Question-and-solution

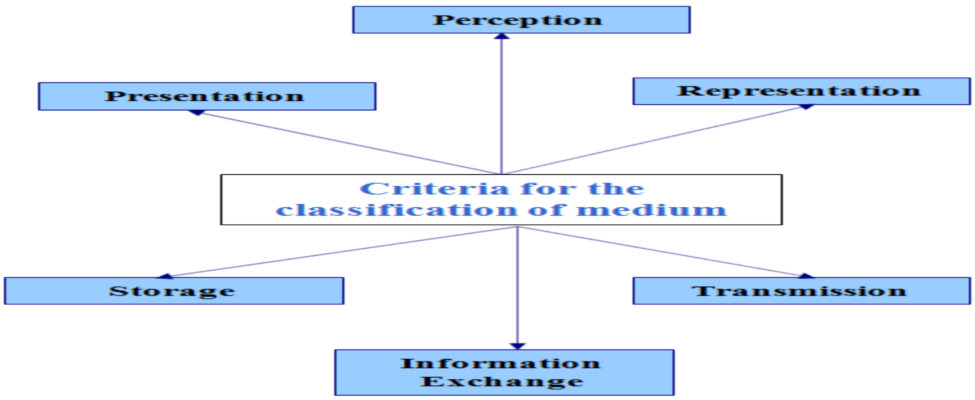
Multimedia System (Pokhara University)

QNO 1.1 What is a multimedia? Explain the multimedia system with suitable practical examples?

Multimedia- Definitions

- Any kind of system that supports more than one kind of media
- Multimedia means integration of continuous media (e.g. audio, video) and discrete media (e.g. text graphics , images) through which digital information can be conveyed to the user in appropriate way.
  - *Multi* - many; much; multiple
  - *Medium* - a interleaving substance through which something is transmitted or carried on

Classification based on perception (text, audio, video) is appropriate for defining multimedia



Time always takes separate dimension in the media representation

- Based on time-dimension in the representation space, media can be
  - Time-independent (*Discrete*)
    - Text, Graphics
  - Time dependent (*Continuous*)
    - Audio, Video
    - Video, sequence of frames (images) presented to the user *periodically*.
    - Time dependent a periodic media is not continuous!!
  - Discrete & Continuous here has no connection with internal representation!! (Relates to the viewers impression…)
- Multimedia is *any* combination of *digitally manipulated* text, art, sound, animation and video.
- A more strict version of the definition of multimedia do not allow just any combination of media.
  - It requires
    - Both continuous & discrete media to be utilized
    - Significant level of independence between media being used
- Multimedia is interactive when the end-user is allowed to control what and when the elements are delivered.
- Interactive Multimedia is Hypermedia, when the end-user is provided with the structure of linked elements through which he/she can navigate.
- Multimedia is *linear*, when it is not interactive and the users just sit and watch as if it is a movie.
- Multimedia is nonlinear, when the users are given the navigational control and can browse the contents at will.

Multimedia System

- Following the dictionary definitions, *Multimedia system is any system that supports more than a single kind of media*
  - Implies any system processing text and image will be a multimedia system!!!
  - Note, the definition is quantitative. A qualitative definition would be more appropriate.
  - The kind of media supported should be considered, rather the number of media
- A multimedia system is characterized by computer-controlled, integrated production, manipulation, storage and communication of independent information, which is encoded at least through a continuous (time-dependent) and a discrete (time-independent) medium.

Properties of Multimedia System:

The uses of term multimedia are not every arbitrary combination of media. Justify.

1. **Combination of media:** A simple text processing program with in corporate image is often called a multimedia application. Because two media are processed through one program. But one should talk multimedia only when both continuous and discrete media are utilized. So text processing program with incorporated images is not a multimedia application.
2. **Computer support integrated->** computer is idle tools for multimedia application
3. **Independence:** An important aspect of different media is their level of independence from each other.

Multimedia- Applications

Multimedia plays major role in following areas

- Instruction
- Business
    - Advertisements
    - Training materials
    - Presentations
    - Customer support services
- Entertainment
    - Interactive Games
- Enabling Technology
    - Accessibility to web based materials
    - Teaching-learning disabled children & adults
- Fine Arts & Humanities
    - Museum tours
    - Art exhibitions
    - Presentations of literature

QNO1.2 Explain the importance of medium in multimedia system. Explain different types of mediums in detail.

Multimedia System: one way of defining multimedia can be found in the meaning of composed world multi-many, much multiple.

Medium: An intervening substance through which some thing is transmitted or carried on.

Computer System medium:

1. Text
2. Image
3. Sound
4. Video

Representation Dimension of media:

Media are divided into two types in respect to time in their representation space:

1. Time independent (discrete): Information is expressed only in its individual value. E.g.: text, image, etc.
2. Time dependent (continuous): Information is expressed not only it's individual value, but also by the time of its occurrences. E.g.: sound and video.

A multimedia system is characterized by computer-controlled, integrated production, manipulation, storage and communication of independent information, which is encoded at least through a continuous (time-dependent) and a discrete (time-independent) medium.

Multimedia Systems

A **Multimedia System** is a system capable of processing multimedia data and applications.

A **Multimedia System** is characterised by the processing, storage, generation, manipulation and rendition of Multimedia information.

Classification of Media:

1. The perception media
2. The representation Media
3. The Presentation Media
4. The storage media

5. The transmission media
6. The information Exchange media

Perception media: Perception media help human to sense their environment. The central question is how human perceive information in a computer environment. The answer is through seeing and hearing.

Seeing: For the perception of information through seeing the usual such as text, image and video are used.

Hearing: For the perception of information through hearing media such as music noise and speech are used.

Representation media: Representation media are defined by internal computer representation of information. The central question is how the computer information is coded? The answer is that various format are used to represent media information in computer.

i. Text, character is coded in ASCII code
ii. Graphics are coded according to CEPT or CAPTAIN video text standard.
iii. Image can be coded as JPEG format
iv. Audio video sequence can be coded in different TV standard format(PAL, NTSC,SECAM and stored in the computer in MPEG format)

Presentation Media: Presentation media refer to the tools and devices for the input and output of the information. The central question is, through which the information is delivered by the computer and is introduced to the computer.

Output media: paper, screen and speaker are the output media.

Input Media: Keyboard, mouse, camera, microphone are the input media.

Storage media: Storage Media refer to the data carrier which enables storage of information. The central question is, how will information be stored? The answer is hard disk, CD-ROM, etc.

Transmission media: Transmission Media are the different information carrier that enables continuous data transmission. The central question is, over which information will be transmitted? The answer is co-axial cable, fiber optics as well as free air.

Information exchange media: Information exchange media includes all information carrier for transmission, i.e. all storage and transmission media. The central question is, which information carrier will be used for information exchange between different places? The answer is combine uses of storage and transmission media. E.g. Electronic mailing system.

Usages of Multimedia Application:

1. Education
2. Training
3. Entertainment
4. Advertisement
5. Presentation
6. Business Communication
7. Web page Design

QNO1.3 Explain data stream characteristics for continuous media with practical examples

Data streams

- *Data Stream* is any sequence of individual packets transmitted in a time-dependent fashion
  - Packets can carry information of either continuous or discrete media

Traditional data streams characteristics:

A sequence of individual packets transmitted in time dependent fashion is called data stream. The data stream will be used as a synonym data flow.

Transmission of information carrying different media leads to data stream with very different features. The attributes of synchronous, asynchronous and isochronous data transmission conform the field of computer communication and switching.

i. Asynchronous Transmission mode:

- ➔ The asynchronous transmission mode provides for communication with no timely restriction.
- ➔ Packets reach the receivers as fast as possible.

➔ All information of discrete media can be transmitted as asynchronous data stream. If an asynchronous mode is chosen for transmission of continuous media, additional technique most be applied to provide the time restriction. E.g.: Ethernet, protocol of worldwide internet for e-mail transmission.

ii. **Synchronous Transmission mode:**
- Defines maximum end-to-end delay
- Packets can be received at an arbitrarily earlier time
- For retrieving uncompressed video at data rate 140Mbits/s & maximal end-to-end delay 1 second the receiver should have temporary storage 17.5 Mbytes

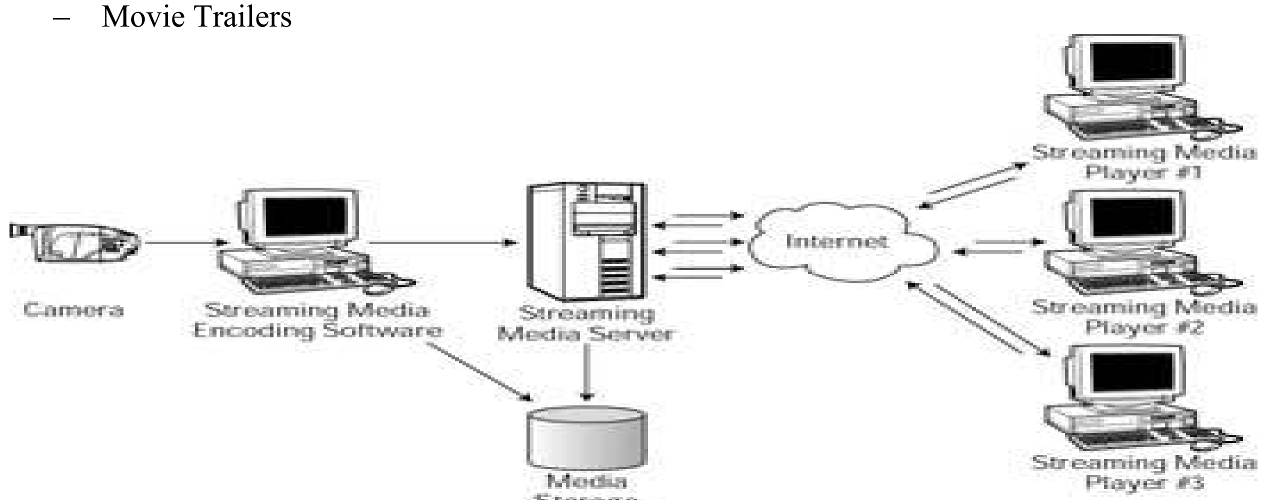
iii. **Isochronous Transmission mode:**
- Defines maximum & minimum end-to-end delay
- Storage requirements at the receiver reduces

Characterizing continuous media streams

- Periodicity
  - Strongly periodic
    - PCM coded speech in telephony behaves this way.
  - Weakly Periodic
  - Aperiodic
    - Cooperative applications with shared windows
- Variation of consecutive packet size
  - Strongly Regular
    - Uncompressed digital data transmission
  - Weakly regular
    - Mpeg standard frame size I:P:B is 10:1:2
  - Irregular
- Contiguous packets
  - Are contiguous packets are transmitted directly one after another/ any delay?
  - Can be seen as the utilization of resource
  - Connected / unconnected data streams

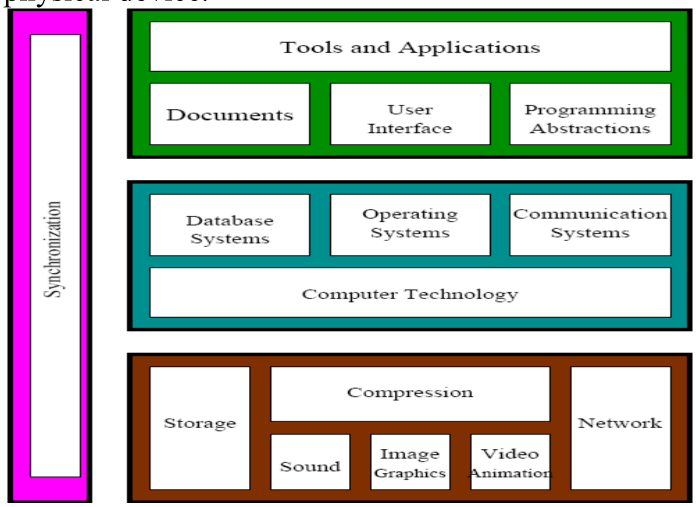
Streaming Media

- Popular approach to continuous media over the internet
- Playback at users computer is done while the media is being transferred (no waiting till complete download!!!)
- You can find streaming in
  - Internet radio stations
  - Distance learning
  - Cricket Live!!!
  - Movie Trailers



QNO1.4 What is Global Structure of multimedia? Draw its block diagram. Explain all its domain?

