MULTIMEDIA ARCHITECTURE

MULTIMEDIA SYSTEM ARCHITECTURE

Multimedia encompasses large variety of technologies and integration of multiple architectures

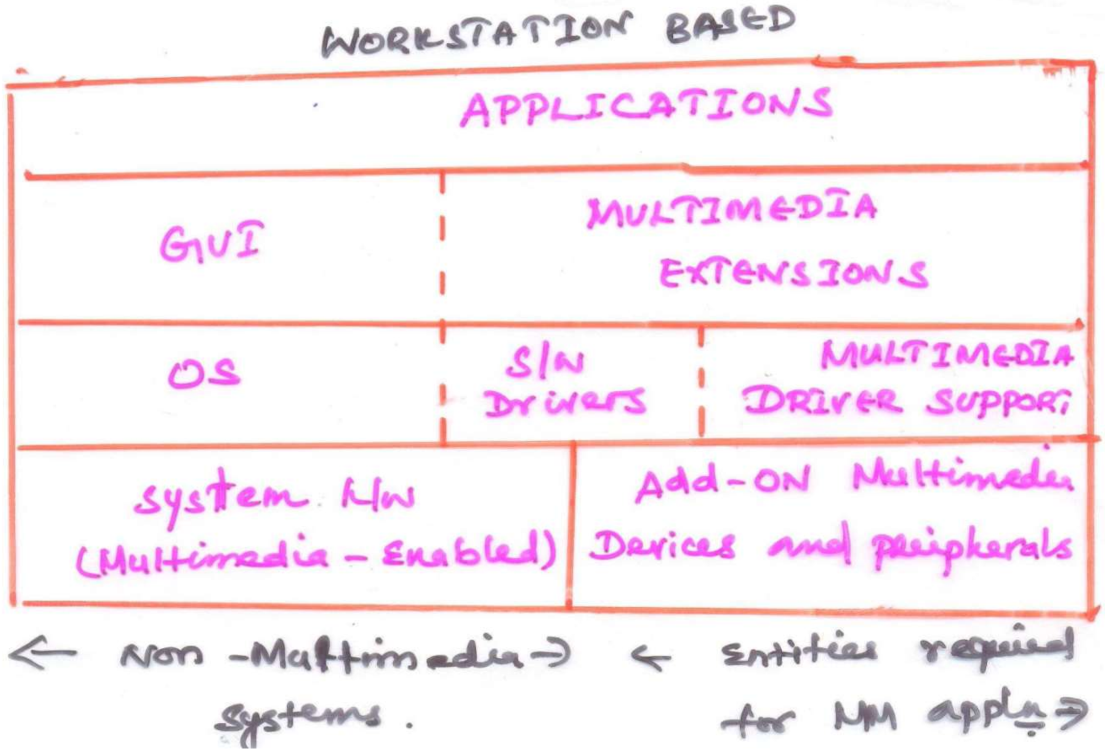
Multimedia systems integrate with standard user interface systems Microsoft windows, X windows or Presentation manager

System can operate with or without special hardware for multimedia

Hardware interfaces are needed for video animation and compression boards

Standard device independents APIs and file formats allow a wide range of applications published by various developers to publish applications.

WORKSTATION BASED ARCHITECTURE



•RHS of the figure represents new architectural entities required for supporting multimedia applications.

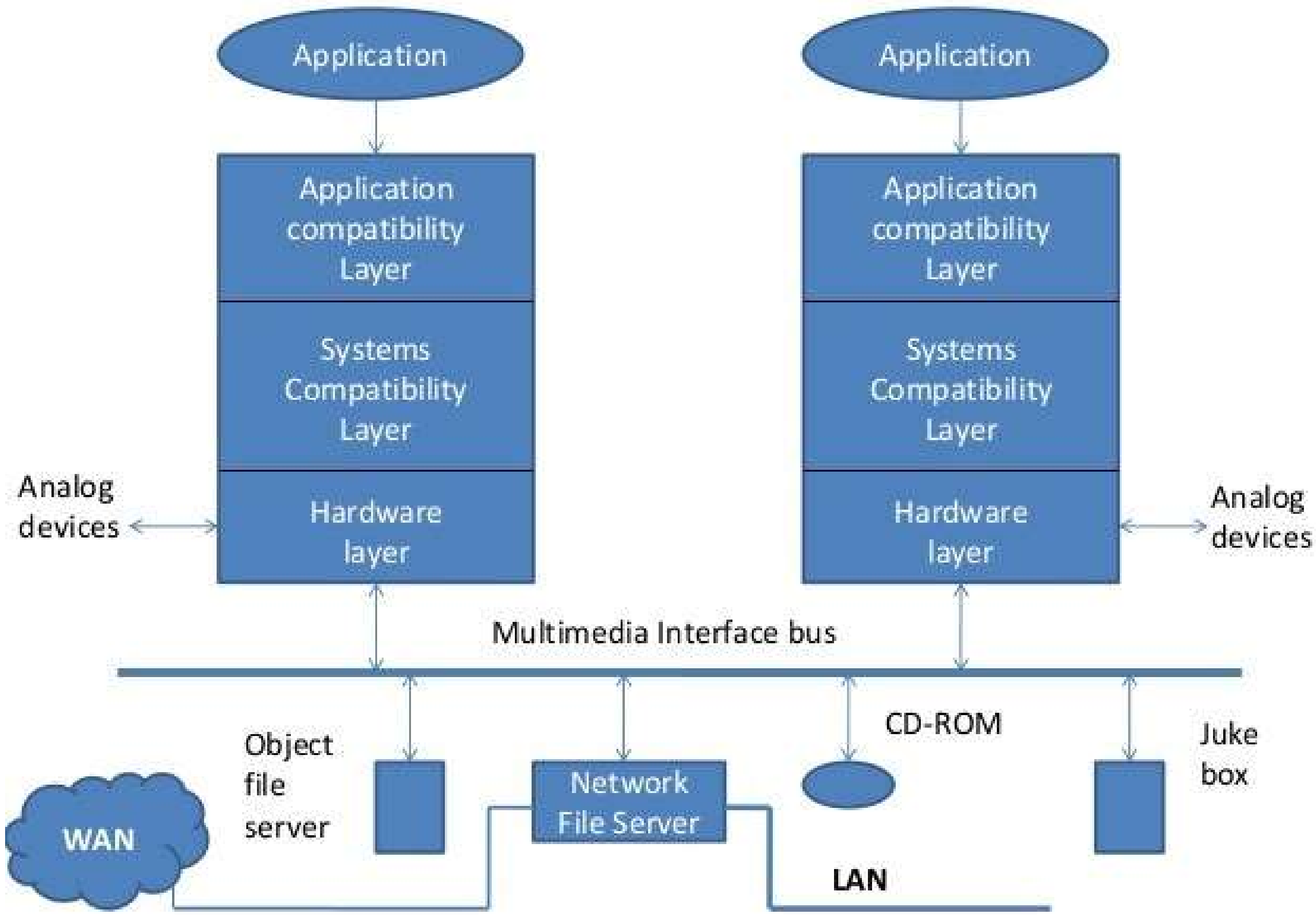
•The add-on multimedia devices and peripherals includes scanners,video cameras and sound equipments along with device controllers and encodinghardware

•Software device driver: interfaces application to the device.

•GUI designed for windows require control extensions to support full motion video

MULTIMEDIA ARCHITECTURE BASED ON INTERFACE BUS

Fig: Multilevel architecture based on interface bus



MULTIMEDIA ARCHITECTURE BASED ON INTERFACE BUS

Interactive Multimedia Association (IMA) has a task group that defines the architectural framework for multimedia.

The task group has two areas of concentration:

- Desktops
- Servers

Desktop focus is to define the interchange formats that allow multimedia objects to be displayed on any workstation or PCs

Server focus is for defining the class libraries for multimedia objects that enables distributed multimedia applications across multivendor platforms

Architecture of IMA is based on defining interfaces to a multimedia interface bus.

Interface bus would be the interface between systems and multimedia sources providing streaming I/O services, including filters and translators.

NETWORK ARCHITECTURE FOR MULTIMEDIA SYSTEMS

Multimedia systems have special networking requirements because of

- Large volume of data such as images and video messages

Increasing demands on the network, requirements of different applications and need to optimize network resource leads to task based approach in networking.

Task Based Multi-level Networking

Ethernet and Token ring (broadcast networks) – uniform solution for all tasks.

Emergence of Groupware technologies - customizing the network to a task

Tasks can be broken down into following types based on data volume, source and speed:

- Data transfer for text
- Data transfer for images
- Data transfer for audio and video clips
- Data duplication to user workstations
- Data replication among servers

NETWORK ARCHITECTURE FOR MULTIMEDIA SYSTEMS

Text Transfer:

- Least demanding among various transfer types
- Volumes are smaller
- Small and potentially low cost network service can provide adequate performance of text transfers.

. Image Transfer:

- More demanding due to high volume of data
- A black and white image even after compression is very large.
- Gray scale image after compression is also very large.
- TCP/IP with ethernet configuration is found suitable

NETWORK ARCHITECTURE FOR MULTIMEDIA SYSTEMS

Audio & video clips:

- More intense than imaging because of the third dimension- “time”
- Momentary pause has to be taken care
- A 10-Mbits/sec network was used to stream the data
- Requires more expensive components both at workstations and at servers
- Distribute the network bandwidth for more demanding applications - this approach is called multilevel networking.

