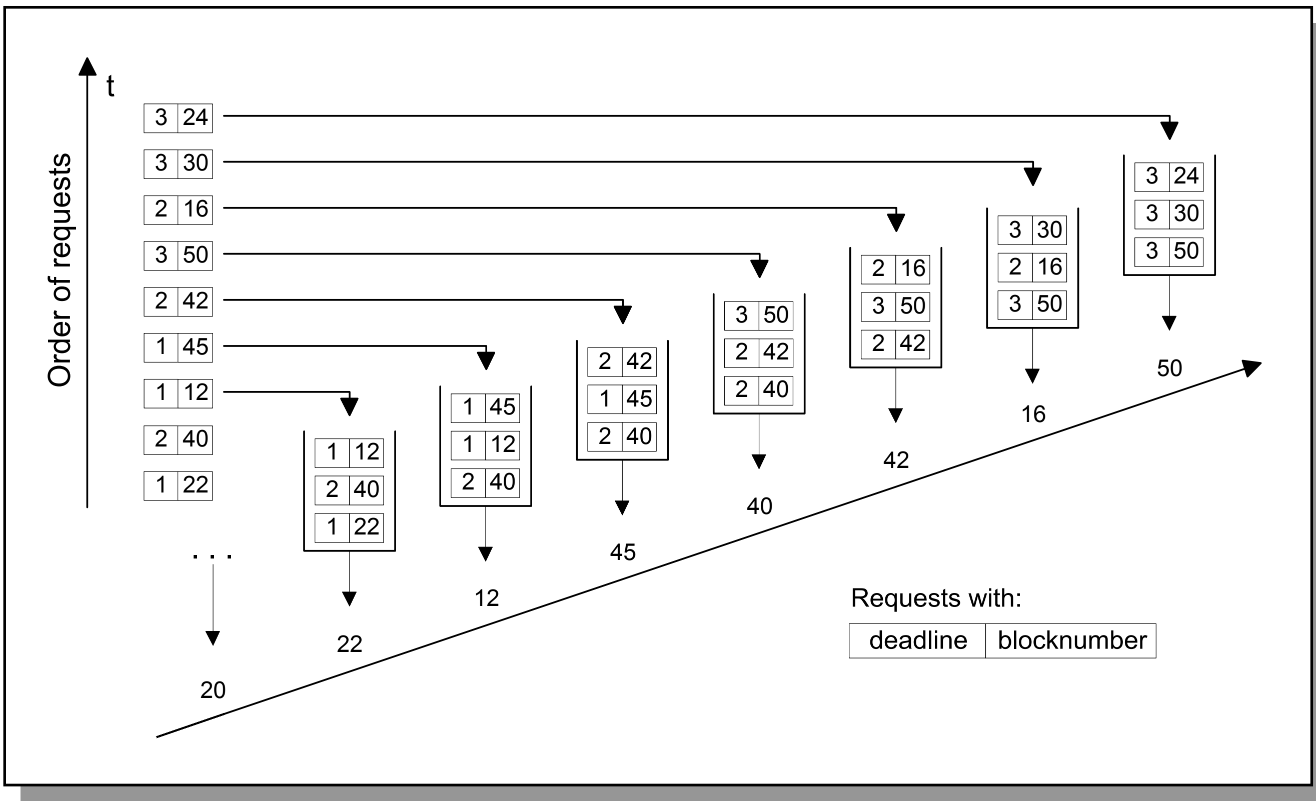
C-SCAN only retrieves data in one direction to ensure fairness.

⇒ One idle head movement from one edge to the other between two consecutive scans.

⇒ Performance of C-SCAN is somewhat less than SCAN.



Earliest Deadline First (EDF)



Employment of EDF in the strict sense results in poor throughput and excessive seek time.

SCAN-EDF: Algorithm

- All requests are forced to have release times that are multiples of the period *p*.

  ⇒ All requests have deadlines that are multiple of the period *p*.

- Like in EDF, the request with the earliest deadline is always served first.

- Among requests with the same deadline, the specific one that is first according to the scan direction is served first.