- *Application domain* — provides functions to the user to develop and present multimedia projects. This includes *Software tools*, and multimedia projects *development methodology*.
- *System domain* — including all supports for using the functions of the device domain, e.g., operating systems, communication systems (networking) and database systems.
- *Device domain* — basic concepts and skill for processing various multimedia elements and for handling physical device.

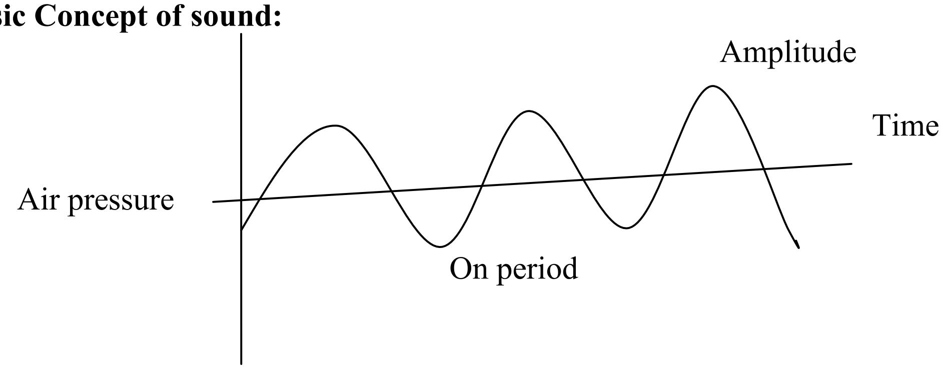


QNO2.1. Explain the basic concept of sound? How can you represent using computer?

Sound: Sound is a physical phenomenon produce by vibration of matter such as violin string or block of wood. As a matter vibrates pressure vibration are created in the air surrounding it.

The alternation of high and low pressure is propagated through air in a wave like motion. When a wave reaches a human air a sound is heard.

Basic Concept of sound:



Sound is produced by vibration of matter. During the vibration the pressure vibration are created in the air surrounding it. The pattern of oscillation is called wave form.

The wave form repeats the same shape at regular interval and this portion is called period. Sound wave is never perfect, smooth or uniformly periodic.

Frequency of the sound is the reciprocal value of period. It represents the number of period in a second and is measured in Hz cycles/sec.

Note: 1 KHz = 1000 Hz

The frequency range is divided into

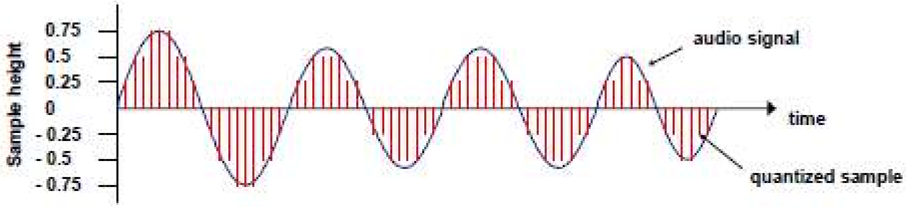
0 to 20 Hz Infra Sound

20Hz-20 KHz Human hearing frequency range.

Multimedia system typically makes use of sound only within the frequency range of human hearing. We will call sound in human hearing range audio and wave is called acoustics signals.

Amplitude: a sound has an amplitude property subjectively heard as loudness. The amplitude of a sound is the measure of the displacement of air pressure waveform from its mean.

Computer Representation of Sound:



- A transducer converts pressure to voltage levels
- The analog signal is converted into a digital stream by discrete sampling:
- The analogous signal is sampled in regular time intervals, i.e. the amplitude of the wave is measured
- Discretization both in time and amplitude (quantization) to get representative values in a limited range (e.g. quantization with 8 bit: 256 possible values)
- Result: series of values:

0.25	0.5	0.5	0.75	0.75	0.75	0.5	0.5	0.25	0	- 0.25	- 0.5	...

Sampling: Sound wave form the smooth, continuous is not directly represented in the computer. The computer measures the amplitude of the wave form in the regular time interval to produce the series the numbers. Each of this measurement is called sample. This process is called sampling.

Sampling rate: the rate at which a continuous wave form is sampled is called sampling rate. Like frequency, sampling rate are measured in Hz. For loss less digitization the sampling rate should be at least twice of the maximum frequency response.

Quantization:

Diagram goes here…

Just as a wave form is sampled at discrete times the value of sample is also discrete. The quantization of the sample value depends on the number of bits used in measuring the height of the wave form. The lower quantization lower quality of sound, higher quantization higher quality of sound.

QNO 2.2. What are MIDI devices? Why MIDI devices are required in multimedia system?Explain different components of MIDI?

Music: The relationship between music and computer has become more and more important, especially considering the development of MIDI (Musical Instrument and Digital Interface) and its important contribution in the music industry. MIDI interface between musical instrument and computer.

MIDI Basic Concept:

MIDI is the standard that manufacturer of electronic musical instrument have agreed upon. It is a set of specification, they use in building their instrument. So that the instrument of different manufacturers can communicate without difficulty. MIDI is defined in 1983.

MIDI Interface Components:

MIDI interface has two different components:

i. Hardware:

   Hardware connects the equipment. It specifies the physical connection between musical instruments. A MIDI Port is built in to the instrument and it specifies MIDI cable and deals with electronic signals that are sent over the cable.

ii. A data format:

Data format encodes the information travelling through the hardware. MIDI doesn't transmit an audio signal. For this propose MIDI data format is used. The MIDI encoding includes, besides the instrument specification, the notion of the beginning and ending node, basic frequency and sound volume. MIDI data format is digital. The data are grouped into MIDI message. Each MIDI message can communicate one musical event between instruments.

MIDI Reception Mode:

Mode 1: Omni on / poly
Mode 2: Omni on / mono
Mode 3: Omni off / poly
Mode 4: Omni off / mono

To tune a MIDI device to one or more channels, the device most be set to one of the MIDI reception mode. There are four MIDI reception modes which are given above.

The first half of the mode name specifies how MIDI device monitors the incoming MIDI channels. If Omni is turned on, the MIDI device monitors all the MIDI channels and responds to all channel messages, no matter which channels they are transmitted on. If the Omni is turned off, the MIDI device respond only to channel message, the device is set to receive.
The second half of the mode name tells the MIDI device how to play nodes coming in over the MIDI cable. If the option Poly is set the device can play several nodes at a time. If the node is set to mono the device play only one node at a time like a monophonic synthesizer.

MIDI Devices:

Synthesizer

- Sound generator
- Microprocessor
- Keyboard
- Control panel
- Auxiliary Controller
- Memory

Sequencers

Midi synthesizers are component that generates sound based on input MIDI software message. It is the heart of any MIDI system. A typical synthesizer looks like simple piano keyboard with a panel full of buttons. Most synthesizers have following components:

1. Sound Generator: Sound generator do the actual work of synthesizing. The propose of the rest of synthesizer is to control the sound generator. The principal propose of the generator is to produce a audio signals that becomes sound when fed into loud speaker. By changing the voltage oscillation of the sound signal, a sound generator changes the quality of the sound, i.e. pitch, loudness.
2. Microprocessor (Controller): It communicate with keyboard to know what notes a musician is playing and what command the musician want to send. Microprocessor then specifies note to sound generators. In other words microprocessor sends and receives MIDI message.
3. Keyboard: Keyboard provides the musician direct control of the synthesizer, pressing the keys on the keyboard. Which instruct the microprocessor what notes to be played and how long to play them?
4. Control Panel: It controls those functions that are not directly concerned with notes and duration. The panel controls include a slider that sets the overall volume of synthesizer a button that turns the synthesizer on, off and menu.
5. Auxiliary Controller: Auxiliary controller are available to give more control over the note played on the keyboard. It concern with notes and duration. Two common variables on synthesizer are pitch band and modulation.

6. Memory: Synthesizer memory is used to store patches for the sound generators and setting of the control panel.

Sequencers: It is an electronic device in cooperating with both hardware and software, which is used as storage server for generated MIDI data. A sequencer may be computer. Sequencer transforms the note into MIDI message.

MIDI Message:

1. Channel Message
   i. Channel Voice Message
   ii. Channel mode Message
2. System Message
   i. System Real-time message
   ii. System Common Message
   iii. System exclusive Message

MIDI Message: It transmits information between the MIDI devices and determines what kinds of musical events can be passed from device to device. The format of the MIDI message consists of the status byte (first byte of any MIDI message) which describe the kind of message and data byte. MIDI message are divided into following types:

1. Channel Message: It go only to specified devices. There are two types of channel message.
   i. Channel Voice Message: Channel Voice message send actual performance data between MIDI devices describing keyboard action, controller action and control panel changes. They describe music by defining pitch, note on, note off channel pressure, etc.
   ii. Channel Mode Message: Channel Mode message determine the way that a receiving MIDI device respond to channel voice message. It deals with how to play notes coming in over MIDI cables. Channel mode message includes Omni On, Omni Off, note off, note on, etc.

2. System message: System message go to all devices in a MIDI system because no channel number are specified. There are three types of system message.
   i. System Real-time Message: These are very short and simple, consisting of only one byte. They carry extra data with them. These messages synchronize the timing MIDI devices in performance. To avoid delay these message are sent in the middle of other message if necessary.
   ii. System Common Message: System common message are commands that prepares sequencers and synthesizer to play song. E.g. Song selected, find the common starting place in the song.
   iii. System Exclusive message: System exclusive message allow MIDI manufacturers to create customized MIDI message to send between the MIDI devices.

MIDI Software: The software application generally falls into four major categories.

1. Music recording and performance application
2. Musical notation and printing application
3. Synthesizer Patch editor and library patch
4. Music education application

MIDI and SMPTE Timing Standard

MIDI reproduces traditional note length using MIDI clock, which are represented through timing clock message. Using MIDI clock a receiver can synchronize with the clock cycle of sender. To keep the standard timing reference the MIDI specification state 24 MIDI clock = 1 quarter note.

As an alternative the SMPTE timing standard (society of Motion Picture and Television Engineer) can be used. The SMPTE timing standard was originally developed by NASA as a way to make incoming data from different tracking stations so that the receiving computer could fill what time each piece of data was created. SMPTE format consists of hour: minutes: second: frames: bits

-30 frames per second, SMPTE uses a 24 hour clock from 0 to 23 before recycling.

QNO 2.3 Explain the process of speech generation? Explain different components of speech recognition and understanding? Explain the research areas of speech analysis? Consider 8 bit analog to digital converter (ADC) with maximum amplitudes of 5 volts. If the input is 4.5 volts calculate its digital values?

Speech: speech can be perceived understood, and generated by human and also by machines. A human adjust himself/ herself very efficiently to different speakers and their speech habit. The human brain can recognize the very fine line between speech and noise.

Speech signals have two properties which can be used in speech processing.

1. **Voice speech signals** show during certain time intervals almost periodic behavior. Therefore we can consider these signals as quasi-stationary signals for around 30ms
2. **The spectrum of the audio signals** shows characteristics maxima, which are mostly 3-5 frequency bands. These maxima called format.

Speech generation:

1. Speech output system could transfer text into speech automatically without any lengthy processing.
2. Generated speech most be understandable and most sound natural.

Types of Speech generation:

The important requirement of speech generation is the generation of the real time signals. The easiest method for speech generation is to use pre-coded speech and play it back in the timely fashion.