NETWORK ARCHITECTURE FOR MULTIMEDIA SYSTEMS

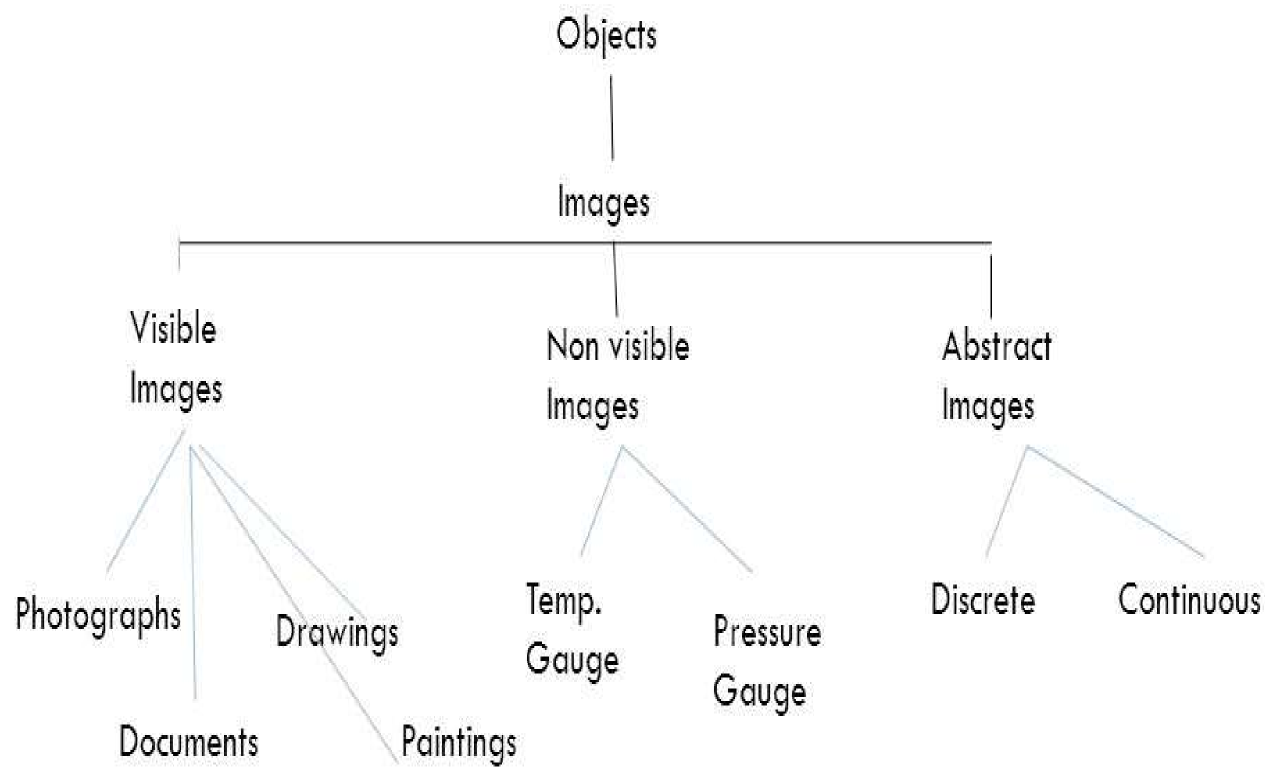
Duplication

- Duplication – process of duplicating an object that the user can manipulate
- This is based on data transfers between servers within a network
- Imaging objects are duplicated on the host or at workstations for short duration while the user works with a specific document
- The object is discarded after use. No synchronization between source and duplication
- Disadvantage as image/object keeps changing

High speed server to server:

- it can achieved by the process of replication.
- Replication is defined as the process of maintaining two or more copies of the same object in a network
- Periodically resynchronize to provide faster and more reliable access to data.

DEFINING OBJECTS FOR MULTIMEDIA SYSTEMS



Basic objects

- Text
- Images
  - Visible
  - Non-visible
  - Abstract
- Audio and voice
- Full motion and Live video

DEFINING OBJECTS FOR MULTIMEDIA SYSTEMS

Visible images - drawings (blueprints, engg drawings, town layouts), documents (scanned or images), paintings (scanned or created using paint application), photos, still frames using video camera

Compression information should also be stored- needed for decompression

Information about resolution, orientation of image is also stored

Non-visible images are those that are not stored as images but as displayed as images – temp, pressure gauges and other metering displays

Abstract images – computer generated images based on some arithmetic calculations – eg: fractals

MULTIMEDIA DATA INTERFACE STANDARDS

Earliest and simplest video processing standards used were:

- Intel's DVI (Digital Video Interface)
  - Processor-independent specification for video interface
- Apple's Quicktime
  - Designed by Apple computers, to support multimedia applications
  - Capable of handling various formats of digital videos, pictures, sounds, panoramic images.
  - Video file formats - QuickTime movie (mov), MPEG-2,4 , AVI, 3GPP
  - Audio file formats - iTunes audio, MP3, WAV, AMR.

Microsoft AVI (Audio Video Interleave)

- Offers low-cost, low-resolution video processing
- Suitable for average desktop users

MULTIMEDIA DATA INTERFACE STANDARDS

Device independent Bitmap (DIB) – bitmap, color and color palette info

RIFF Device independent Bitmap (RDIB) – Resources Interchage File Format(RIFF) – standard in Windows – more complex bitmaps than DIB

Musical Instrument Digital Interface(MIDI) – interace standard between computer and musical intrument

RIFF Musical Instrument Digital Interface – MIDI within RIFF envelope – more complex

Palette Rich Format (PAL) – palette of 1 to 256 colors (RGB values)

Rich Text Format (RTF) – embedding graphics and other file formats within a document

Waveform Audio File Format (WAVE) – digital file representation of digital audio

MULTIMEDIA DATA INTERFACE STANDARDS

Windows Metafile Format (WMF) – vector graphic format in Windows

Multimedia Movie Format (MMM) – used for digital video animation

Apple's Movie Format - standard for file exchange by Quicktime enalbled systems

Digital Video Command set (DVCS) – digital video commands simulating VCR controls

Digital video Media Control Interface (DV-MCI) – Microsoft high level control interface for VCR controls

Vendor Independent Messaging (VIM) – developed by vendors for cross-product messages

Apple's Audio Interchange File Format - compressed audio and voice data

SDTS GIS standard – Spatial Data Transfer Standard – for geographic data

Compression and Decompression

What is Multimedia?

- Accessing information involves various media forms
- Media is some form of information like text, audio, video, etc.
- A web page primary contains text information, but it also can have pictures, animations, video clippings, audio commentaries and so on.
- When there is more than one form of media present, we call them as multimedia.
- Examples: Television(audio, video)
- A multimedia signal is one that integrates signals from several media sources, such as video, audio, graphics, animation, text in a meaningful way to convey some information.

Elements of a multimedia system

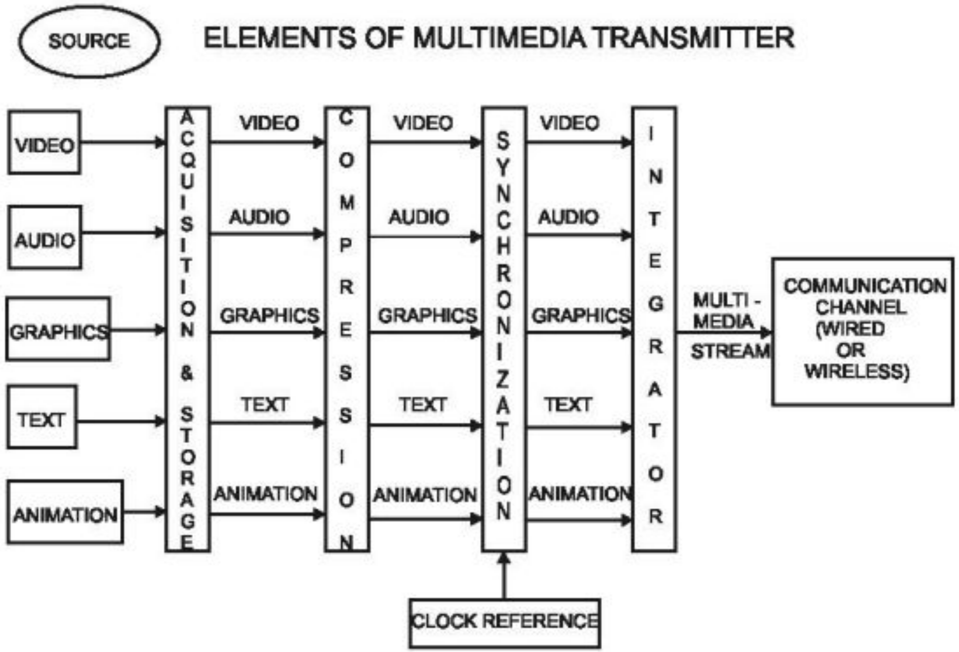


Fig 1.1: *Elements of Multimedia Transmitter*

Elements of a multimedia system

- Data acquisition and storage - get data
- Compression - make data smaller and eliminate inherent redundancies present in the media streams
- Synchronization - synchronize multiple types of media by insertion of time-stamps
- Integrator - integrates the individual streams from all media sources and transmits it through multimedia stream

On the receiver do the literal opposite.

Elements of a multimedia system - Receiver

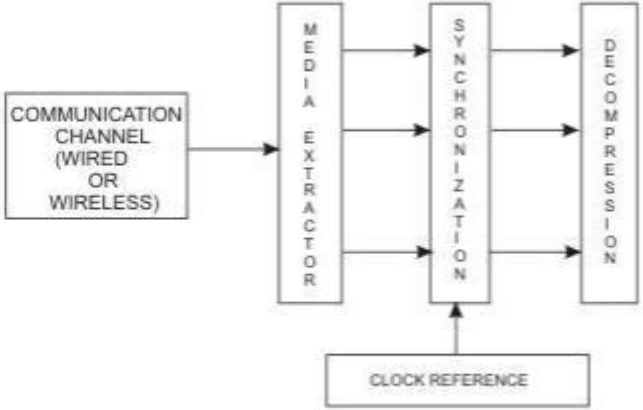


Fig 1.2 Elements of multimedia Receiver

Challenges involved with multimedia communication

- Bandwidth limitations of communication channels.
- Real-time processing requirements.
- Inter-media synchronization.
- Intra-media continuity.
- End-to-end delays and delay jitters.
- Multimedia indexing and retrieval.

- Real time processing requirements:
    - Data compression is a crucial technique for multimedia content, but processing time is a critical factor.
    - If processing time is too long, the benefits of compression can be lost
    - For instance, in still image compression, a 20:1 ratio can transmit an image in a minute, but slow processing could negate the advantage.
- Inter-media synchronization:
    - The media streams are available from different and independent sources and are asynchronous with respect to each other
    - Multimedia standards employ "time-stamping" with a system clock reference (SCR) to ensure proper synchronization. Time-stamps are appended to audio, video, and other media packets before integration.