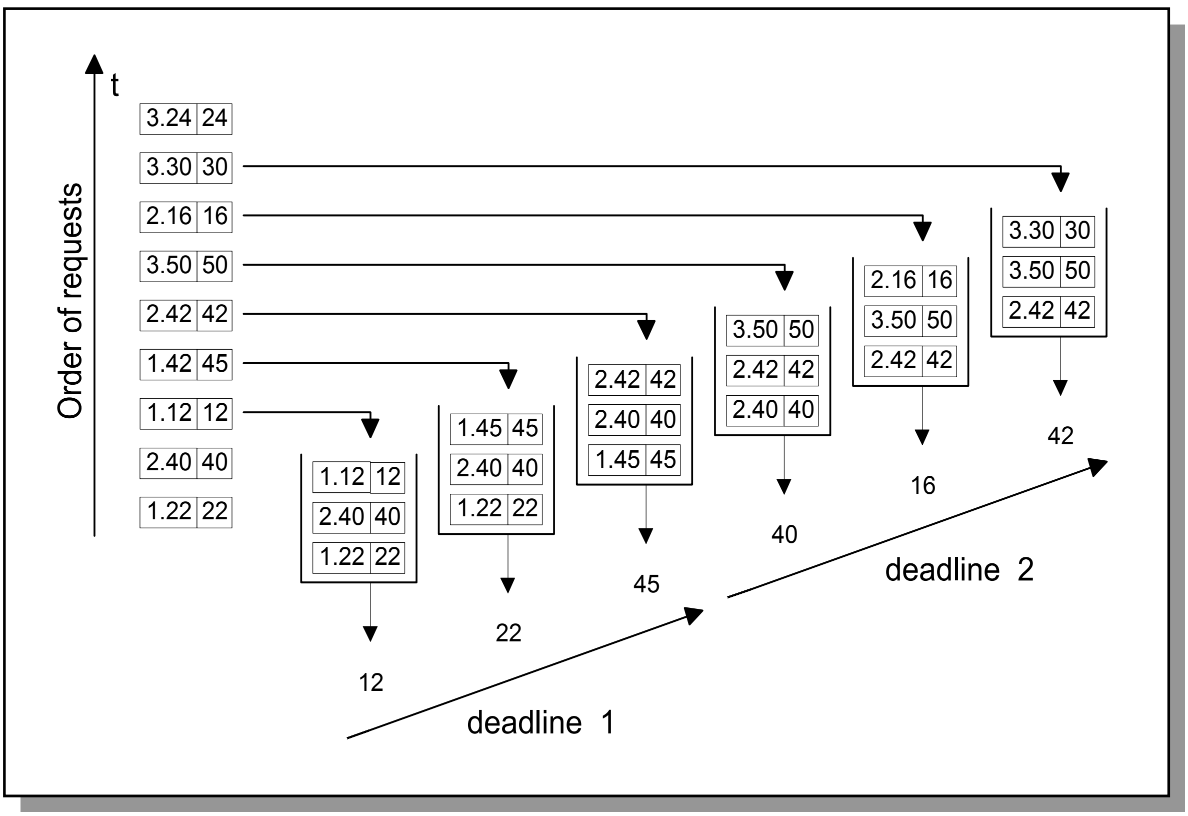
IMPLEMENTATION

If D_i is the deadline of task i and N_i is the track position, the deadline can be modified to be $D_i + f(N_i)$.

Note: The function f prioritizes tasks of the same deadline according to their positions on disk.