Time-dependent sound concatenation:

CRUM

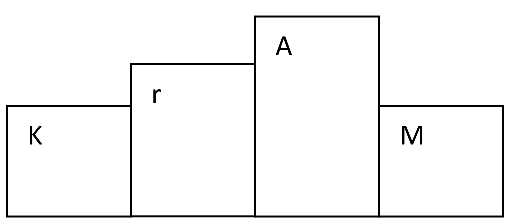


Fig: Phone sound concatenation

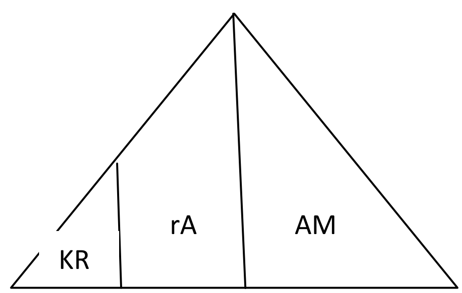


Fig: Di-phone concatenation

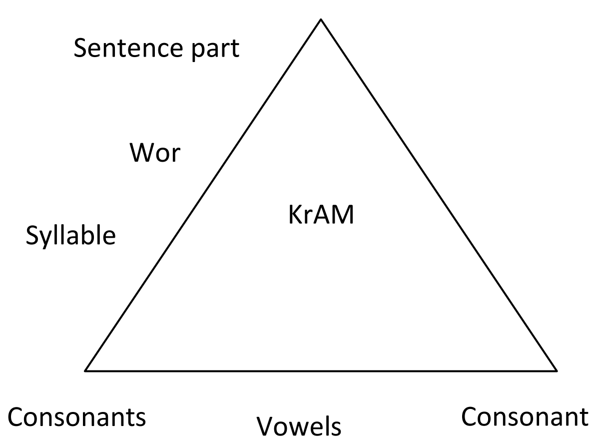


Fig: Word sound concatenation

Speech generation can also be performed by sound concatenation in a timely fashion. Individual speech unit are composed like building blocks, where the composition can occur at different levels.
In the simplest case, the individual phones are understood as speech units. In the above given example individual phone of the word CRUM are shown. It is possible with just a few phones to create from an unlimited vocabulary. The phone sound concatenation shows the problem during the transition between individual phones. This problem is called co articulation which is mutual sound effect. To solve these problems, Di-phone sound concatenation is used. Two phones can constitute a Di-phone. In the above figure, Di-phone of word CRUM is shown. The transition problem is not solved sufficiently on this level.
To make the transition problem easier, syllabus can be created. The speech is generated through set of syllabus. The above given figure word sound concatenation and syllabus sound shows the syllabus sound of word CRUM. The best pronunciation of word is achieved is storage of whole word. This leads towards synthesize the speech sequence.

The transition between individual sound units create an essential problem called co articulation which is the mutual sound influence throughout the several sound.

Frequency dependent sound concatenation:
Speech generation can also be based on frequency dependent sound concatenation. E.g. through formant synthesizing. Formants are the frequency maxima in the spectrum of the speech. Formants synthesize simulate the vocal track through filter. This characteristics value is filters middle frequency and their bandwidth. The method used for sound synthesize in order to simulate human speech is called the linear predictive coding (LPC) method.
Using speech synthesize a text can be transformed into acoustics signals. The typical component of this system is given in the figure below:

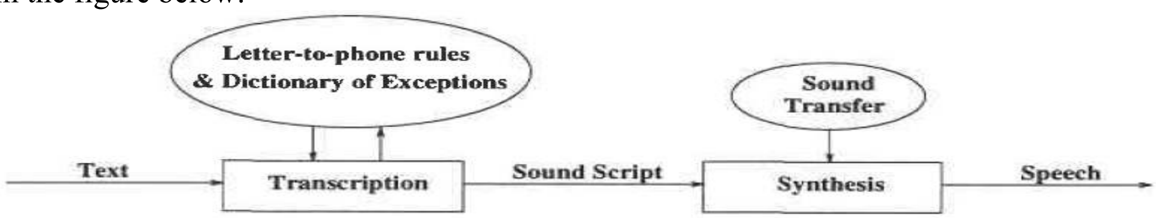


Figure 3.8: Components of a speech synthesis system with time-dependent sound concatenation.

In the first step the transcription is performed where text is translated into sound script. Most transcription methods work with later to phone rules. A dictionary of exception stored in the library.
In the second step the sound script is translated into a speech signals. Besides the problem of co-articulation ambiguous pronunciation most be considered.

Speech analysis:

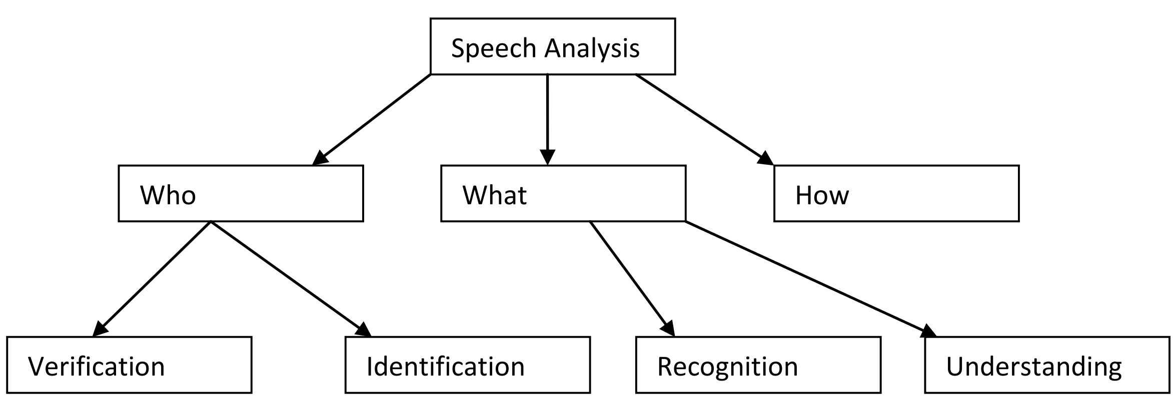


Fig. Speech analysis deals with research area shown in the above figure.

- Human speeches have certain characteristics determined by speaker. So speech analysis can server to analyze who is speaking. i.e. to recognize a speaker, for his identification and verification. The computer identifies and verifies the speaker using an acoustic finger print is digitally stored. It is digitally stored speech probe (certain statement of the speaker).
- Another main part of speech analysis is to analyze what has been said. To recognize and understand the speech itself.
- Another area of speech analysis tries to research speech pattern with respect to how the statement was said. E.g. a speaker sentence sounds differently if a person is angry or happy.

Speech Recognition System and Understanding:

The system which provides the recognition and understanding of speech signal applies this principle several times as follows:

1. In the first step, the principle is applied to a sound pattern and/or word model. An acoustic and phonetic analysis is performed.
2. In the second step, certain speech units go through syntactical analysis. In this step the errors in the previous step can be recognized.
3. The third step deals with semantics of the previously recognized language. Here the decision error of previous step can be recognized and corrected.

There are still many problems into speech recognition and understanding research.

1. Specific problem is presented by Room Acoustic with existed environment noise.
2. Word boundaries must be determined. Very often neighboring words follows into one another.
3. For the comparison of the speech elements to the existing pattern, time normalization is necessary. The same word can be spoken quickly or slowly.

Speech Transmission

The area of speech transmission deals with efficient coding of the speech signal to allow speech/sound transmission at low transmission rates over networks. The goal is to provide the receiver with the same speech/sound quality as was generated at the sender side. This section includes some principles that are connected to speech generation and recognition'

Signal Form Coding

This kind of coding considers no speech-specific properties and parameters. Here, the goal is to achieve the most efficient coding of the audio signal. The data rate of a PCM-coded stereo-audio signal with CD-quality requirements is:

Rate = 2 * 44100/s * 16 bit / 8 bit/byte = 176400b bytes/s = 1411200 bits / s

Telephone quality, in comparison to CD-quality, needs only 64 Kbit/s. Using Difference Pulse Code Modulation (DPCM), the data rate can be lowered to 56 Kbits/s without loss of quality. Adaptive Pulse Code Modulation (ADPCM) allows a further rate reduction to 32 Kbits/s.

Source Coding

Parameterized systems work with source coding algorithms. Here, the specific speech characteristics are used for data rate reduction. Channel vo-coder is an example of such a parameterized system (Figure g.I2). The channel vo-coder is an extension of a sub-channel coding. The signal is divided into a set of frequency channels during speech analysis because only certain frequency maxima are relevant to speech.

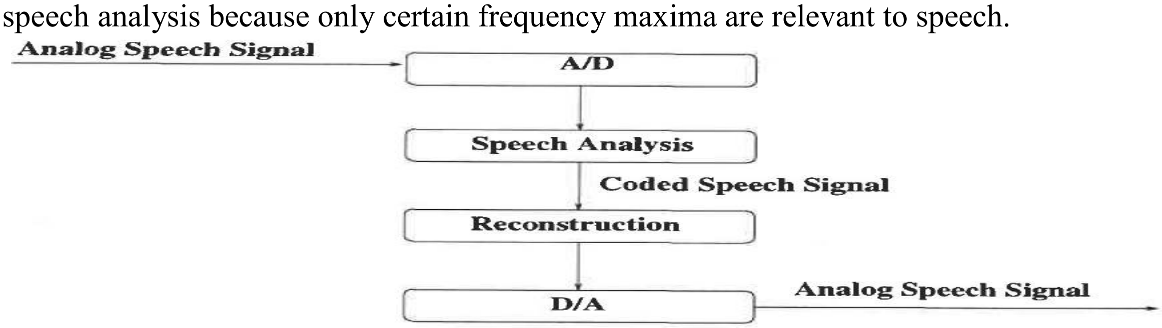


Figure 3.12: Source coding in parameterized systems: components of a speech transmission system.

- **Recognition / Synthesis Methods**

There have been attempts to reduce the transmission rate using pure recognition/ synthesis methods. Speech analysis (recognition) follows on the sender side of a speech transmission system and speech synthesis (generation) follows
on the receiver side (see Figure 3.13).

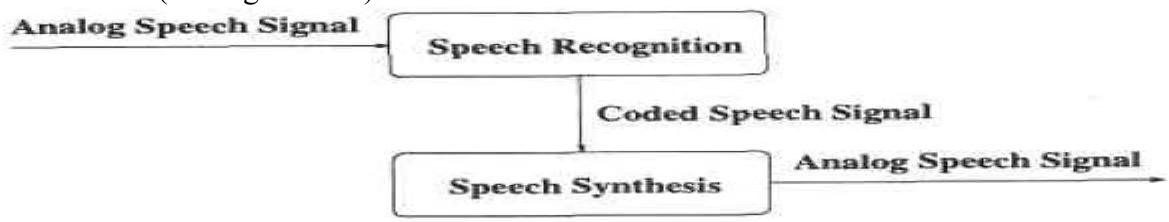
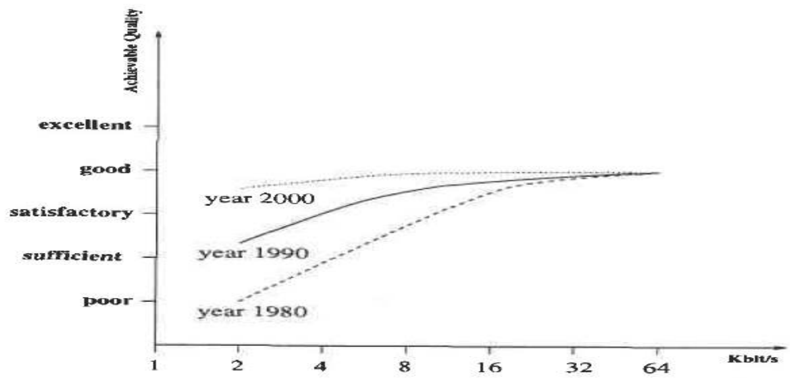


Figure 3.13: Recognition/synthesis systems: components of a speech transmission sgstem.

o Achieved Quality

The essential question regarding speech and audio transmission with respect to multimedia systems is how to achieve the minimal data rate for a given quality.

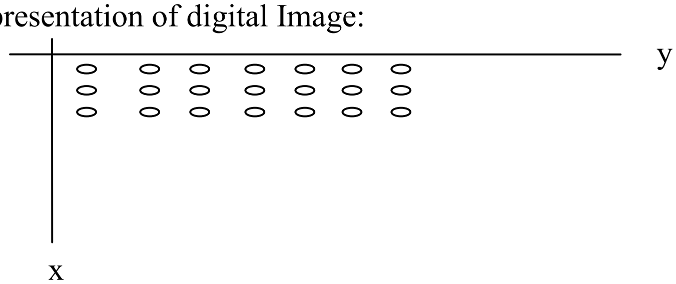


QNO 3.1 How can you represent a digital image? Difference between captured image and stored image format? Explain different types of graphics format? What are different possibilities of image transmission? Why image recognition is required? Explain different steps used in image recognition system? Difference between sound and speech? What is the general design issues considered in the user interfaces? Explain with suitable examples?