- Intra-media continuity:
  - The extent of data compression with acceptable reconstruction quality is highly data-dependent. Wherever redundancy is more, high compression ratios are achievable, but redundancy may vary. Due to this we get variable bit rates, especially in video sequences.
  - To address this, a buffer is used to accommodate variable bit rate sources and maintain a constant bit rate for transmission.
  - Buffer management is crucial to avoid underflow (channel has no data) or overflow (buffer full so data discarded) issues, which can disrupt continuity during presentation.
- End-to-end delays and delay jitters:
  - In a multimedia broadcast or multimedia conferencing, if the users receive the multimedia contents after considerable delays or different users receive the same contents at different times, the interactivity is lost.
  - The multimedia standards available till date have addressed this problem and specified what is acceptable.

- Multimedia Indexing and retrieval:
  - **Growing Multimedia Data:** Digital storage media prices decrease, while storage capacity increases, leading to a increase in multimedia file availability.
  - **Challenges in File Retrieval:** Managing a large number of multimedia files without proper indexing can make retrieval difficult due to search complexities.
  - **Content-Based Query Systems:** Efficient retrieval can be achieved by organizing multimedia files based on their content and implementing content-based query systems.
  - **Video Summaries:** Quick browsing of multimedia files often requires video summaries, but creating these summaries is a challenging task.

Redundancies

- The biggest challenge in multimedia communication is to transmit the multimedia signals, especially the image and the video signals through limited bandwidth channels.
- So we need to compress this data in order to send it across quicker.
- We rely on the redundancies in the data to compress the data.
- 2 types of redundancies:
  - Statistical Redundancy
  - Psychovisual Redundancy

Statistical Redundancy

- **Statistical Redundancy**: Occurs in images because neighboring pixels tend to have similar intensities, except at object boundaries or illumination changes.
- **Spatial Redundancy in Still Images:** In still images, statistical redundancies are primarily spatial, present along both the x and y dimensions, due to the nature of natural two-dimensional images.
- **Temporal Redundancy in Video:** Video signals exhibit temporal redundancy as intensities of the same pixel positions across successive frames are often very similar, except in the presence of substantial motion.

Psychovisual Redundancy

- **Psychovisual Redundancy:** Arises from human perception capabilities.
- **Perception vs. Detail:** Our eyes are more sensitive to slow illumination changes than fine details and rapid intensity changes.
- **Balancing Preservation and Quality:** Exploiting psychovisual redundancy involves determining the extent to which we should preserve image details for perception and where we can compromise on reconstructed image quality.

Types of compression

- Lossless image compression
- Lossy image compression

Lossless compression

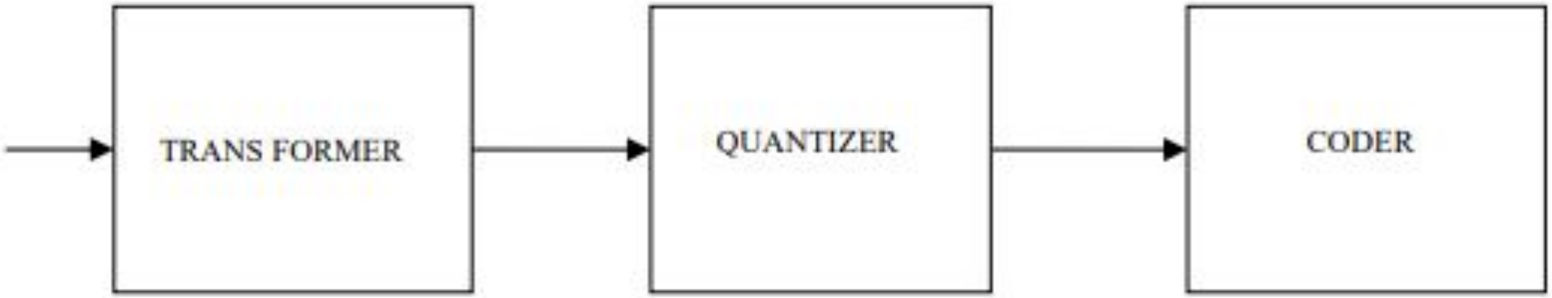
- **Lossless Image Compression:** Does not result in data loss, ensuring that the data stream before encoding and after decoding is identical, preserving image quality.
- **Statistical Redundancy Exploitation:** Lossless compression relies on statistical redundancy within the data.
- **Symbol Coding:** By transforming the image into symbols and assigning shorter code words to frequently occurring symbols and longer code words to less frequent symbols, compression is achieved.
- **Reversible Process:** The encoding and decoding process in lossless compression is fully reversible due to one-to-one mapping between symbols and their codes.
- **Limited Bandwidth Reduction:** Lossless compression achieves only modest bandwidth reduction for data transmission but maintains image quality without distortion.

Lossy compression

- **Lossy Image Compression:** Involves data loss, resulting in a reduction in the quality of image reconstruction.
- **Symbol Transformation:** Similar to lossless compression, lossy compression transforms the image into symbols.
- **Quantization:** Symbols are mapped to a discrete set of allowable levels, which leads to data compression.
- **Irreversible Mapping:** Quantization, a many-to-one mapping, is irreversible, and exact reconstruction is not possible.
- **Acceptable Loss:** Lossy compression is acceptable when the reduction in reconstruction quality is tolerable to human visual perception, and it allows for significant compression.
- **Exploiting Psychovisual Redundancy:** Lossy compression schemes rely on psychovisual redundancy and aim to identify areas where quality loss can be tolerated.

Lossless compression	**Lossy compression**
There is no loss of data	There is always a loss of data
Exactly reversible.	Not reversible.
Exploits statistical redundancy	Exploits psychovisual redundancy.

Elements of image compression system



- Transformer:
    - **Transformer Block:** Part of image compression(no compression happens here), it transforms input data to facilitate compression.
    - **Local vs. Global Transformation:** The transformation can be local (involving nearby pixels) or global (involving the entire image or pixel blocks).
- Quantizer:
    - **Quantizer block:** It generates a limited number of symbols for representing the transformed signal.
    - **Many-to-One Mapping:** Quantization is a many-to-one mapping, meaning it's an irreversible process.
    - **Two types:**
        - Scalar quantization, which quantizes data element by element
        - Vector quantization, which quantizes a block of data at once.
    - **Lossy Block:** Quantization is the only lossy block in the image compression system, as it introduces some loss of data.

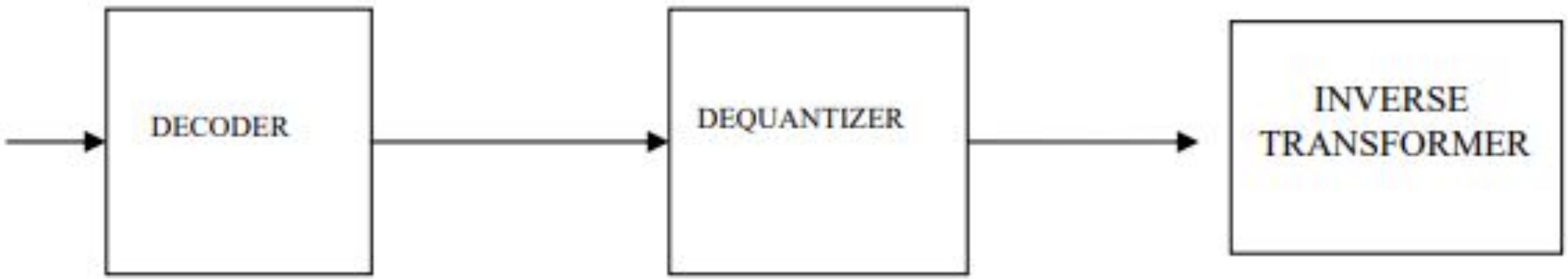
→ Coder:

- **Coding Symbols:** Coders assign code words to the symbols obtained after quantization.

2 types:

- **Fixed-length coding (FLC):** which have codeword length fixed, irrespective of the probabilities of occurrence of quantized symbols.
- **Variable length coding (VLC):** also known as entropy coding, assigns code words in such a way as to minimize the average length of the binary representation of the symbols. This is achieved by assigning shorter code words to the more probable symbols.

Elements of Image decompression system



Literally the opposite

Quantizer

- Quantization is the process of mapping a set of continuous-valued samples into a smaller, finite number of output levels.
- 2 types: scalar, and vector quantization
- In **scalar quantization**, each sample is quantized independently.

A scalar quantizer Q(.) is a function that maps a continuous-valued variable s having a probability density function p(s) into a discrete set of reconstruction levels r_i (i=1,2,....,L) by applying a set of the decision levels d_i (i=1,2,....,L)

$$Q(s) = r_i \quad \text{if } s \in (d_{i-1}, d_i] \quad i=1,2,...,L$$

L is the number of levels.

- In **vector quantization**, each of the samples is not quantized. Instead, a set of continuous-valued samples, expressed collectively as a vector is represented by a limited number of vector states.

  Let $\dot{s} = Q(s)$ be quantized variable, then error $e = s - \dot{s}$, and distortion D is measured as MSE(Mean square error) $D = E(s - \dot{s})^2$

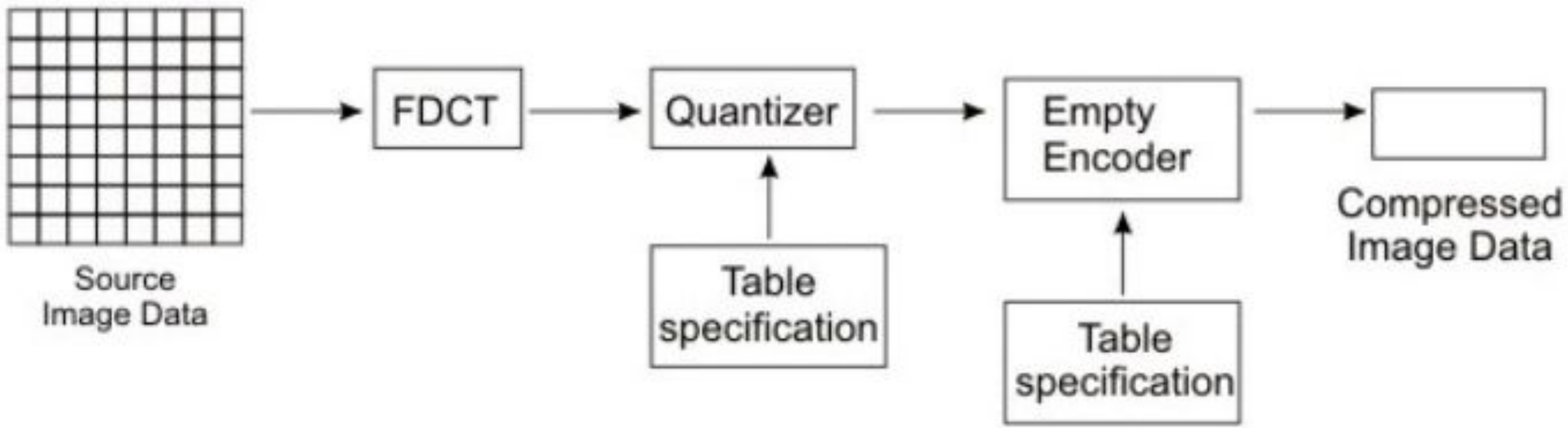
  Now choose d_i and r_i such that D is minimized.

