Multimedia Servers

- Multimedia: digital audio, video, images,..
- Streaming audio and video
  - Very different characteristics from textual and numeric files
  - Need different techniques for managing multimedia data
- Video: sequence of images played out at a constant rate
- Digital video is often stored in compressed format

Need For Video Compression

- Large data rate and storage capacity requirement

Satellite imagery	180x180 km^2 30 m^2 resolution	600 MB/image
NTSC video	30 frames/s, 640x480 pixels, 3 bytes/pixel	30 MBytes/s

- Compression algorithms exploit:
  - Spatial redundancy (i.e., correlation between neighboring pixels)
  - Spectral redundancy (i.e., correlation between different frequency spectrum)
  - Temporal redundancy (i.e., correlation between successive frames)

Requirements for Compression Algorithms

- Objectives:
  - Minimize the complexity of the encoding and decoding process
  - Ensure a good quality of decoded images
  - Achieve high compression ratios
- Other general requirements:
  - Independence of specific size and frame rate
  - Support various data rates

Classification of Compression Algorithms

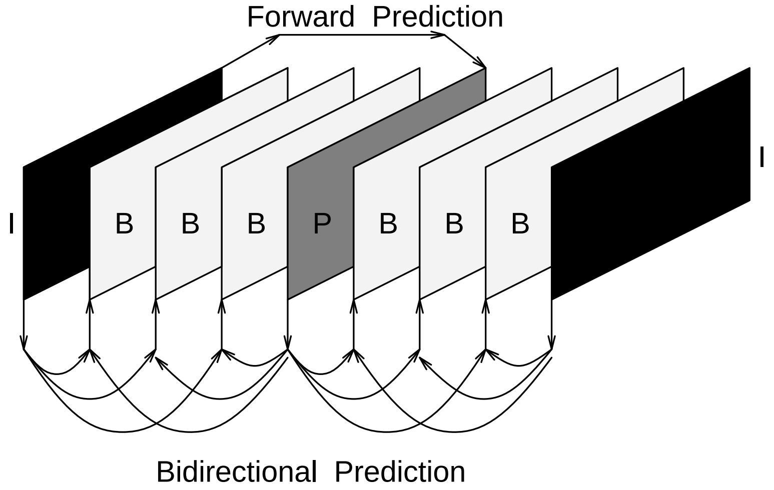
- Lossless compression:
  - Reconstructed image is mathematically equivalent to the original image (i.e., reconstruction is perfect)
  - Drawback: achieves only a modest level of compression (about a factor of 5)
- Lossy compression:
  - Reconstructed image demonstrates degradation in the quality of the image $\Rightarrow$ the techniques are irreversible
  - Advantage: achieves very high degree of compression (compression ratios up to 200)
  - Objective: maximize the degree of compression while maintaining the quality of the image to be “virtually lossless”

MPEG - An Overview

- Two categories: intra-frame and inter-frame encoding
- Contrasting requirements: delicate balance between intra- and inter-frame encoding
  - Need for high compression ⇒ only intra-frame encoding is not sufficient
  - Need for random access ⇒ best satisfied by intra-frame encoding
- Overview of the MPEG algorithm:
  - DCT-based compression for the reduction of spatial redundancy (similar to JPEG)
  - Block-based motion compensation for exploiting the temporal redundancy
    - ∗ Motion compensation using both *causal* (predictive coding) and *non-causal* (interpolative coding) predictors

Exploiting Temporal Redundancy

- Three types of frames in MPEG:
  - I-frames:
    - ∗ Intra-coded frames, provide access points for random access - yield moderate compression
  - P-frames:
    - ∗ Predicted frames are encoded with reference to a previous I or P frame
  - B-frames:
    - ∗ Bidirectional frames encoded using the previous and the next I/P frame
    - ∗ Achieves maximum compression