Image:

An image may be defined as two dimensional function f(x, y) where, x y are the spatial co-ordinate and the amplitude of 'f' at any pair of co-ordinates x, y is called the intensity or gray level of image at that point x, y and amplitude values of f are all finite, discrete quantities we all the image is digital image.

Representation of digital Image:



Digital image is represented by two dimensional matrix e.g. I[x] [y]

When I is the two dimensional matrix then, I [x][y] is the intensity value at the position corresponding to the row x and column y of the matrix.

Pixel:

Image Analysis:

Image analysis is concerned with techniques for extracting description from the image (shape, position, any distance, color) that are necessary for object recognition and classification.

Image Format:

Captured image format :

Captured Image Format is specified by:

spatial resolution (pixels x pixels) and *color encoding* (bits per pixel)

Examples:

VideoPix™ / SunVideo™ card:

spatial resolution: 320 x 240 pixels

color encoding: 1-bit (binary image), 8-bit (color or greyscale) 24-bit (color-RGB)

SunVideo™ card (capture and compression card with 30* fps (frames per second)):

CellB	30 fps	MPEG1 IP frames	17 fps
JPEG	30 fps	Capture YUV	30 fps
MPEG1 I frames	30 fps	Capture RGB-8/24	30 / 12 fps

* At least 25 frames per second are necessary if continuous motion is to be presented.

- However, it is possible to "cheat" a little and to work with smaller frame rates.

Stored image format :

A Stored image format **is a 2-dimensional array of values representing the image in a** bitmap **or** pixmap**, respectively.**

The data of a bitmap is a binary digit, data in a pixmap may be a collection of:

– 3 numbers representing the intensities of red, green, and blue components of the color

– 3 numbers representing the indices to tables of **red**, **green** and **blue** intensities single numbers as index to a table of color triples

– single numbers as index to any other data structures that represents a color / -system

– 4 or 5 spectral samples for each color

Each pixel may have additional information i.e. the normal vector to the surface.

Further properties of images: width, height, depth, size, version, etc.

[In many cases is data compression recommended or mandatory]

Formats:

– RIFF (Resource Interchange File Format) including formats for bitmaps, vector drawings, animation, audio and video (extension is BRIM)

– GIF (Graphical Interchange Format), X11 Bitmap, Sun Rasterfiles, PostScript, IRIS, JPEG, TIFF (Tagged Image File Format) etc.

Graphics Format

Graphics image formats **are specified through:**

– *graphics primitives*: lines, rectangles, circles, ellipses, text strings (2D), polyhedron (3D)

– *attributes*: line style, line width, color affect.

Graphics primitives and their attributes represent a higher level of an image representation. The graphics package determines which primitives are supported.

Advantages:

+ Reduction of the graphical image data
+ Easier manipulation of graphical images.

Disadvantage:

– Additional conversion step from graphical primitives and attributes to its pixel representation

Formats:

· SRGP (Simple Raster Graphics Package), one way conversion to bit-/pixmap
· PHICS (Programmer's Hierarchical Interactive Graphics Systems) and
· GKS (Graphical Kernel System) only image representation is in pixmap

Image Synthesis (generation)

Image synthesis **(generation) is an integral part of all computer graphical user interfaces and indispensable for visualising 2D, 3D and higher dimensional objects. E.g.:**

Graphical User Interface (GUI): **desktop windows system with icons and menu items**

- Office Automation and Electronic Publishing: desktop publishing, Hypermedia systems
- Simulation and Animation for Scientific Visualisation and Entertainment

Pictures can be dynamically varied by adjusting the animation speed, portion of the total scene in view, the amount of details shown etc.

- *Motion Dynamics:* Objects are moved and enabled with respect to a stationary or also dynamic observer, e.g. flight simulator.
- *Update Dynamics:* Objects being viewed are changed in shape, color, or other properties, e.g. deformation of an in-flight aeroplane structure.

image analysis (recognition).

Pixel information is very often insufficient! There is no information about:

- *shape* of an object
- *orientation* of an object
- *three-dimensional interpretation* of an object

➔ Image Analysis (Image Processing) is required.

Some applications or special topics are:

- Image enhancement
- Pattern recognition
- Scene analysis
- Computer vision

Example: Traffic scenes taken by a camera installed in a car

Problems: Is there a traffic sign visible?
Which traffic sign?
Is a moving car in front of our car?
Which type of car?
Which relative speed to our speed?

Image analysis is concerned with techniques for extracting descriptions about images:

- computation of perceived brightness and color
- partial or complete recovery of 3D data in a scene
- location of discontinuities corresponding to objects in a scene
- characterisation of the properties of uniform regions in a image

Image Enhancement

- improves image quality by eliminating noise (extraneous or missing pixels) or by enhancing contrast, i.e. X-ray images, computerized axial tomography (CAT)

Scene Analysis and Computer Vision

- deals with recognising and reconstructing 3D models of a scene from several 2D images, i.e. industrial robot sensing (relative sizes, shapes, positions, colors)

Pattern Detection and Recognition

- deals with detecting and clarifying standard patterns and finds distortions from these patterns, i.e. Optical Character Recognition (OCR)
- *static character recognition* (OCR), *dynamic recognition* (handwriting)

Turning radar detector:

$$0000000000000000000000\underbrace{11111111111111111}_{\text{an object}}0000000000000000000000000000000$$

Imperfect radar: some 0-1 mistakes:

00000001000011000000000001111110011011111110010000000001000001110000000000000

Elimination of mistakes by **digital filtering:**

Simplest version: **symmetrical (2m-1, m) filter**

$$a_1 a_2 \ldots a_i \ldots a_n \rightarrow a'_1 a'_2 \ldots a'_h \ldots a'_n \quad \text{with} \quad a'_h = \begin{cases} 1 & \text{if } \sum_{i=h-m+1}^{h+m-1} a_i \geq m \\ 0 & \text{else} \end{cases}$$

Example:

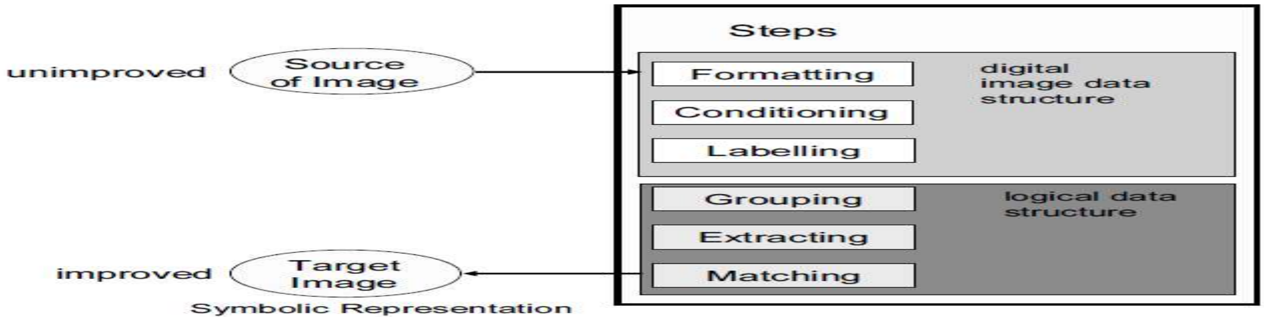
0000010000100011101101101111100010

↓

0000000000000011111111111111100000

Filtering by (5,3)-filter

Image Recognition Steps



Formatting

- capturing of an image and transforming to a digital representation

Conditioning

- based on a model that assumes that the observed image is composed of an informative pattern modified by uninteresting variations
- estimates informative pattern on the basis of the observed image
- suppresses noise and perform background normalisation by suppressing uninteresting systematic or patterned variations
- applied uniformly and context-independent

Labelling

- based on a model that assumes that the informative pattern has structure as a spatial arrangement of events
- determines in what kind of spatial events each pixel participates
- labelling operation:
  - *edge detection*, *corner detection,*
  - *thresholding* (e.g. filters significant edges and labels them),
  - *identification of pixels* that participate in various shape primitives

Grouping:

- identifies events by collecting or identifying maximal connected sets of pixels participating in the same kind of event (neural networks!)
- determines new sets of entities
- changes the logical data structure
- entities of interest after grouping are sets of pixels
- e.g. *line-fitting* is a grouping operation, where edges are grouped into lines

Extracting:

- computes for each group of pixels a list of properties:
- centroid, area, orientation, spatial moments, grey tone moments, spatial-grey tone moments, circumscribing circle, inscribing circle, number of holes in a region, average curvature in an arc, etc.
- measures topological or spatial relationships between two or more groupings, i.e. clarifies whether two groupings touch, are spatially close or layered

Matching:

- determines the interpretation of some related set of image events recognised previously with the extracting step
- associates events with some given 3D objects or 2D shapes
- *template matching* is a classical example of a wide variety of matching operations, compares examined pattern with known and stored models and chooses the best match

Conclusion:

- Conditioning, labelling, grouping, extracting and matching constitute a canonical decomposition of the image recognition problem.
- Each step prepares and transforms the data to facilitate the next step.
- On any level the transformation is an unit process and data are prepared for the unit transformation to the next higher level.
- Depending on the application, the sequence of steps has more than one level of recognition and description process.

Digital Image processing:

The field of digital image processing refers to the processing of digital image by means of digital computer.

steps of digital image processing:

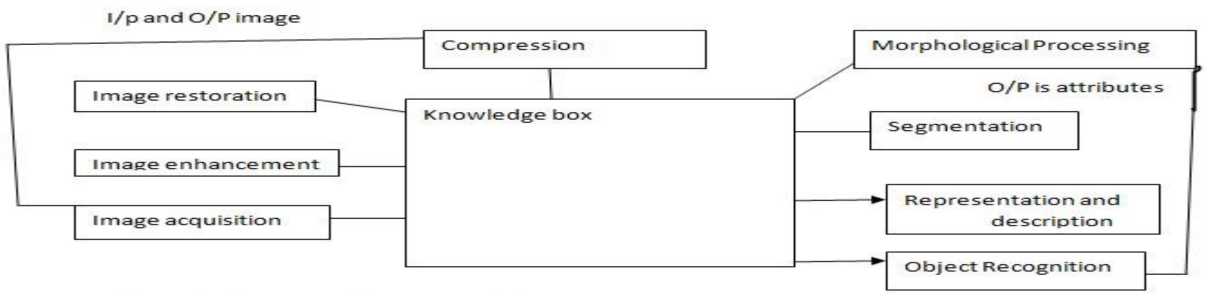


Fig: Fundamental steps in digital image processing

<u>Difference between Image processing and computer graphics:</u>

Computer graphics is a field related to generation of graphics using computer. It includes creation, storage and manipulation of image of the object. On the other hand image processing is a processing of a digital image by means of digital computer. It is the field related to image analysis (image enhancement, pattern detection, recognition, computer vision, etc.)

QNO4.1. What do you mean by visual representation? Explain the important measures in the visual representation?

Video signal representation includes:

– visual representation
– transmission
– digitization

Important measures for the *visual representation*:

1. Vertical Detail and Viewing Distance

Smallest detail that can be reproduced is a pixel.