Still Image Compression Standards - JPEG

Introduction

- **Importance of Coding Standards:** In the rapidly evolving field of imaging technology and image compression, coding standards are essential.
- **Compatibility and Interoperability:** Coding standards ensure compatibility and interoperability between image communication and storage products from different vendors.
- **No Standards, No Communication!!!! :** Without standards, encoders and decoders cannot effectively communicate, leading to challenges in data exchange and service provision.
- **JPEG Standard:** The JPEG standard is the most widely adopted and used standard for compressing and coding continuous tone monochrome and color images of various sizes and sampling rates.

Architecture

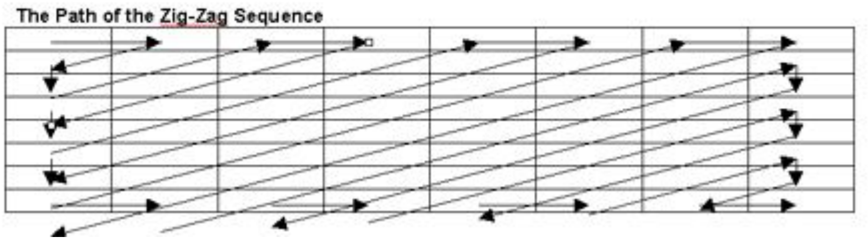


➜ Forward Discrete Cosine Transform (FDCT):

- The still images are first partitioned into non-overlapping blocks of size 8x8
- The image samples are shifted from unsigned integers with range [0, 2^{p-1}] to signed integers with range [-2^{p-1}, 2^{p-1}], where p is number of bits(8 here)
- DCT algorithm transforms spatial data into frequency components using cosine functions

➜ Quantization:

- Each of the 64 coefficients from the FDCT outputs of a block is uniformly quantized according to a quantization table.
- Since the aim is to compress the images without visible artifacts, each step-size should be chosen as the perceptual threshold or for “just noticeable distortion”.
- The quantized coefficients are zig-zag scanned, this is done to exploit redundancy



➔ Encoder:

- Then we encode DC and AC.
- DC
  - **DC represents overall brightness**: The DC coefficient reflects the average brightness of a block in an image.
  - **Encoding DC**: To save space, the DC coefficient is encoded as the difference between its value and the DC coefficient of the previous block. This process uses less data because it focuses on how much the brightness changes from one block to the next.
  - **Example:** If the DC coefficient of one block is 100 and the next block's DC coefficient is 105, instead of storing 105, you encode it as +5 (indicating an increase of 5).
- AC
  - **AC represents image details**: AC coefficients encode the fine details within an image block.
  - **Encoding AC**: AC coefficients are grouped into pairs, each pair consisting of (run, level). The "run" indicates how many consecutive zero coefficients came before the non-zero coefficient. The "level" represents the actual value of the non-zero coefficient.
  - **Run-Length Encoding**: This method efficiently compresses sequences with long runs of zeros, and it allows you to represent both zero and non-zero values compactly.
  - **Example**: Instead of storing a long string of zeros, you can encode it as (5 zeros, 10), meaning there are 5 consecutive zeros followed by a non-zero value of 10.

→ **Entropy Coder:**

- JPEG standard specifies two methods - Huffman and arithmetic coding.
- **Baseline sequential JPEG**: Huffman coding is used exclusively.
- Huffman coding requires that one or more sets of coding tables are specified by the application. The same table used for compression is used needed to decompress it.
- The baseline JPEG uses only two sets of Huffman tables – one for DC and the other for AC

Modes of operation in JPEG

- Baseline or sequential encoding
- Progressive encoding
    - Progressive scanning through spectral selection
    - Progressive scanning through successive approximation
- Hierarchical encoding
- Lossless encoding

Baseline encoding

- Baseline sequential coding is designed for images with 8-bit samples.
- Utilizes Huffman coding exclusively for entropy encoding.
- Each image block is encoded in a single left-to-right and top-to-bottom scan.
- Encodes and decodes 8x8 blocks with full precision one at a time.
- Supports interleaving of color components
- The sequence includes FDCT, quantization, DC difference calculation, and zig-zag ordering.
- Products claiming JPEG compatibility must include support for at least the baseline encoding system.

Progressive encoding

- Each block is encoded in multiple scans.
- Each scan follows zig-zag ordering, quantization, and entropy coding similar to baseline encoding, but takes much less time to encode and decode.
- Each scan contains only part of the complete information, leading to faster encoding and decoding.
- With the first scan, a basic image can be reconstructed, and successive scans refine image quality
- **Example:** web page image loading: you initially see a lower-quality image, but it progressively improves.

Progressive scanning through spectral selection

- **Initial Low-Frequency Scan:** The first scan transmits specific low-frequency DCT coefficients within each block.
- The image reconstructed at the decoder from the first scan appears blurred due to the absence of high-frequency details.
- Subsequent scans encode bands of coefficients with higher frequencies than the previous scan and so with each scan the image is enhanced further.

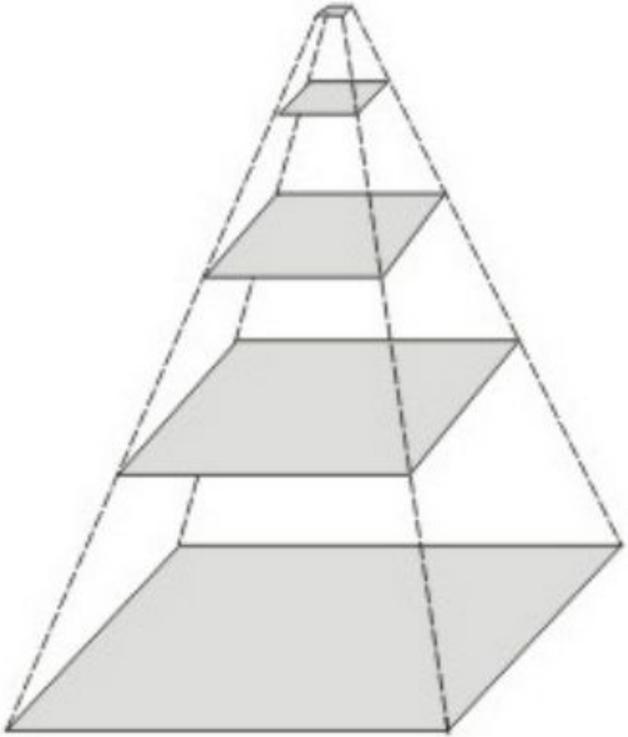
Progressive scanning through successive approximation

- Each scan encodes all coefficients within a block, but not to their full quantized accuracy.
- In the first scan, only the N most significant bits of each coefficient are encoded (N is specifiable).
- Successive scans add lower significant bits of coefficients until all bits are sent.
- The resulting reconstruction quality is good even from the early scans, as the high frequency coefficients are present from the initial scans.

Hierarchical encoding

- Pyramidal Structure: Image to be encoded is organized into a pyramidal structure with multiple resolutions.
- Layer Arrangement: Original image (finest resolution) is at the lowermost layer, and reduced resolution images are on upper layers.
- Resolution Reduction: Each layer decreases resolution with respect to the adjacent lower layer by a factor of two, in the horizontal, vertical, or both directions.
- Special Case of Progressive Encoding: Hierarchical encoding is akin to progressive encoding but with increasing spatial resolution between stages.

Hierarchical encoding

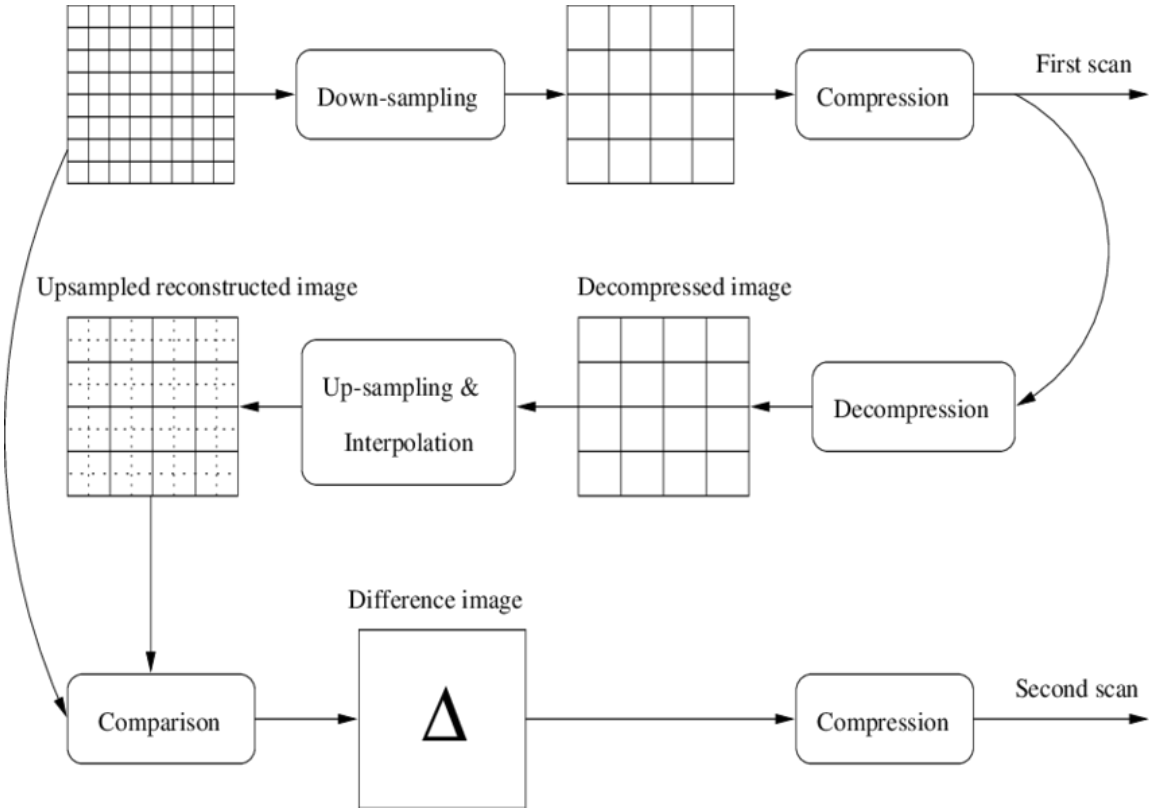


Hierarchical encoding

Steps Involved in Hierarchical Encoding:

1. **Resolution Reduction**: Obtain reduced resolution images starting with the original. Reduce resolution by a factor of two (horizontally, vertically, or both).
2. **Topmost Layer Encoding**: Encode the reduced resolution image from the topmost (coarsest) layer using baseline, progressive, or lossless encoding.
3. **Decode and Up-sample**: Decode the above image. Interpolate and up-sample it by a factor of two horizontally and/or vertically. Use this interpolated image as a prediction for encoding the next lower layer (finer resolution).
4. **Difference Encoding**: Encode the difference between the next lower layer and the predicted image using baseline, progressive, or lossless encoding.
5. **Repeat Stages**: Repeat the encoding and decoding steps until the lowermost layer (finest resolution) is reached.

Hierarchical encoding

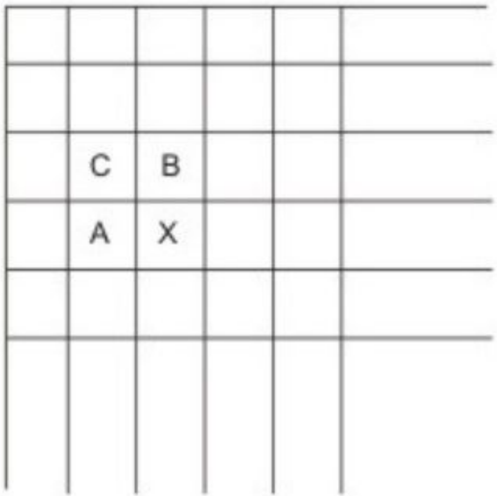


Lossless encoding

- **Predictive Coding**: Lossless encoding in JPEG employs a simple predictive coding mechanism. It doesn't use the FDCT + Entropy coder for encoding or the Entropy decoder + IDCT for decoding.
- **Quantization Elimination**: Unlike lossy encoding, which involves quantization, lossless encoding eliminates the quantization step to preserve all image details.
- While lossless encoding doesn't achieve as high compression ratios as lossy encoding, it's suitable for scenarios where preserving every bit of image data is essential.

- **Predictive Coding Mechanism**: Instead of using the 8x8 block structure, lossless encoding predicts each pixel based on three adjacent pixels. The prediction is based on one of eight possible predictor modes. An entropy encoder is then used to encode the predicted pixel obtained from the lossless encoder

Selection Value	Prediction
0	None
1	A
2	B
3	C
4	A+B-C
5	A+(B-C)/2
6	B+(A-C)/2
7	(A+B)/2



Explain MPEG compression process in detail.

MPEG compression removes two types of redundancies:

Spatial redundancy:

- Pixel values are not independent, but are correlated with their neighbors both within the same frame and across frames. So, to some extent, the value of a pixel is predictable given the values of neighboring pixels.
- It is removed with the help of DCT compression.

Temporal redundancy:

- Pixels in two video frames that have the same values in the same location (some objects repeated again and again in every frame).
- It is removed with the help of Motion compensation technique Macroblock
- Each macroblock is composed of four 8*8 Luminance (Y) blocks and two 8*8 Chrominance (Cb & Cr) blocks.
- This set of six blocks is called a macro block.
- It is the basic hierarchical component used achieving high level of compression.
- The key to achieving a high rate of compression is to remove much redundant information as possible.
- Entropy encoding and Huffman coding are two schemes used for encoding video information.
- MPEG takes the advantage of the fact that there exists a correlation between successive frames of moving pictures.

MPEG constructs three types of pictures namely:

- Intra pictures (I-pictures)
- Predicted pictures (P-pictures)
- Bidirectional predicted pictures (B-pictures)

The MPEG algorithm employs following steps:

Intra frame DCT coding (I-pictures):

The I-pictures are compressed as if they are JPEG images.

- First an image is converted from RGB color model to YUV color model.
- In general, each pixel in a picture consists of three components: R (Red), G (Green), and B (Blue).
- But (R, G, B) must be converted to (Y, Cb, Cr) in MPEG-1, then they are processed.
- Usually we use (Y, U, V) to denote (Y, Cb, Cr).

Apply DCT

DCT is performed on small blocks of 8*8 pixels to produce blocks of DCT coefficients.

The NXN two-dimensional DCT is defined as:

F(u,v)=2NC(u)C(v)∑x=0N−1∑y=0N−1f(x,y)cos(2x+1)uπ2Ncos(2y+1)vπ2N

$$F(u,v) = \frac{2}{N} C(u) C(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) \cos\frac{(2x+1)u\pi}{2N} \cos\frac{(2y+1)v\pi}{2N}$$

C(u),C(v)={12–√foru,v=0

$$C(u), C(v) = \{ \frac{1}{\sqrt{2}} for u, v = 0$$

=1otherwise

$$= 1 otherwise$$

The inverse DCT (IDCT) is defined as:

f(x,y)=2N∑u=0N−1∑v=0N−1C(u)C(v)cos(2x+1)2Ncos(2y+1)vπ2N

$$f(x, y) = \frac{2}{N} \sum_{u=0} N-1 \sum_{v=0} N-1 C(u) C(v) cos\frac{(2x+1)}{2N} cos\frac{(2y+1)v\pi}{2N}$$

Where x, y are spatial co-ordinates in the image block u, v are co-ordinates in the coefficient block.

Apply Quantization

- Quantization is process that attempts to determine what information can be safely discarded without significant loss in visual fidelity.
- MPEG uses a matrix called quantizer (Q[i,j]) to define quantization step. Every time when a pixels matrix (X[i,j]) with the same size to Q[i,j] comes ,use Q[i,j] to divide x(i,j) to get quantized value matrix Xq[i,j].
- Quantization Equation Xq[i,j] = Round(X[i,j)/Q[i,j])
- After Quantization, perform Zig-zag scanning to gather even more consecutive zeroes.
- Then various compression algorithms are applied including Run-length and Huffman encoding.
- In Huffman coding, we give shorter keywords to more frequently coefficients & longer keywords to least frequently occurring coefficients.
- Hence achieving final level of compression.

Motion-compensated inter-frame prediction (P-pictures):

- In most video sequences there is a little change in the contents of image from one frame to the next.
- Most video compression schemes take advantage of this redundancy by using the previous frame to generate a prediction of current frame.
- It removes temporal redundancy by attempting to predict the frame to be coded from previous frame.
- This is based on current value to predict next value and code their difference called as prediction error.
- Motion compensation assumes that current picture is some translation of previous frame.
- The frame to be compared is split in to blocks first and then best matching block is searched.
- Each block uses previous picture for estimating prediction.
- This search process is called as prediction.

Motion-compensated inter-frame prediction:

- By reducing temporal redundancy, P-pictures offer increased compression compared to I-pictures.
- Motion Estimation is to predict a block of pixel value in next picture using a block in current picture. The location difference between these blocks is called Motion Vector. And the difference between two blocks is called prediction error.
- In MPEG-1, encoder must calculate the motion vector and prediction error. When decoder obtains this information, it can use this information and current picture to reconstruct the next picture. We usually call this process as Motion Compensation.

B-frame (Bidirectional predictive frame):

-Frames can also be predicted from future frames. Such frames are usually predicted from two directions, i.e. from the I- or P-frames that immediately precede or follow the predicted frame.

-These bidirectionally predicted frames are called B-frames. A coding scheme could, for instance, be IBBPBBPBBPBB.

-B-pictures uses the previous or next I-frame or P-frame for motion compensation and offers the highest degree of compression.

Each block in a B-picture can be forward, backward or bidirectionally predicted.