Multimedia Storage Servers

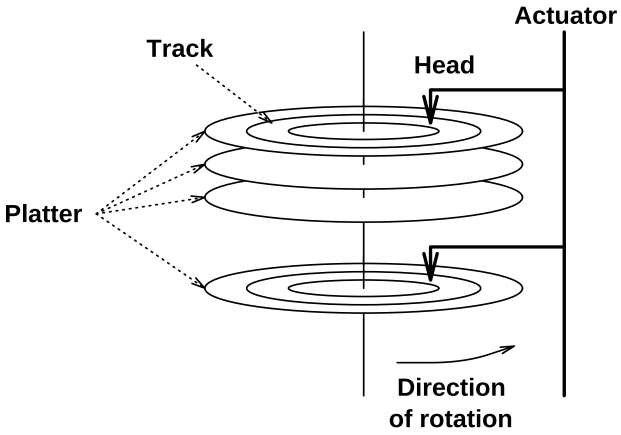
- Digitally stores heterogeneous data objects (consisting of audio, video, imagery, textual, and numeric data) on extremely high capacity storage devices
- Fundamental differences in data type characteristics and requirements
  - Best-effort service for text vs. real-time for video
  - Small read/writes for text vs. large read/writes for video
  - ....

Approach

- Techniques for efficiently managing video data
  - Placement techniques
  - Fault tolerance issues
  - Scheduling, retrieval, and admission control
  - I/O stream sharing (buffering, batching, caching, ...)
- Methodology:
  - What are the fundamental issues ?
  - How to address these issues ? (Theory)
  - How to instantiate the solutions ? (Practice)

Terminology

- Disk fundamentals:
  - Seek time
  - Rotational latency
  - Transfer rate
  - Scheduling algorithms: FCFS, SCAN, SSTF, SATF



Terminology (Cont'd)

- Disk arrays
- Striping
  - Interleave the storage of each media stream among disks
  - Stripe unit: maximum amount of logically contiguous data that is stored on a single disk
  - Degree of striping: Number of disks across which a media stream is striped
- Redundant and non-redundant disk arrays

Video Storage Server: Fundamentals

- Data transfer rate of disks $\gg$ data rate requirement of isolated video streams $\Rightarrow$ designing single-user video servers is straightforward
- Server stores digitized video streams on an array of disks
- Clients can request the retrieval of video streams for real-time playback
- Two possible server architectures:
  - Client-pull
  - Server-push

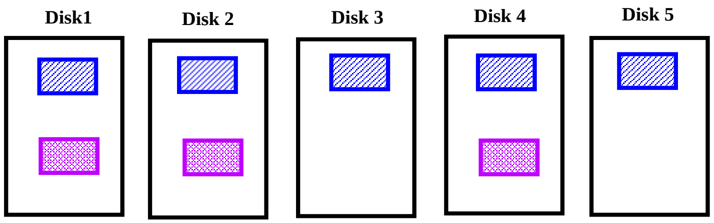
Client-pull Architecture

- Server retrieves data only in response to an explicit request from client
- Used in conventional file system to provide *best-effort* service
- Adapting client-pull architecture for video: clients ensure playback continuity by
  - Determining the playback instant of a frame
  - Estimating response time for each request
  - Issuing a read request accordingly
- Response time: a function of the system load ⇒ varies widely over time ⇒ estimation is non-trivial

Server-push Architecture

- Periodicity of video playback ⇒ service clients in periodic *rounds*
- Round: retrieve a fixed number of frames for each media stream
- Continuous retrieval ⇒ total service time must not exceed the playback duration of frames retrieved during a round

Efficient Placement on Disk Arrays



- Stripe video streams on disk arrays in terms of blocks (or stripe units)
- Two paramaters: stripe unit size and degree of striping
- Stripe unit size (block size): use large (128-512 KB) block size
  - Large block size reduces disk seek and rotational latency overheads

Retrieval Techniques

- Streaming media data imposes real-time constraints on retrieval
  - Need to retrieve 30 frame in each second
  - Client or server buffering can provide some leeway but still need guarantees
- Performance guarantees on retrieval → need to limit the number of clients accessing a server
- Employ admission control algorithms

Admission Control

- Server push retrieval: retrieve f frames in each periodic round R
- Continuous playback requirements: retrieval time of $f_1, f_2, ..f_k$ frames for all k clients should not exceed R
- Admission control test
  - Estimate resoure needs of new client (time to retrieve f_i frames)
  - Verify if total resource needs $\leq$ capacity (total retrieval time $\leq R$)
  - If so, admit, else deny