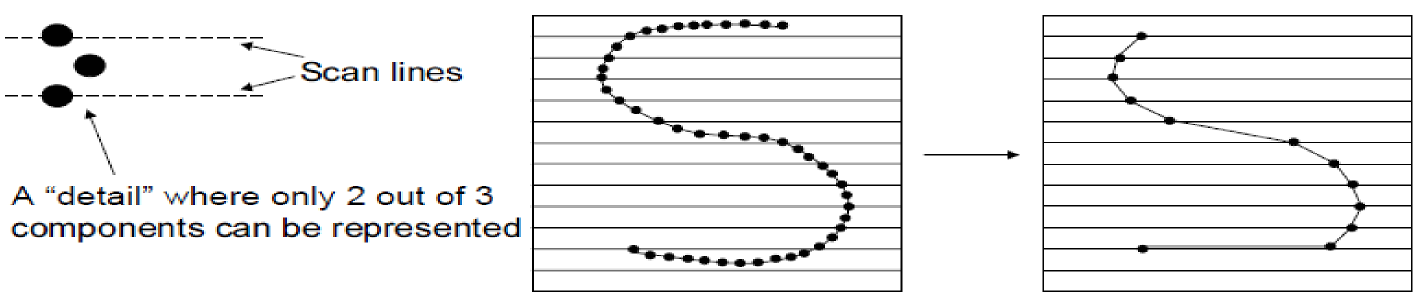
– Ideally: One pixel for every detail of a scene.

– In practice: Some details fall between scanning lines.

Kell factor: only 70% of the vertical details are represented, due to the fact that some of the details of the scene fall between the scanning lines. (Determined by experience, measurements).

The Kell factor of ≈ 0.7 is independent of:

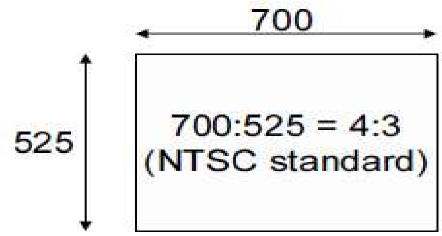
– way of scanning, i.e. whether the scanning is *progressive* (sequential scanning of lines)
– or whether the scanning is interlaced (alternate scanning, line 1, line 3, ... line n-1, line 2, line 4, ...)



Geometry of the television image is based on the *aspect ratio*, which is the ratio of the picture width W to the height H (W:H).

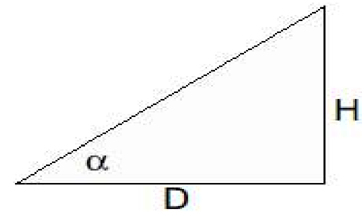
The conventional aspect ratio is 4/3 = 1.33 (W/H = 4/3)

Modern systems use 16/9 = 1.77



The *viewing distance* D determines the angle α subtended by the picture height H.

This angle is measured by the ratio of the picture height to viewing distance: $\tan(\alpha) = H/D$.



2. Horizontal Detail and Picture Width

Picture width for conventional television service is 4/3 · picture height.(see previous slide)

3. Total Detail Content of the Image

total number of picture elements = number of vertical elem. · number of horizontal elem.

= vertical resolution² · aspect ratio

= 525² · 4/3 (for NTSC)

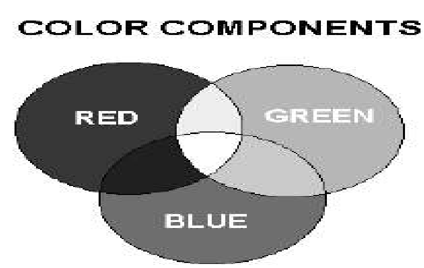
4. Perception of Depth (3D impression)

In natural vision: angular separation of the images received by the two eyes.

Television image: perspective appearance of objects, choice of focal length of camera lens (however: usually not real), changes in depth of camera focus.

5. Luminance and Chrominance

Color perception by the human brain is possible through the *additive composition* of Red, Green and Blue light (RGB system). The relative intensities of RGB values are transmitted to the monitor where they are reproduced at each point in time.



Another possibility is to use a different representation of color information by means of the YUV system

where

Y is the brightness (or „luminance“) information

and

U and V are color difference signals („chrominance“).

Y, U and V are functions of R, G and B.

Color encoding during transmission uses:

one luminance (brightness) signal Y (from 0/dark to 1/white), where

$Y = 0.30 \cdot R + 0.59 \cdot G + 0.11 \cdot B$ (R = G = B = 1, i.e. full red, green, blue ➔ Y = 1)

due to the color sensitivity of the human eye and two chrominance (color) signals.

$U = c_1 \cdot (B-Y)$; $V = c_2 \cdot (R-Y)$;

c_1 , c_2 = constants reflecting perception aspects of the human eye and the human brain!

Color impression due to:

– relative intensities of “Red”, “Green”, “Blue”.

– combinations of these intensities.

6. Temporal Aspects of Illumination

Motion is represented by a rapid succession of slightly different still pictures (frames). A discrete sequence of pictures is perceived as a continuous sequence of picture (due to "Lucky weakness" of the human brain).

Between frames, the light is cut off briefly.

For a realistic presentation, two conditions are required:

– repetition rate must be high enough to guarantee smooth motion

– the persistence of vision must extend over the interval between flashes.

7. Continuity of Motion

To perceive continuous motion the frame rate must be higher than 15 frames/sec.

For smooth motion the frame rate should be 24 - 30 frames/sec.

8. Flickering

Through slow motion a periodic fluctuation of brightness, a flicker effect, arises.

How to avoid this disturbing effect?

– A first trick: display each picture several times.

E.g.: 16 pictures per second: very inconvenient flicker effect.

➔ display every picture 3 times, i.e. with a refresh rate of 3 · 16 = 48 Hz.

– To avoid flicker a refresh rate of at least 50 cycles/sec is needed.

– Computer display achieves 70 Hz of refresh rate by the use of a refresh buffer.

– TV picture is divided into two half-pictures by line interleaving

– Refresh rate of 25 Hz (PAL) for the full TV picture requires a scan rate of 2 · 25 Hz = 50 Hz.

9. Temporal Aspect of Video Bandwidth

The eye requires a video frame to be scanned every 1/25 second.

Scan rate and resolution determines the video bandwidth needed for transmission.

During one cycle of video frequency (i.e. 1 Hz) at most two horizontally adjacent pixels can be scanned.

Vertical resolution and frame rate relates to horizontal (line) scan frequency:

vertical lines (b) · frame rate (c) = horizontal scan frequency

Horizontal resolution and scan frequency relate to video bandwidth:

horizontal lines (a) · scan frequency / 2 = video bandwidth

➔ video bandwidth = a · b · c/2

(since 2 horizontally adjacent pixels can be represented simultaneously

during one cycle of video frequency).

A computer system with a resolution of a = 1312 and b = 800 pixels out of which 1024 · 786 are visible and a frame rate c = 100 Hz needs:

a horizontal scan frequency of
800 · 100 Hz = 80 kHz

and a video bandwidth of
1312 · 80 kHz / 2 = 52.48 MHz

Signal Formats

- YUV
- YIQ
- Composite Signals
  - Instead of sending each component on one channel, send them all
- Computer Video Formats
  - Current video digitization hardware differ in
    - Resolution of digital images (frames)

- Quantization
- Frame rate

Motion depends on the display hardware

- Computer Video Formats
  - Color Graphics Adapter (CGA)
    - Resolution: 320 x 200
    - 2 bits / pixel
  - Enhanced Graphics Adapter (EGA)
    - Resolution: 640 x 350
    - 4 bits / pixel
  - Video Graphics Array (VGA)
    - Resolution: 640 x 480
    - 8 bits / pixel
  - Super Video Graphics Array (SVGA)
    - Resolution: 1024 x 768, 1280 x 1024, 1600 x 1280
    - 8 bits / pixel
  - Video accelerators are needed to avoid reduced performance at higher resolutions
    - Check the storage requirements of the above systems!

QNO.4.2 What is computer based animation? Why computer based animation is required? Explain different steps employed for computer based animation system? Explain about different animation languages?

to animate = "to bring to life"

***Animation* covers changes in:**

time-varying positions (motion dynamics),

shape, color, transparency, structure and texture of an object (update dynamics) as well as lightning, camera position, camera orientation and focus.

***Basic Concepts* of animation are:**

Input Process

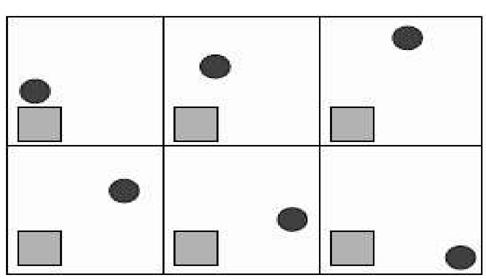
- Key frames, where animated objects are at extreme or characteristic positions must be digitized from drawings.
- Often a post-processing by a computer is required.

Composition Stage

Foreground and background figures are combined to generate an individual frame.

Placing of several low-resolution frames of an animation in an array leads to a *trail film* (pencil test), by the use of the *pan-zoom* feature (This feature is available for some frame buffers).

The frame buffer can take a part of an image (pan) and enlarge it to full screen (zoom). Continuity is achieved by repeating the pan-zoom process fast enough.

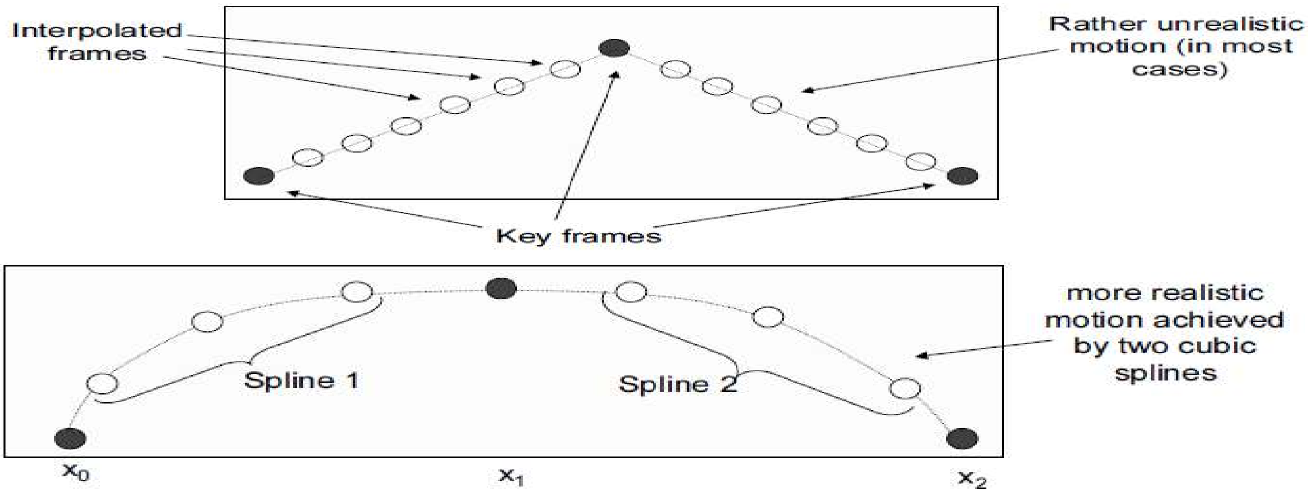


In-between Process

Composition of intermediate frames between key frames.

Performed by linear interpolation (lerping) between start- and end-positions.

To achieve more realistic results, cubic spline-interpolation can be used.



A function s: is called cubic interpolating spline to the points $a = X_0 < X_1 < \ldots < X_{n+1} = b$, if

1. s is twice continuous differentiable
2. i = 0, ..., n which means is a polynomial of degree 3

This line is smooth, because the polynomials have equal primary and secondary derivations at the points X_0 , X_1 ,... ,X_{n+1}

Categories for *Animation languages*:

Linear-list Notations

– Events are described by starting and ending frame number and an action (event).

– 17, 31, C, ROTATE "HOUSE", 1, 45 means:

between frames 17 and 31 rotate the object HOUSE around axis 1 by 45 degrees, determining the amount of rotation at each frame from table C.

General-purpose Languages

– Embed animation capability within programming languages.

– Values of variables as parameters to the routines that perform animation.

– e.g. ASAS, which is built on top of LISP:

(grasp my-cube): cube becomes current object

(cw 0.05): spin it clockwise, by a small amount

Graphical Languages

(Motivation: Textual languages cannot visualize the actions by looking at the programme code.)

– Describe animation in a more visual way than textual languages.

– Express, edit and comprehend the changes in an animation.

– Explicit descriptions of actions are replaced by a picture of the action.

– Examples are: GENESYS, DIAL and S-Dynamic System.