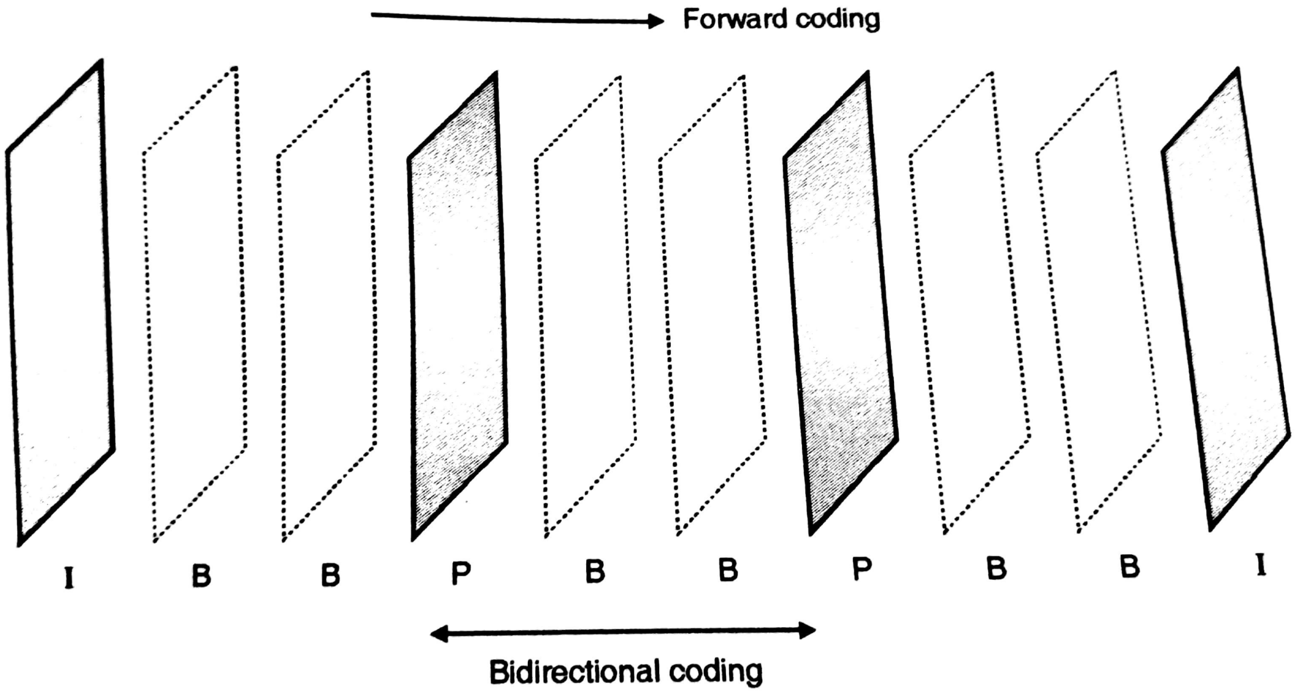
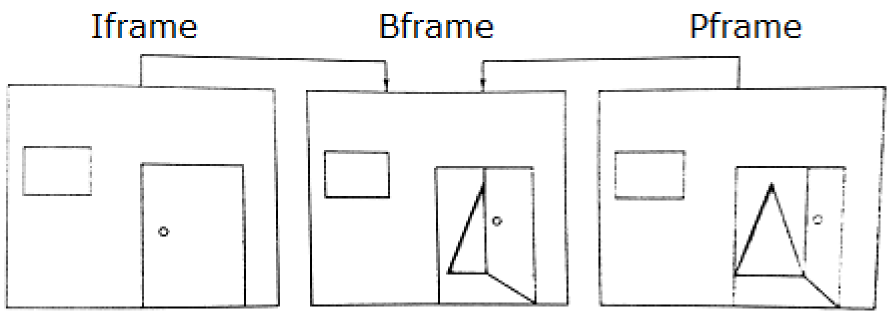


Fig.: Illustrated use of three picture types in a typical GOP

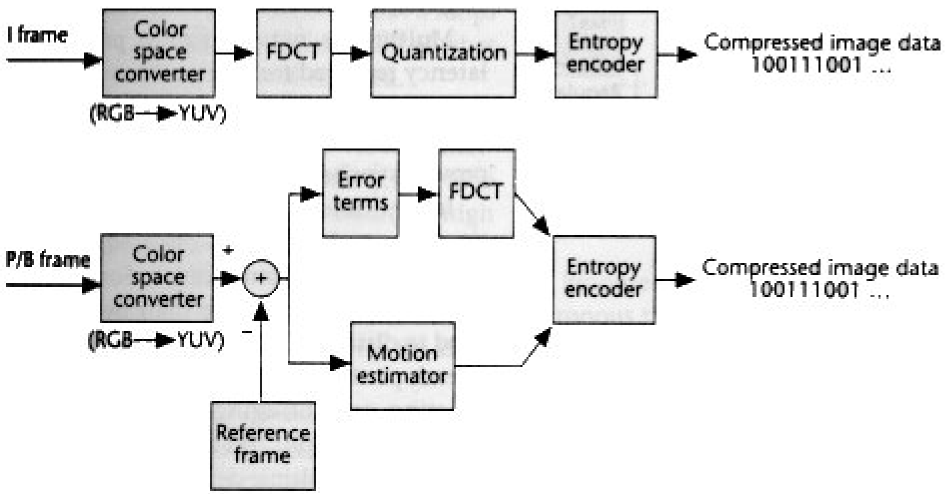


I-, P- and B-frames

Bidirectional predicted pictures (B):

- Bidirectional predicted pictures utilize three types of motion compression techniques.
- Forward motion compensation - uses past picture information.
- Backward motion compensation - uses future picture information .
- Bidirectional compensation - uses the average of the past and future picture information.

MPEG encoder:



It's our last ever working day today :-(

ug2020-cse (ug2020-cse@ssn.edu.in)

It's our last ever working day today :-(

Hypermedia

Messaging

--

Thanks and Regards,

Adithi Shankar

Thandav Krishna

CSE - A

Sans Serif

Send

What is Hypermedia Messaging?

- Hypermedia messaging is a communication approach that combines elements of hypermedia and messaging to create interactive and dynamic communication experiences in the digital realm
- **Hypermedia** combines text, images, audio, video, and hyperlinks in digital systems, enabling non-linear, interactive content with clickable links for easy access to related resources and actions.
- **Messaging** in communication tech entails sharing text, media (like images, audio, and video), and interactive content among users or systems, spanning instant messaging, email, chat apps, and beyond.
- **Hypermedia messaging** combines both, enabling dynamic, interactive conversations with multimedia and links. It enhances communication by offering context and interactive possibilities, ideal for engaging and sharing information in critical applications.

Importance of Interactive and Dynamic communication in the digital age

1. **Enhanced Engagement**: Interactive and dynamic communication captivates audiences, keeping them engaged, as opposed to static, one-way communication.
2. **Immediate Feedback**: It allows for real-time feedback, facilitating rapid responses and adjustments, crucial in a fast-paced digital environment.
3. **Personalization**: Dynamic content can be tailored to individual preferences, delivering a more personalized and relevant user experience.
4. **Information Richness**: Interactive media and dynamic content convey information more effectively, enhancing comprehension and retention.
5. **Competitive Advantage**: Organizations that adopt interactive and dynamic communication gain a competitive edge by staying relevant and appealing to modern audiences.
6. **Adaptability**: In a rapidly changing digital landscape, dynamic communication can adapt to evolving technologies and user expectations, ensuring continued effectiveness.

Mobile Messaging

Mobile messaging refers to messaging platforms that include E-mail, SMS, instant messaging apps, which have become central to modern communication. They support multimedia content and interactive features.

- Key mobile messaging platforms include SMS, instant messaging apps, and mobile email clients, which have become central to modern communication. They support multimedia content and interactive features.

New Standards for Interplay Among Communication Media Types

- **UMA (Unlicensed Mobile Access):** UMA is a standard that enables seamless handover of voice and data services between cellular and Wi-Fi networks. It allows mobile devices to switch between cellular and Wi-Fi connections without interrupting voice calls or data sessions. UMA promotes uninterrupted communication and helps offload network traffic to Wi-Fi, improving network efficiency.
- **X.400:** X.400 is a messaging standard developed by the International Telecommunication Union (ITU). It defines a set of protocols for email and messaging systems, ensuring interoperability between different email systems. While it's less commonly used for personal messaging, X.400 is essential for secure and standardized communication in business and government sectors.

Hypermedia Message Components

1. Text Messages
2. Rich-Text Messages
3. Voice Messages
4. Full-Motion Video

Text Messages

- The earliest messaging systems used a limited subset of plain ASCII text.Based initially on teletype technology and later on used as operating system-supported messaging applications.
- New messaging standards have added new capabilities such as class of service,delivery reports(blue-ticks on whatsapp), time stamps and so on.
- Some systems provide extended reporting capabilities as when the recipient actually opened and read the message.

Rich-Text Messages

- Microsoft defined a standard for exporting and importing text data that include character set, font-table, section and paragraph formatting, document formatting and color information called Rich-Text Format(RTF).
- This format is used for storage, import and export of text files across a variety of word-processing and messaging systems. When sections from one document are cut and pasted in another the font and formatting information is retained.
- RTF facilitates to create messages in one word processor and edit it in another application to display the text in nearest equivalent fonts and formats.
- RTF carries format information that includes character sets, font and color tables; document, section, paragraph, general and character formatting; and specialized characters.

Extensions to RTF

- RTF is further extend in two ways (bit-maps, images, icons, and so on)
    - By adding graphics
    - By adding file-attachments
- Bitmaps can be in any of the standard graphic formats such as Windows metafile, Windows bitmaps, TIFF, PCX. It may contain a complete image by itself,a representation of an attachment, or an embedded or linked object. The representation. This representation is in form of icon or button.
- Clicking on the icon with a mouse allows retrieving attachments, or launches the authoring tool or the server application for retrieving and rendering or editing linked objects.

Voice Messages

- The invention of telephone allowed communications among people dispersed geographically.Over the years telephone became an essential tool for business functions.
- Answering machines have solved the problem of recipient not being able to answer the calls. Late Voicemails have replaced the answering machines, voicemails recorded the message in the case of recipient cannot answer after a certain number of rings.
- Hypermedia messaging systems extend the concept of voicemail to voice messages that are linked in text-based messages.
- From the perspective of a computer there is no difference between a human recorded voice or recorded music other than the quality of sound reproduction.

Voice Messages - Music

- The Musical Instrument Digital Interface (MIDI) was developed initially by the music industry to allow computer control of musical recordings and instruments .
- MIDI interfaces are being used for a variety of peripherals, including digital pianos, digital organs, video games with high fidelity sound output.
- From a hyper media perspective, whether the object is voice or music, it is stored in compressed form on an object server.
- An integrated messaging system allows embedding or linking the music file in MIDI format to the e-mail message.

Full-Motion Video Management

It is easier to watch a video over reading a large document if both of them contain the same information.

- Full-motion video management is a comprehensive system or approach used to handle, process, and maintain video content that captures full-motion, real-time visuals. It is widely employed in various industries, including security and surveillance, entertainment, education, and more

Full Motion Video Authoring Systems

The needed tools for creation and editing of multimedia objects are. The video capture program should contain

- Fast and simple capture of digital video from analog sources such as video camera or videotape
- Compression and decompression interfaces as the video is being captured
- A video editor with the ability to decompress, combine, edit, and compress clips.
- Video indexing and annotation software for marking sections of a video clip and recording annotations
- Identifying and indexing video clips for storage

Full-Motion Video Playback Systems

Eg: VLC media player, Windows media player

These are as important as the authoring systems. They detach the embedded video reference object, interpret its contents, and retrieve the actual video clip from a specialized video server and launch the playback application.