SCAN-EDF: Example

$$N_{max} = 100 \quad \text{and} \quad f(N_i) = N_i \,/\, N_{max}$$

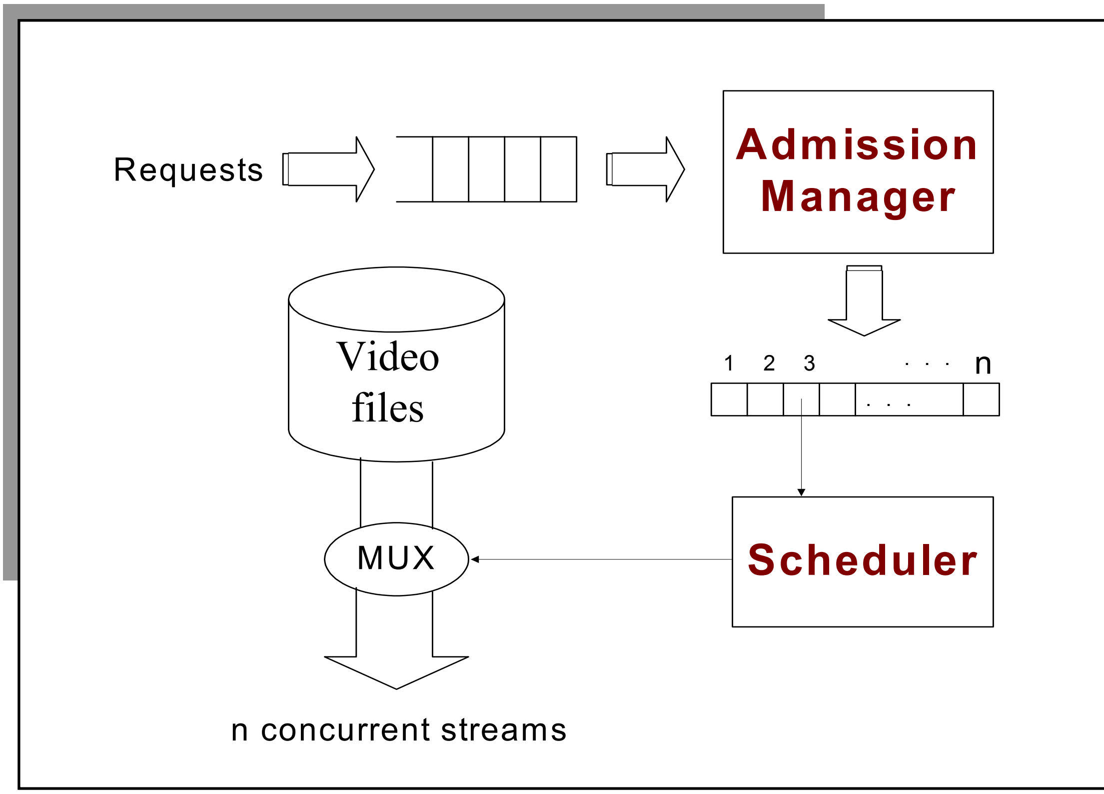


Note: The simple function *f* does not take into consideration the current direction of the head movement.

SCAN-EDF: Advantages

- **EDF**: Not optimal with respect to disk movement.
- **SCAN**: Not taking deadline into account.
- **SCAN-EDF**: Having the benefits of both EDF and SCAN.

Admission Control

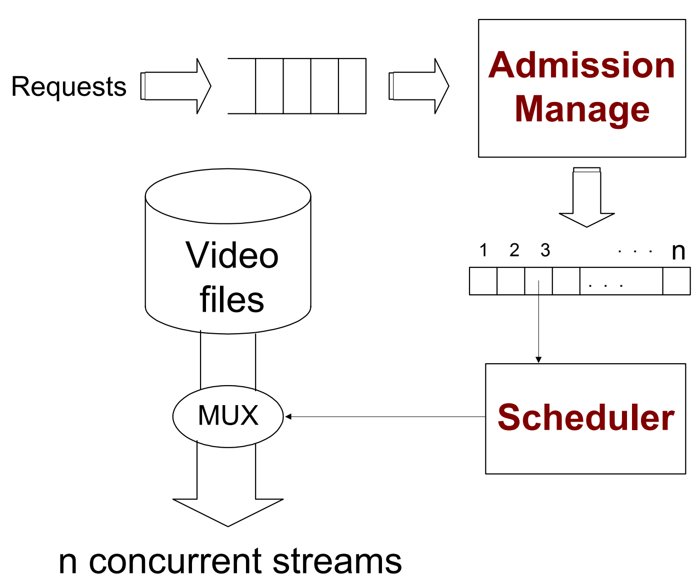


- ***Admission manager*** determines whether a new client can be admitted for service without disturbing the clients being served.
- Once the client is admitted, its requirements must be satisfied during the course of service.
- After a new client is admitted, the ***scheduler*** schedules the client of when and how it is served.

Admission Criterion

- ***Service time*** is the total time spent retrieving media blocks of currently served clients for one round.
- ***Round duration*** is the minimum playback duration among the currently served clients for a round.
- The ***admission criterion*** is:

$$Service\_time \leq Round\_duration$$



Admission Control: Goals

- To serve as many clients as possible per time unit (**high throughput**).
- To maintain a **high utilization** of the resources.
- To offer **minimum latency** for the clients.

Deterministic Admission Control

- Different admission control techniques compute the service time differently.
- Deterministic approach assumes the worst-case scenarios in computing the service time.

Example: *The service time for SCAN disk scheduling is computed as follows.*

$$service\_time = seek\_time\_per\_track \times \max\_no\_of\_track + block\_retrieval\_time \times \sum_{j=1}^{n} k_j$$

$$block\_retrieval\_time = seek\_time\_per\_track + \max\_rotational\_delay + transfer\_time\_per\_block$$

Deterministic Admission Control: Advantages

- **Advantages**:

  - The continuity requirements of each client are not violated during the entire course of their playback.

  - The admission control algorithm is easy to implement.

- **Disadvantages:**

  - The media server is underutilized since the average time for retrieving a block is usually much lower than the worst case value.

  - The throughput of the system is much less than the peak one.

Admission Control: Statistical Approach

- It extrapolates the average block retrieval time in future rounds based upon the history of the average retrieval times of the most *W* recent rounds.

- It admits the client if the following criterion is satisfied:

$$Predicted\_avg\_retrieval\_time \times \left(\sum_{i=1}^{n+1} k_i \right) \leq Round\_duration$$

Statistical Approach

Advantage:

- Server resources are better utilized.
- throughput is significantly increased.

Disadvantages:

- It does not provide absolute guarantee to the clients since the algorithm employs prediction.
  - ⇒ There are *overflow rounds*.
  - ⇒ Several techniques can be applied to distribute the media loss among clients.
- The algorithm is more complicated to implement.

Disk Stripping

An effective way to distribute the workload evenly across *n* disks is to stripe video files across the disks in a round-robin fashion.

⇒ The entire aggregate bandwidth of disks is available to show the most popular movies.

⇒ Unpopular movies do not render the disks that store them underutilized.

⇒ Multiple concurrent streams of a video to be supported without having to replicate the video.