Techniques for *controlling animations* (independent of the language which describes the animation):

Full Explicit Control:

– Complete way of control, because all aspects are defined:

simple changes (scaling, translation, rotation) are specified or key frames and interpolation methods (either explicit or by direct manipulations by mouse, joystick, data glove) are provided.

Procedural Control:

– Communication between objects to determine properties.

– Physically-based systems: position of one object may influence motion of another (ball cannot pass a wall).

– Actor-based systems: actors pass their position to other actors to affect their behavior (actor A stays behind actor B).

Constraint-based Systems

– “Natural” way of moving from A to B is via a straight line, i.e. linearly. However, very often the motion is more complicated.

– Movement of objects is determined by other objects, they are in contact with.

– Compound motion may not be linear and is modeled by constraints (ball follows a pathway).

Tracking Live Action

– Trajectories of animated objects are generated by tracking live action.

– Rotoscoping: Film with real actors as template, designers draw over the film, change background and replace human actors with animated counterparts.

– Attach indicators to key points of actor’s body. Tracking of indicator positions provides key points in the animation model.

– Another example:

Data glove measures:

– position and orientation of the hand

– flexion and extension of fingers and fingerparts

From these informations we can calculate actions, e.g. movements

Kinematics:

– Description using the position and velocity of objects.

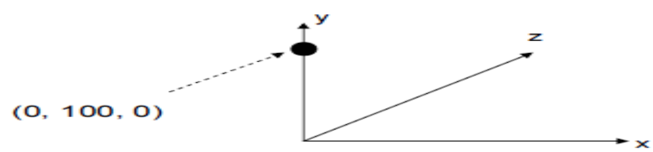
– e.g. At time t = 0 the CUBE is at the origin. It moves with the constant acceleration of 0.5 m/s² for 2 sec. in the direction of (1,1,4).

$$\left.\begin{array}{c}(0, 0, 0) \rightarrow (1, 1, 4) \\ 2 \text{ seconds} \\ b = 0.5 m/s^2\end{array}\right\} \text{kinematic description of a motion of a cube}$$

Dynamics:

– Takes into consideration the physical laws that define the kinematics.

– e.g. At time t = 0 the CUBE is in position (0 meters, 100 meters, 0 meters) and has a mass of 5 Kg. The force of gravity (g) acts on the cube (Result in this case: the ball will fall down).



For the display of animations with raster systems the animated objects have to be *scan-converted* to their pixmap in the frame buffer.

This procedure has to be done at least 10 (better: 20) times per second in order to give a reasonably smooth effect.

Problem:

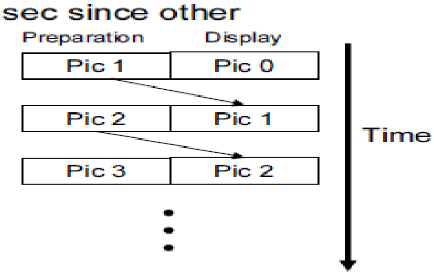
Frame rate of 20 pictures/sec. requires manipulation, scan-conversion and display of an object in only 50 msec.

Scan conversion should only use a small fraction of these 50 msec since other operations (erasing, redrawing, ... etc) have to be done too.

Solution:

Double-buffering: frame buffer is divided into two images, each with half of the bits of the overall frame buffer (“pipeline”).

While the operation (like rotating) and scan-conversion is processed for the second half of the pixmap, the first half is displayed and vice versa.



Symbolic Representation

- Graphical descriptions (circle) of an animated object (ball) + operations (roll)
- Animation is displayed at the receiver by scan-conversion of objects to pixmap
- Transmission rate depends on (transmission rate is context dependent):
  - size of the symbolic representation structure, size of operation structure,
  - number of animated objects and of commands.

(Transmission time is shorter since symbolic representation is smaller than pixmap representation, but display time is longer since scan converting has to be done)

Pixmap Representation

- Longer times for data transmission than with symbolic representation, because of the large data size of pixmap.
- Shorter display times, because no scan-conversion is necessary at receiver side.
- Transmission rate = size of pixmap · frame rate (fixed transmission rate).

QNO4.3 Explain different types of television system? Justify why HDTV is better than the conventional system?

Video format standards for conventional television systems:

NTSC (National Television Systems Committee)

- Exact frame rate of 29.97 Hz to maintain 4.5 MHz between the visual and audio carriers.
- Delay between frames: 1000 ms / approx. 30 frames per sec. = 33.3 ms.
- The chrominance signal C for NTSC transmission can be represented as:

$$C = \underbrace{\frac{B-Y}{2.03}}_{0.493\cdot(\text{B-Y})}\sin(\omega_c t) + \underbrace{\frac{R-Y}{1.14}}_{0.877\cdot(\text{R-Y})}\cos(\omega_c t)$$

For transmission the signal is shifted up to 3.58 MHz.

PAL (Phase Alternating Line, invented by W. Bruch/Telefunken 1963)

- Frame rate of 25 Hz, delay between frames: 1000ms / 25 frames per sec. = 40ms
- Quadrature amplitude modulation similar to NTSC, but color carrier is not suppressed.

Phase of the R-Y (V) signal is reversed by 180 degrees from line to line, to reduce color errors that occur from amplitude and phase distortion during transmission.

The chrominance signal C for PAL transmission, with U = B - Y, can be represented as:

$$\begin{aligned} C &= \frac{U}{2.03}\sin(\omega_c t) \pm \frac{V}{1.14}\cos(\omega_c t) & U = B\text{-}Y \\ &= 0.493(B-Y)\cdot\sin(\omega_c t) \pm 0.877\cdot(R-Y)\cdot\cos(\omega_c t) & V = R\text{-}Y \end{aligned}$$

SECAM (Sequential Couleur Avec Memoire)

- Based on frequency modulation, frame rate of 25 Hz.

System	Total Lines	Visible Lines	Vertical Resolution	Horizontal Resolution	Video Bandwidth	Aspect Ratio	Optimal Viewing Distance
NTSC -i	525	484	242	330	4.2 MHz	4/3	7 m
NTSC -p	525	484	340	330	4.2 MHz	4/3	5 m
Pal -i	625	575	290	425	5.5 MHz	4/3	6 m
PAL -p	625	575	400	425	5.5 MHz	4/3	4.3 m
SECAM -i	625	575	290	465	6 MHz	4/3	6 m
SECAM -p	625	575	400	465	6 MHz	4/3	4.3 m

-i interlaced, -p progressive (non-interlaced)

Improved Definition TV (IDTV)

EDTV systems are conventional systems which offer improved vertical and/or horizontal resolution by some "tricks".

Comb filters improve horizontal resolution by more than 30% according to literature.

Separate black and white from color information to eliminate rainbow effects while extending resolution.

Progressive (non-interlaced) scanning improves vertical resolution.

Insertion of "blank lines" in between "active lines", which are filled with information from:

- above line
- below line
- same line in previous picture

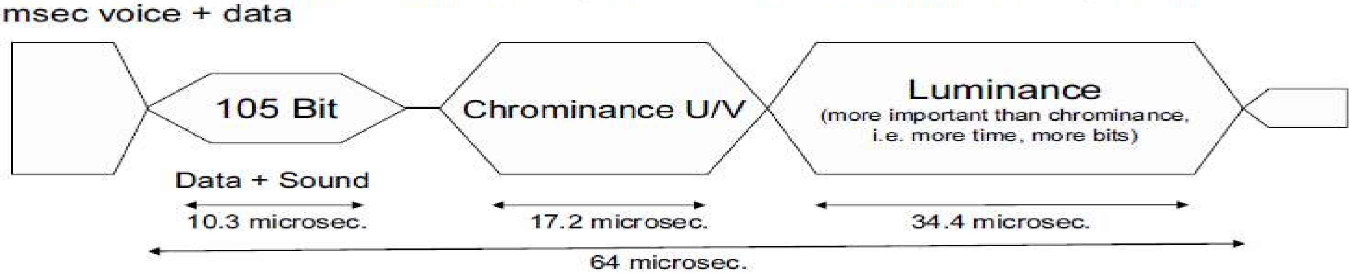
Other EDTV developments are:

IDTV (Improved-Definition Television)

- Intermediate level between NTSC and HDTV (High-Definition Television) in the U.S.
- Improve NTSC image by using digital memory to double scanning lines from 525 to 1050.
- One 1050-line image is displayed in 1/60 sec (60 frames/sec).
- Digital separation of chrominance and luminance signals prevents cross-interference

D2-MAC (Duobinary Multiplexed Analogue Components)

- Intermediate level between current television and HDTV in Europe.
- Uses time-multiplexing for component transmission. 64 microsec. time slot is divided into: 34.4 msec luminance signal, 17.2 msec chrominance signal and 10.3 msec voice + data



- 625 lines (574 visible), aspect ratios 4/3 and 16/9 are supported
- Audio/data are transmitted in duobinary coding with a rate of
  105 Bits / 64 msec = 1.64 MBits/s
- 2 high quality stereo channels or up to 8 channels of lower audio quality

Composite Coding

HDTV is characterized by:

Higher Resolution, approx. twice as many horizontal and vertical pixels as conventional systems, (lines transmitted > 1000), total bandwidth 5 - 8 times larger than today.

Aspect Ratio: 16/9 = 1.777.

Preferred *Viewing Distance:* between 2.4 and 3.3 meters.

Digital Coding is essential in the design and implementation of HDTV:

1. ***Composite Coding*** **(sampling of the composite analog video signal i.e. all signal components are converted together into a digital form) is the straightforward and easiest alternative, but:**
   - cross-talk between luminance and chrominance in the composite signal,
   - composite coding depends on the television standard,
   - sampling frequency cannot be adopted to the bandwidth requirements of components,
   - sampling frequency is not coupled with color carrier frequency.

Alternative:

2. *Component Coding* (separate digitization of various image components):
   - the more important luminance signal is sampled with 13.5 MHz,
   - the chrominance signals (R-Y, B-Y) are sampled with 6.75 MHz.
     - ➔ Global bandwidth: 13 Mhz + 6.75 MHz > 19 MHz
   - Luminance and chrominance signals are quantized uniformly with 8 Bits.

Due to high data rates (1050 lines · 600 pixels/line · 30 frames/sec) bandwidth is approx.

19 MHz and therefore substandards (systems which need a lower data rate) for transmission have been defined (see the section on Transmission).

Worldwide 3 different HDTV systems are being developed:

United States
- Full-digital solution with 1050 lines (960 visible) and a scan rate of 59.94 Hz.
- Compatible with NTSC through IDTV.

Europe
- HD-MAC (High Definition Multiplexed Analog Components), developed by EUREKA.
- 1250 lines (1000 visible) and a scan rate of 50 Hz.
- Halving of lines (625 of 1250) and of full-picture motion allows simple conversion to PAL or D2-MAC.
- HD-MAC receiver uses digital image storage to show full resolution and motion.

Japan
- MUSE is a modification of the first NHK (Japan Broadcasting Company) HDTV Standard.
- MUSE is a Direct-Broadcast-from-Satellite (DBS) System, where the 20MHz bandwidth is reduced by compression to the 8.15 MHz available on the satellite channel.
- Full detail of the 1125 line image is retained for stationary scenes, with motion the definition is reduced by approx. 50%.

QNO 5.1. What are different compression issues? Why image compression is required? Explain different types of compression? Calculate the total memory required for SVGA video of 24 bit true color for 5 minutes without compression? What do you mean by Huffman coding or decoding? Explain it with suitable example and algorithm? Design Huffman code for probabilities: 0.1, 0.4, 0.15, 0.05, 0.1, and 0.2.

Data Compression

On compressed graphics, audio and video data require considerable storage capacity which in the case of uncompressed video is not often feasible in today's CD technology. The same is for multimedia communication. The data transfer of uncompressed video data over digital network requires very high bandwidth to be provided for a single point to point communication. To provide visible and cost effective solution, most multimedia system handle compressed digital and audio stream data.