A number of factors are involved in playing back the video correctly.

- Compression format used for storing the video clip relates to the available hardware and software facilities of decompression.
- Resolution of the screen and the system facilities
- CPU processing power and the expected level of degradation

Video for Windows (VFW)

- Microsoft Windows is the most commonly used environment for multimedia messaging. Initially VFW established new components for data interchange, such as common file format for video called the audio visual interleaved(AVI).
- VFW provides capture, edit, playback tools for full motion videos.
- The tools provided by VFW are:
    - The VidCap tool, designed for fast video capture
    - The VidEdit tool, designed for decompression, editing, and compression full-motion digital video
    - The VFW playback tool

VFW takes advantage of the key elements of Windows such as Object Linking and Embedding(OLE) and Dynamic Data Exchange(DDE). VFW provides developers with the ability to add full-motion video to any Windows based application.

Apple’s QuickTime

Apple's QuickTime is a multimedia framework and media player developed by Apple Inc. It has historically been used for playing, creating, and streaming audio and video content on Apple's macOS and Windows platforms. QuickTime supports a wide range of codecs and file formats, making it a versatile tool for multimedia applications. However, Apple has gradually phased out QuickTime support on Windows, and in recent macOS versions, it has been replaced by newer technologies. Despite its decreasing prominence, QuickTime has had a lasting impact on multimedia playback and authoring in the digital realm.

Intel's Indeo

- **Codec Technology**: Indeo is a video codec technology used for compressing and decompressing digital video files, primarily in the 1990s and early 2000s.
- It reduces the size of video files through successive compression methodologies, including YUV subsampling, vector quantization, Huffman's run-length encoding and variable content encoding.
- It takes advantage of Intel i750 video processor if it is available in the system.
- It determines the hardware available and optimizes the playback for the hardware by controlling the frame rate.
- The compressed file must be decompressed for playback, it is done dynamically during the playback process.
- It is also provided with VFW

Hypermedia Linking and Embedding

- Linking in Hypertext Documents
  - Linking and Embedding
  - Linking Objects
  - Embedding Objects
  - Design Issues

Hypermedia Linking and Embedding

1. Linking as in hypertext applications. Hypertext systems associated keywords in a document with other documents.
2. Linking multimedia objects is stored separately from the document and the link provides a pointer to its storage. An embedded object is a part of the document and it retrieved when the document is retrieved.
3. Linking and embedding in a context specific to Microsoft Object Linking and Embedding

When a multimedia object is incorporated in a document, its behaviour depends on whether it is **linked** or **embedded**.

Linking Objects

When an object is linked, the source data object, called the **link source** continues to reside wherever it was at the time the link was created. This maybe at the object server where it was created, or where it may have been copied in a subsequent replication.

- Link can contain information about the multimedia object storage, its presentation parameters, and the link reference is transferred, but the actual multimedia document remains at its original locations.
- Note that linked object is not part of the hypermedia document and does not take up the storage space within the hypermedia document.

Embedded Objects

When the multimedia object is embedded, a copy of the object is physically stored in the hypermedia document. In addition, presentation information and the information about the server application that can display/play or edit is also stored.

- Any changes made in the embedded object will not be reflected in the other copies
- Graphics and images can be inserted in a rich-text document or embedded using OLE techniques.
- Voice and audio components can be included in a text message.

Design Issues of Linking and Embedding

Under a distributed environment, OLE creates significant headaches for users if there is incomplete link tracking between documents that have been mailed between PCs and the applications which created the objects.

- Users need robust link tracking across the distributed environment.

Hypermedia Linking and Embedding refer to ways in which different pieces of information, such as text, images, or multimedia objects, can be connected within documents or applications. Let's break down the key concepts in simpler terms:

- Linking in Hypertext Documents:
  - In hypertext, words or keywords in a document are connected to other documents. Clicking on a link takes you to the linked document.
  - Multimedia objects, like images or videos, can be linked in a similar way. The link acts like a pointer to where the multimedia object is stored.
- Linking vs. Embedding:
  - Linking Objects:
    - When an object is linked, it stays in its original location (where it was created or copied).
    - The link contains information about the object's storage and presentation parameters.
    - The actual multimedia document remains at its original location and is not part of the hypermedia document.
  - Embedded Objects:
    - When an object is embedded, a copy is stored in the hypermedia document itself.
    - Presentation information and details about the application that can display or edit the object are also stored.
    - Changes made to the embedded object don't affect other copies, and the object is part of the hypermedia document.
- Design Issues of Linking and Embedding:
  - Linked Objects:
    - The link contains information about where the multimedia object is stored.
    - Linked objects don't take up storage space within the hypermedia document.
  - Embedded Objects:
    - A copy of the object is stored in the hypermedia document.
    - Changes to the embedded object won't affect other copies.
    - Different types of media, like graphics or audio, can be embedded.
- Challenges in a Distributed Environment:
  - In a distributed environment (across different computers), issues can arise with link tracking.
  - OLE (Object Linking and Embedding) can create problems if links between documents aren't properly tracked, especially when documents are shared between different computers.

In simpler terms, linking is like pointing to where something is stored, while embedding is like putting a copy directly into a document. Both have their uses, and in a distributed setting, proper link tracking is crucial to avoid issues.

Creating Hypermedia Messages

By definition, a hypermedia message can be a complex collection of a variety of objects. While an ordinary text message includes only text, and possibly some input from a spreadsheet, hypermedia may require several more steps for completion namely:

1. Planning
2. Creating each component
3. Integrating components

Here are some key points explained in simpler terms:

VIM Interface:
- It's like a set of rules that allow different email systems and apps to communicate.
- Developers use the VIM interface to create apps that understand and use email.

Store-and-Forward Method:
- Messages are sent and stored temporarily before reaching the recipient.
- Each system using VIM has containers where messages are kept.

Address Books:
- Apps using VIM have address books with info about users, groups, and apps.
- These address books also know where the message containers are for each user or app.

Messaging Sequence:
- Sender uses VIM to create a message, finds the recipient's address in the address book, and sends the message.
- The messaging system takes care of delivering the message to the right place.
- Receiver checks their container for new messages, reads them, and decides what to do with each.

VIM Messages:
- Messages have types, like email messages.
- They have headers with info like sender and recipient addresses.
- Messages can contain different types of information, like text, pictures, and even other messages.

Mail Messages:
- A type of message with specific rules.
- Can include different parts like text, pictures, and attachments.

Message Delivery:
- If a message is delivered successfully, a report is sent to the sender.
- If a message can't be delivered, a report is sent to the sender.
- Messages stay marked as unread until the receiver's app checks for new messages.

Distinguished Names:
- Each user or app has a unique name.
- This name helps authenticate the user or app during communication.
- The address book keeps all these unique names.

VIM Services:
- VIM provides services like creating and sending messages.
- It helps with composing messages, sending and receiving them, and managing the address book.

Vendor Independent Messaging (VIM)

- VIM interface is designed to facilitate messaging between VIM enabled electronic mail systems as well as other applications ie it is implemented as an API.
- It allows developers to provide **mail-aware** and **mail-enabled** applications.
- VIM makes use of of communication using the **store-and-forward** method.
- Messages are delivered to a **container** and each VIM associated system contains one or more of these containers.
- VIM-aware applications also use **address books** to store information about users, groups, applications etc.
- These address books contain information about the message containers of the users, groups and applications referenced in the book.

VIM Messages

- VIM defines messaging as a store-and-forward method of application-to-application or program-to-program data exchange.
- Here is a typical messaging sequence:
  - Sender
    - Uses VIM interface to construct a message.
    - Uses VIM address book to determine the address of the receiver.
    - The message and the address is sent to the messaging system.
    - The messaging system assumes responsibility for routing and delivering the message.
  - Receiver
    - The receiver notices the presence of new messages in its container.
    - Uses VIM interface to read the message.
    - It decides whether to delete, store in container or extract and store elsewhere.

VIM Messages Definition

- Each message has a message type associated with it which defines the **syntax** of the message and the type of information that it can contain.
- For example a **mail message** is a type of message.
- VIM messages contain a **header** at minimum. In addition it may contain one or more **message items**.
- The header consists of attributes like recipient address, originator address etc.
- A message item is an arbitrary-sized data of a defined type.
- A message may also contain file attachments.
- VIM also allows nesting of messages ie a message within another message.
- A VIM message can also be digitally signed.

VIM Mail Message

- A mail message is a message of a well-defined type that must include a message header and may include note parts, attachments and other application defined components.
- Note parts may include texts, bitmaps, pictures, sound, video etc.

VIM Message Delivery

- On successful delivery of a message, a delivery report is generated and sent to the sender if the sender requested for the report.
- If the message cannot be delivered a non-delivery report is sent to the sender.
- A message delivered to a message container remains marked unread until an application calls VIMOpenMessage().
- A receipt is sent to the sender once the message has been opened if the sender requested for it.

VIM Distinguished Names

- The concept of distinguished names is similar to that of X500.
- The name is a unique identity and a client provides its distinguished name for authentication when a session is created.
- A recipient can be addressed based on name and by address.
- The name corresponds to the distinguished name while the address corresponds to the recipient’s message container.
- The address book is the repository for all the distinguished names.

VIM Services

- The VIM interface provides a number of services for creating and mailing a message namely
  - Electronic message composition and submission.
  - Electronic message sending and receiving.
  - Message extraction from mail system.
  - Address book services.

MAPI, or Messaging Application Programming Interface, is a system in Windows that helps different applications talk to each other and to messaging services. Instead of just providing a way for apps to send messages, MAPI creates a structure that separates applications from the underlying messaging services. Its main goals are to keep apps separate from messaging services, make sending basic messages a standard feature for all apps, and support workgroup apps that rely on messaging.

MAPI Architecture:
• MAPI has two parts: one for client apps and one for messaging services.
• The client part connects apps to MAPI, and the service part links MAPI to the messaging system.
• This setup lets any messaging app use any messaging service that has a MAPI driver.

Telephony API:
• Telephony, like using the telephone for reading emails with speech recognition, is part of messaging.
• TAPI (Telephony API) is a standard created by Microsoft and Intel for integrating telephony with messaging.

X400, X500, and Internet Messaging:
• X400 and X500 are standards for message handling and directory systems.
• Internet Messaging uses MIME (Multipurpose Internet Mail Extensions) to include different types of content in messages, like images or videos.

Integrated Document Management:
• Some messaging systems, like Lotus Notes, allow users to attach, embed, or link various types of multimedia content.
• Multimedia Object Server and Mail Server Interactions: Mail servers store email messages and references to multimedia, not the actual multimedia. In smaller groups, the same resources can be shared for storing both multimedia and email files.

In simple terms, MAPI helps apps and messaging services work together, telephony can be part of messaging, there are standards for message handling, and some messaging systems let you include different types of content in your messages.

MAPI Support

- The focus of MAPI is to provide a messaging architecture instead of just providing a messaging API in Windows.
- MAPI provides a layer of functionality between applications and underlying messaging systems.
- The primary goals of MAPI include:
  - Separate client applications from the underlying messaging services.
  - Make basic mail-enabling a standard feature for all applications.
  - Support messaging-reliant workgroup applications.

MAPI Architecture

- The MAPI architecture provides two perspectives: a client API and a service provider interface.
- The client API provides the link between the client application and MAPI.
- The service provider interface links the MAPI and the messaging system.
- The two interfaces combine to provide an open architecture such that any messaging application can use any messaging service that has a MAPI driver.

Telephony API

- Telephony is not often considered an integral part of a message-handling system until one views it from the perspective of the telephone being an integral component of the overall messaging interface for the user.
- The telephone can be used for “reading” email using speech recognition remotely.
- The TAPI standard is a perfect example. It has been developed by Microsoft and Intel.

1. X.400 Standard:
• What it is: X.400 is a standard that defines the rules for electronic messaging, particularly email.
• How it works: It provides guidelines for how different email systems should communicate with each other. It covers things like addressing, message formats, and how messages are delivered between different email servers.
• Key Points:
• X.400 ensures that emails can be sent and received across different email systems, even if they're from different vendors or providers.
• It's a set of rules to make sure that the information in an email is understood universally.

2. X.500 Standard:
• What it is: X.500 is a standard that defines how directory services should work in a network.
• How it works: It lays out the guidelines for creating a directory service, which is like an organized list of information about users, resources, and other things in a network. This helps in finding and accessing information efficiently.
• Key Points:
• X.500 provides a way to organize and store information in a directory, making it easy to search for and retrieve specific details.
• It's often used for things like user authentication, where a system needs to verify the identity of a user before granting access to certain resources.
• X.500 directories are hierarchical, like a tree structure, making it easy to navigate and locate information.

In simple terms, X.400 is about how emails are sent and received, making sure they work across different systems. On the other hand, X.500 is about organizing and finding information in a network, often used for things like user authentication. Both standards help different parts of a computer network work together smoothly.
Certainly, let's break down X.400 using the provided key terms:

1. Message Store:
• Definition: A place where messages are stored, typically on a mail server or a user's mailbox.
• In X.400: X.400 defines how messages are stored and managed in a message store. It ensures that messages are stored securely and can be retrieved when needed.

2. Message:
• Definition: A unit of information sent from one person or system to another, typically in electronic form.
• In X.400: X.400 sets the standards for the format and structure of messages. It defines how messages should be composed, including headers, body content, and any attachments.

3. Delivery Notification:
• Definition: A confirmation that a message has been successfully delivered to the recipient.
• In X.400: X.400 supports delivery notifications, allowing the sender to receive confirmation that the message reached the recipient's mail system. This ensures the sender knows the message was successfully delivered.

4. Receipt Notification:
• Definition: A notification sent to the sender to confirm that the recipient has opened or read the message.
• In X.400: X.400 includes provisions for receipt notifications. When enabled, the sender can receive a notification indicating that the recipient has opened or read the message.

5. Sender MTA (Message Transfer Agent):
• Definition: Software responsible for transferring messages from the sender to the recipient.
• In X.400: X.400 defines the Sender MTA as the component responsible for initiating the transfer of messages from the sender's system to the recipient's system.

6. Receiver MTA (Message Transfer Agent):
• Definition: Software responsible for receiving and delivering messages to the recipient's message store.
• In X.400: X.400 defines the Receiver MTA as the component responsible for receiving messages from the Sender MTA and delivering them to the intended recipient's message store.

7. Query:
• Definition: A request for information or a search operation.
• In X.400: X.400 supports queries, allowing users or systems to search for specific messages or information within the messaging system. This can be useful for finding and retrieving specific messages efficiently.

In summary, X.400, within the context of these key terms, provides standards for how messages are stored, delivered, and managed in a messaging system, ensuring reliable communication between Sender MTAs and Receiver MTAs with features like delivery and receipt notifications. Additionally, it supports queries for efficient information retrieval.

Certainly, let's explore X.500 in the context of the provided key terms:

1. DSA (Directory System Agent):
• Definition: A Directory System Agent is a component responsible for managing and providing access to directory information in an X.500 directory system.
• In X.500: X.500 defines DSAs as servers that store and retrieve directory information. DSAs handle queries from DUAs (Directory User Agents) and ensure the integrity and security of the directory data.

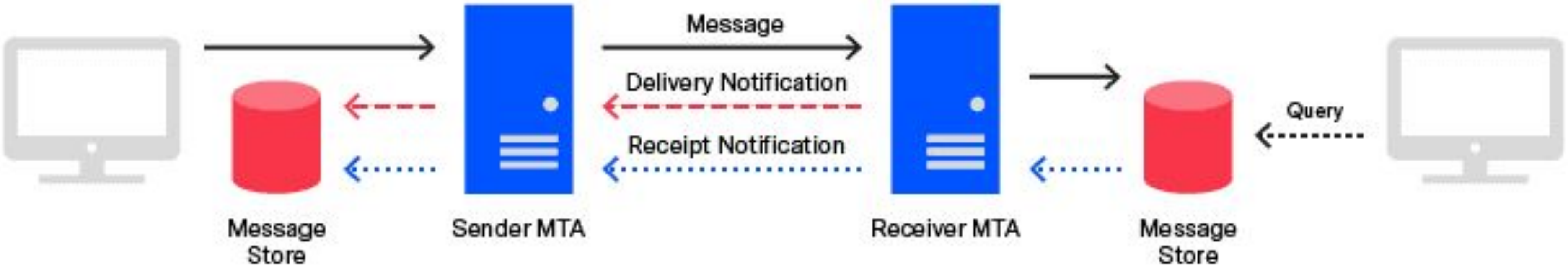
2. DUA (Directory User Agent):
• Definition: A Directory User Agent is a client application or interface used by individuals or systems to interact with a directory service.
• In X.500: X.500 defines DUAs as the clients that connect to DSAs to search for, retrieve, or update directory information. DUAs facilitate user interaction with the directory system.

In summary, X.500 is a standard that defines a directory service architecture, and within this framework:

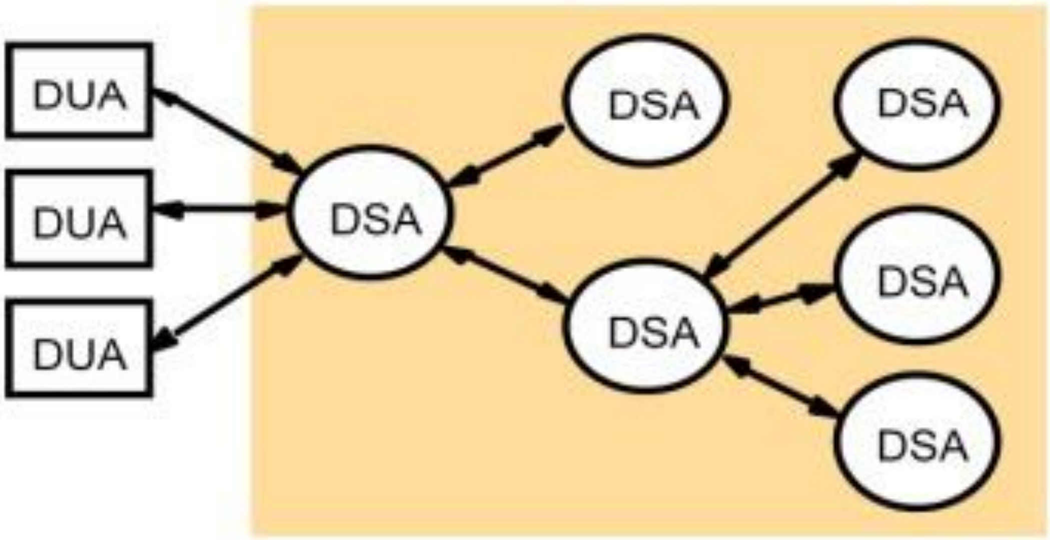
• DSA (Directory System Agent): This is the server component responsible for managing and providing access to directory information. DSAs handle the storage and retrieval of data in the directory system.

• DUA (Directory User Agent): This is the client application or interface used by individuals or systems to interact with the directory service. DUAs are the users' tools for querying and updating directory information through DSAs.

X400 Message Handling Service



X500 Directory System Standards



Internet Messaging

- Here we discuss the Multipurpose Internet Mail Extensions (MIME).
- This specification defines mechanisms for generalising the message content to include multiple body parts and multiple data types.
- The additional functionalities that it provides include:
    - A MIME version header file that distinguishes MIME messages from text-only single-body-part messages.
    - A content-type header field that describes the type and representation of the data in the body parts.
    - A content-transfer encoding methodology to allow non-MIME intermediate hosts to pass messages through their mail transport mechanisms.

Integrated Document Management

- **Integrated Document Management for Messaging:** Specialised messaging systems like Lotus Notes provide this service. This means that users can attach, embed or link a variety of multimedia content. It also allows forwarding of messages.
- **Multimedia Object Server and Mail Server Interactions:** Mail servers is used to store all email messages. It also contains references to multimedia in the form of links and not the actual multimedia object. It is possible in smaller groups to share the same physical resources for storing multimedia content as well as mail files