Coding requirement:

Entropy Encoding	Run-length Coding	
	Huffman Coding	
	Arithmetic Coding	
Source Coding	Prediction	DPCM
		DM
	Transformation	FFT
		DCT
	Layered Coding	Bit Position
		Subsampling
		Sub-band Coding
	Vector Quantization	
Hybrid Coding	JPEG	
	MPEG	
	H.261	
	DVI RTV, DVI PLV	

Entropy encoding

- Data stream is considered to be a simple digital sequence without semantics
- Lossless coding, decompression process regenerates the data completely
- Used regardless of the media´s specific characteristics
- Examples: **Run-length encoding**, **Huffman encoding**, **Arithmetic encoding**

Source encoding

- Semantics of the data are taken into account
- Lossy coding (encoded data are not identical with original data)
- Degree of compression depends on the data contents
- Examples: Content prediction techniques; e.g. use of spatial redundancies between still images for data compression.
  Discrete Cosine Transformation (DCT) as transformation technique of the spatial domain into the two-dimensional frequency domain

Hybrid encoding

- Used by most multimedia systems
- Combination of entropy and source encoding
- Examples: JPEG, MPEG, H.261, DVI

Major steps of data compression:

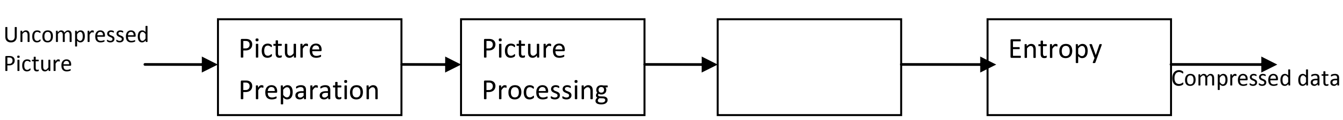


fig: Major steps of data compression

1. **Picture preparation:** this includes analog to digital conversion and generating a appropriate digital representation of the information. An image is divided into block of 8 X 8 pixels and represented by fixed number of bits/pixel
2. **Picture processing:** Processing is the actually the first step of compression process which makes use of sophisticated algorithm of transformation from time to frequency domain. It can be performed using DCD (Discrete Cosine transform).
3. **Quantization Process:** Quantization process the result of the previous step. It specifies the mapping of the real number into integer. This can be considered as the equivalent of µ-law and A- law which apply the audio data. Quantization is performed using the different number of bits per coefficient.

4. **Entropy Coding:** Entropy coding is usually last step of data compression. It compresses a sequence of digital stream without loss. For e.g. a sequence of zero's in the data stream can be compressed by specifying the number of occurrence followed by zero it.

The processing and quantization cab be repeated iteratively several times in feedback loop.
The term spatial domain (time domain) refer to the image plane itself and approaches in this category are based on discrete manipulation of pixel in an image.
Frequency domain processing technique is based on modifying the Fourier transform of an image.

QNO 6.1. What is session management? Draw its block diagram and explain with suitable example?

Session Management

Architecture

A session management architecture is built around an entity - session rnanager - which separates the control from the transport

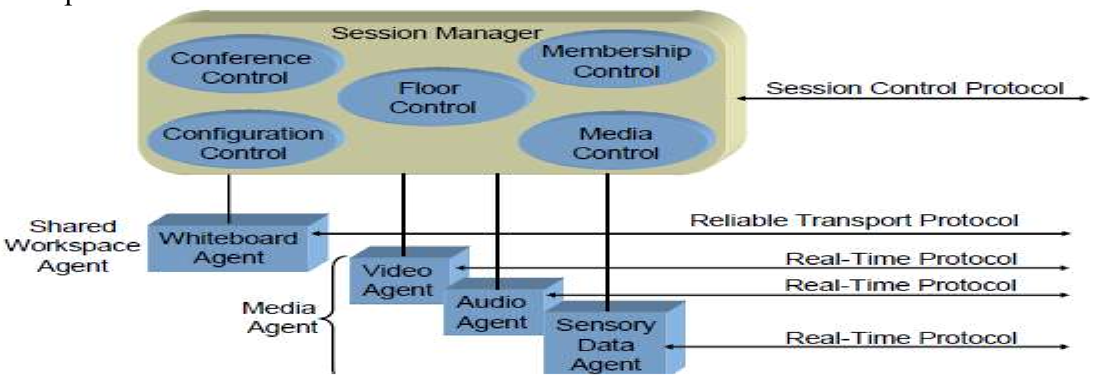


Figure: Session control architecture.

Session Management

• Important part of the multimedia architecture
• Separates the control part from the data part of functionality
• Comprises all parts to coordinate a cooperative work
• The entity which takes over thistasks is called *Session Manager*

Media Agents

• Are separated from the session manager
• Have the responsibility for decisions specific to each type of media – each type of media has its own media agent
• Each media agent performs its own control mechanisms for the particular medium, e.g. mute/unmute, change of video quality, start/stop sending, …

Shared Workspace Agent

• Transmits shared objects among the shared applications, e.g. telepointer coordinates or graphical/textual objects

Session Management

Each session is described by its session state, e.g. by:

- Starting time of session
- Policies associated with a session
- Session name

The session state information is either

- Private (e.g.: local resources) or
- Public and shared among all sessions participants

Session states are changed during the establishment or modification of a session:

- Session manager negotiates, agrees and sets the logical state of its own session
- It negotiates, agrees and sets billing policy with other session managers
- Furthermore, it permits “publishing” a session, i.e. allowing others to join a session

Session Management Functionalities

Local functionalities

_ Control management (for shared workspace, such as floor control)
_ Membership control management (participant authentication, ...)
_ Media control management (communication among media agents, ...)
_ Configuration management (exchange of interrelated QoS parameters or selection of appropriate services according to QoS,)
_ Conference control management (establishment, modification and closing of a conference, ...)

Remote functionalities

_ Communication with other session managers
_ Exchange of state information which may include floor information, configuration information, etc..

Audio/Video Conference Control Service

Conference Control

• Management of conferences (establishing, closing, adding/removing users)
• Providing information about conferences (conference name, duration, etc.)

Centralized or distributed control

• *Centralized control* is easy to implement
• *Distributed control* is much more complicated but less sensitive to failure (central control does not work if central instance is inactive)"Distributed" means to distribute the whole functionality over all participants
• *Replicated control* is a variant of distributed control: each participant has a complete control component, control functionality is performed together

Functions of conferencing control

• Establishing a conference (users agree upon a common conference state, e.g. encoding rules, access rights, chairman of the conference)
• Adding new participants and removing participants during a conference
• Closing a conference
• Management of state information of the conference during its lifetime, e.g.:
_ Conference name
_ Start of the conference
_ Policies associated with the conference
_ Participants
_ Duration
_ Topic

Audio/Video Conferencing Control – Centralized

Central Server

• Stores the conference state (i.e. conference state is centrally organized)
• Manages conference control
• Is allowed to explicitly change the conference state and to distribute the corresponding information to the participants.

Establishment of a Conference with Centralized Control

1. Initiation (by the chairman) of the conference selects an initial group of participants
2. The chairman inquires addresses of the conference participants from a central directory server
3. These participants are explicitly invited
4. Each invited client responds to the invitation
→ Initiator is informed who will participate
5. Negotiation of conference policies and admission of resources are performed. This shared conference state is stored on a central server and distributed to all participants using a reliable messaging service.

Audio/Video Conferencing Control - Distributed

Distributed Control

• *No* global notion of group membership is taken into account
• *No* consistency guarantees
• Periodic exchanges of the conference status between the participants in order to minimize the loss of control by an unreliable messaging service
• Two variants: really distributed control vs. replicated control

Establishment of a Conference with Distributed Control

1. Initiator of the conference establishes a multicast tree with multicast entries
2. Participants join the conference by tuning into a particular multicast entry (multicast address) announced by group rendezvous principle
3. Each participant informs the others about his own status

Comparison of Conference Control Mechanisms

Centralized Conference Control

+ Guaranteed consistency of the conference state
+ Easy to implement
- Adding of a new participant causes large delays (explicit exchange of the conference state among all participants)
- Difficult reestablishment of the conference state after a link failure

Distributed Conference Control (only one control but distributed over several sites)

+ Fault tolerance (easy reestablishment after network link failures)
+Well suited for small conferences
- The conferences participants may not have the same view of the state space
- Hard to implement, costly, difficult to understand

Replicated Conference Control (same control function on different sites)

+ Low "request response time"
- Adding a new participants causes large delays (explicit exchange of the conference state among all participants) and higher network load

Further Control Services

Floor Control

• Provides access to the shared workspace
• Maintains data consistency (see Application Sharing)
• Controls the level of simultaneity and granularity for access control
• Example for a simple floor control: floor-passing mechanism
  _ At any time only one participant has the floor (token principle)
  _ The floor is handed over to another participant when requested
  _ To obtain the floor, the participant must explicitly take a certain action to signal a floor change
• With real-time audio the floor control promotes turn-takings
• Floor control for real-time audio is often used to control bandwidth usage

Media Control

• Mainly includes a functionality for the synchronization of media streams

Configuration Control

• Includes a control of media quality, QoS handling, resource availability, and other system components to provide a session according to user's requirements

Membership Control

• Includes services for invitation to and registration into a session as well as the modification of membership during a session

QNO. 6.2 Why resource management is required? In multimedia communication, Draw the block diagram of resource management architecture and explain it?

Resource

A resource is a system entity required by tasks for manipulating data. Each resource has a set of distinguishing characteristic

- There are active and passive resources. An active resource is, for example, the CPU or a network adapter for protocol processing; it provides a service. A passive resource is, for example, the main memory (buffer space) or bandwidth (link throughput); it denotes some system capabilities required by active resources.
- A resource can be either used exclusively by one process or shared between various processes. For example, a loudspeaker is an exclusive resource, whereas bandwidth is a shared resource.
- A resource that exists only once in the system is known as a single resource) otherwise it is a multiple resource. In a transporter-based multiprocessor system, the individual CPU is a multiple resource.

Resource Management Architecture

Resources are managed by various components of a resource management subsystem in a networked multimedia system

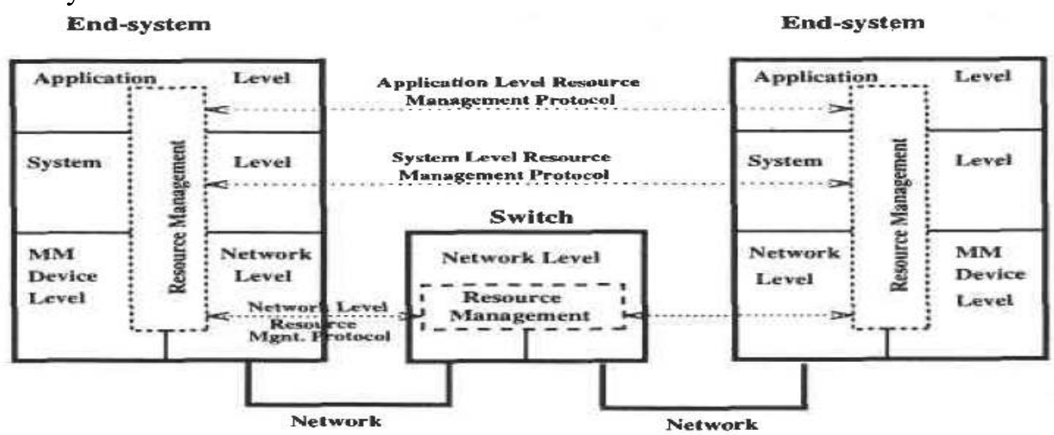


Figure: Resource management in MCSs.

The main goal of resource management is to provide guaranteed delivery of multimedia data. This goal implies three main actions:

(1) to reserue and allocate resources (end-to-end) during multimedia call establishment so that traffic can flow according to the QoS specification, which means distribution and negotiation of the QoS specification for system

Components involved in the data transfer from the source(s) to the sink(s);

(2) to provide resources according to the QoS specification, which means adhering to resource allocation during multimedia delivery using proper service disciplines; and

(3) to adapt to resource changes during on-going multimedia data processing

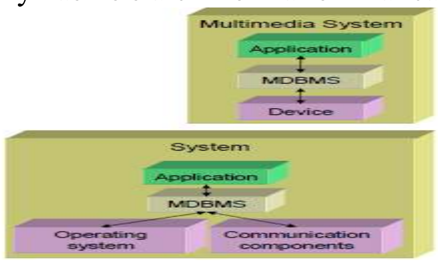
Multimedia Database Management System

- Main task of a Database Management System (DBMS) is to abstract from the details of:
  - _ Storage access
  - _ Storage management
- Location of the MDBMS:
  - _ Embedded between the application domain and the device domain
- *Integration into the system*:
  - _ Through operating system
  - _ Communication components
- *Persistence of data*:
  - _ Data outlive processing programs and technologies, e.g. companies have to keep data in databases for several decades
- *Consistent view of data*:
  - _ Synchronization protocols provide a consistent view of data in a multi-user system
- *Security of data*:
  - _ Transaction concepts ensure security and integrity protection in case of system failure. Recovery of lost data

• *Query and retrieval of data*:

_ Query languages such as SQL (Structured Query Language) enable formulating database queries

_ Each entry has its state information that can be retrieved correctly

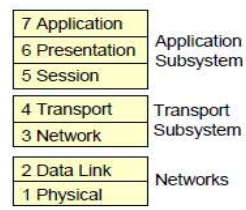


CH-6 Communication system in Multimedia

Multimedia applications have several requirements – as well for the data transmission as for controlling interactivity. For structuring control and transmission functionality and implement common protocols:make a distinction to two system components

1. Application Subsystem

- Responsible for the management and service issues for group cooperation and session orchestration
- Supporting a large scale of multimedia applications, e.g.
  - Multimedia Mail
  - Virtual Reality Applications
  - Video Conferencing
  - CSCW (Computer Supported Cooperative Work)



2. Transport Subsystem

- Transport and network layer protocols for multimedia applications (streaming)

Application Subsystem

Streaming multimedia data often is used in *cooperative computing*

• Beneath streaming services (transport subsystem), some control functionality for cooperation support is needed

• Cooperative Computing is generally known as *Computer Supported Cooperative Work* (CSCW).

Tools for cooperative computing:

_ Electronic mail

_ Shared whiteboards

_ Screen sharing tools

_ Application sharing

_ Text-based conferencing systems

_ Video conference systems (e.g. MBone Tools, ProShare from Intel, PictureTel,Teles Online, NetMeeting from Microsoft)

Cooperation Dimensions

Computer-Supported cooperation may be categorized according to the following parameters:

• *Time*

_ Asynchronous cooperative work (not at the same time)

_ Synchronous cooperative work (at the same time)

• *User Scale*

_ Single user, two users (“dialogue, point to point“, direct cooperation) or groups with more than two users

_ Static or dynamic groups, depending on if the members are pre-determined or not

• *Control*

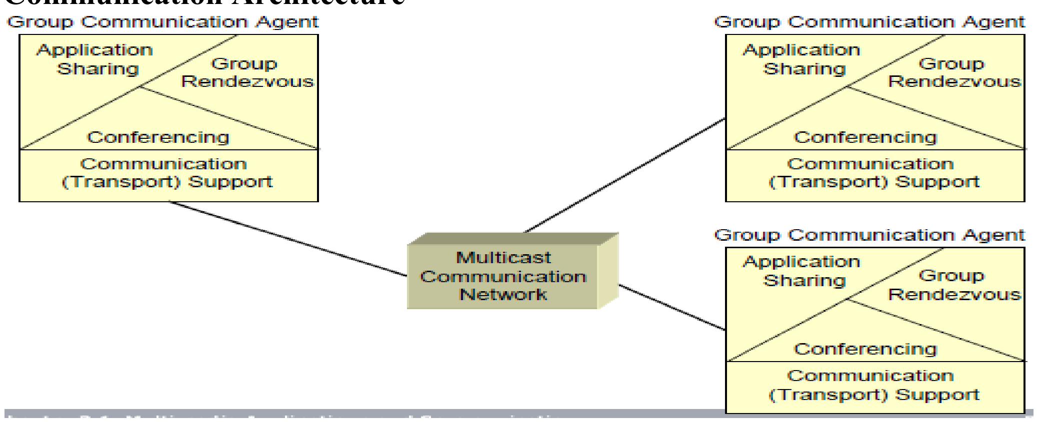
_ Centralized, i.e. controlled by a “chairman”

_ Distributed, i.e. control protocols provide consistent cooperation
• *Locality*
_ Cooperation at the same place
_ Tele-cooperation of users at different places

Cooperation Awareness

• Cooperation-transparent systems
_ Existing applications are extended for cooperation (Single user document editors expanded for simultaneous editing of a shared document among several users, e.g. some text processors)
• Cooperation-aware systems
_ Dedicated software application for CSCW (e.g. Lotus Notes, conferencing systems)

Group Communication Architecture



Group communication:
• Synchronous communication
• Asynchronous communication
Support Model: *Group communication agents* (cooperating via a multicast network)
• *Group rendezvous* (organization of meetings and delivering information)
• *Shared applications* (simultaneous replication and modification of information to multiple users, e.g. telepointing, joint editing)
• *Conferencing (*audio/video*)*
System Model:
• Client/server model
Interface Model:
• Exchanging information within the support model (object oriented)

Clients:
• User interfaces supporting interaction between group members and the system
• User Protocols: Between the clients, e.g.
_ Open, close, dynamically join or leave a meeting
_ Floor passing (*Floor* = kind of token held by the participant currently performing an action)

Servers:
• Specialized servers for different tasks of the group communication work, e.g.
_ *Directory server* (for group rendezvous services)
_ *Application sharing server*
_ *Conference server*
_ *Multimedia server* (for intermediate synchronization)
• Group Work Management Protocols: Between the clients and the servers, e.g.
_ Registration of meetings
_ Queries for information about meetings

Group Rendezvous

Setting up a group and delivering information about groups and meetings:
• *Synchronous Rendezvous Methods*:
_ Explicit invitations (point-to-point or point-to-multipoint)

• Initiator of a meeting has to know where users reside

_ Using directory services (X.500) as a knowledge base with information about

• Name of the meeting/conference
• Registered or authorized participants
• Role of participants

• *Asynchronous Rendezvous Methods*:

_ Using e-mail
_ Using bulletin boards (local bulletin boards or World Wide Web)
_ Simple Conference Announcement Protocol (SCAP)

Application Sharing

Application Sharing enables a cooperative work: several users are working at the same time on the same document at different places:

• Shared objects (documents) are displayed in shared windows
• Shared application program (e.g. word processor) executes input from a participant
• Modifications of the document are distributed among all participants
• Consistency of shared objects has to be guaranteed
• Floor passing control has to be provided

Application Sharing can be realized *centralized* or *distributed (i.e. replicated)*

Centralized

• A single copy of the shared application is available and runs at one site only
• All input is forwarded to that site
• Output (shared document) is distributed to all sites
• Only one participant at a time is having control

+ Easy consistency maintenance (only one copy of the application)

– High network traffic (for output distribution)

– Relatively unreliable (if central site is overloaded or fails then everything is down)

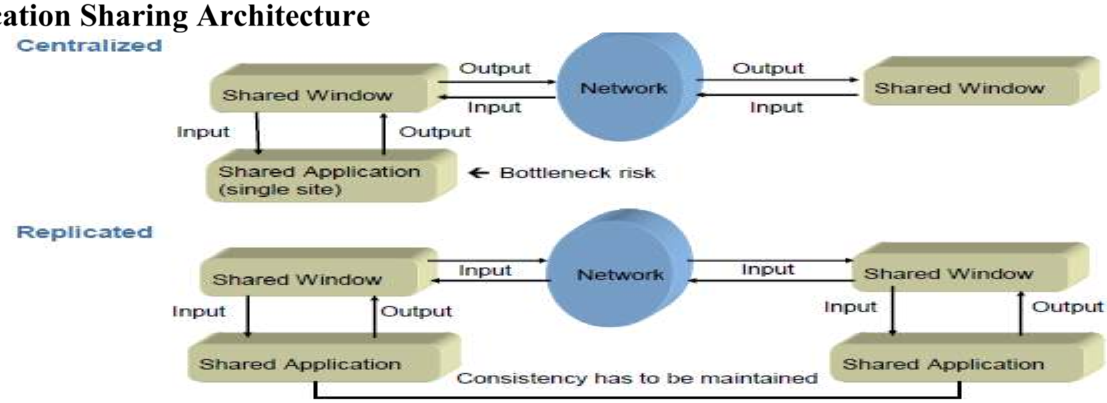
Replicated

• A copy of the shared application runs locally at each site
• Input is distributed to all sites

+ Low network traffic and response times (only input distribution to all sites is necessary) apart from the significant traffic arising from consistency maintenance

– Difficult consistency maintenance

Application Sharing Architecture



For both architectures, consistency of shared documents has to be maintained, e.g. by:

• Centralized locks
• Dependency detection
• Floor passing control

Floor Passing Control

• Only the floor holder has right to manipulate shared documents in shared windows, e.g. to perform the following operations:

_ Open or close shared windows
_ Load a document into a shared window

_ Give input to shared application

• Necessary: control protocol for managing and passing on the floor; it checks whether the active site is floor holder

_ Floor holder accepts and processes inputs; distributes results to other sites

_ No floor holders: discards its own inputs; accepts input which comes from other sites

Audio/Video Conferencing

Audio/video conferencing denotes a management service which controls simultaneous face-to-face communication between multiple users using multiple media (video, audio, text etc.)

• Video:

_ Distinction between several scenarios with different number of active conference participants (all, a few, speaker or moderator only)

_ Large video walls with multiple high-resolution screens may be used especially for

• Conferences with more than four participants

• Display of view-graphs, images, animations etc.

• Audio (true full-duplex high-quality communication with echo cancellation):

_ Used for discussions

_ Important for clarifying visual information

• Maybe application sharing etc. for cooperative work during the conference Important part of A/V conferencing: **session management**

Transport Subsystem

Multimedia applications have high requirements on network protocols:

1. High data throughput

_ Deliver as much data as possible in short time

2. Fast data forwarding

_ Deliver data as fast as possible

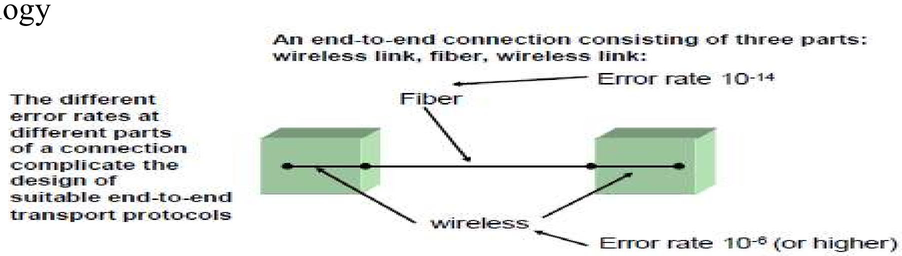
3. Service guarantees

_ Deliver data with regard to negotiated policies (throughput, delay)

4. Multicasting

_ 1:*n* and *m*:*n* point communication [*a*:*b* denotes: *a* senders and *b* receivers] *Needed: special transport resp. network protocols*

Most transport protocols are designed for unreliable and "relatively slow" networks Modern high speed networks need new transport protocols that take into account the properties of this new network technology



Data Throughput

- Audio and video data typically have a stream-like behavior
- Even in compressed mode, they demand high data throughput (16 kbit/s for compressed audio, 64 kbit/s for original PCM-audio, 2 Mbit/s for MPEG-coded video)
- In a workstation or in a network several of those streams may coexist
- Telephone services or video conferencing demand real-time computing of the data streams

This requires not only suitable transport protocols, but also high performance workstations which are able to compute several multimedia streams simultaneously and can transmit the packets in appropriate speed to the network interface

Service Guarantees / Multicasting

Service guarantees

• Service guarantees are important for the acceptance of multimedia applications

• Multimedia applications need guarantees, such as: throughput ≥ minvalue, delay ≤ maxvalue1, jitter ≤ maxvalue2

• To give service guarantees, resource management must be used – without this, in end-systems and switches/routers multimedia systems cannot provide reliable QoS to their users because transmission over unreserved resources may lead to dropping or delaying of packets

Multicasting

• Multicasting is important for multimedia applications in terms of sharing resources like the network bandwidth

• Many multimedia applications, such as video conferencing, have multicast characteristics

Transport Subsystem

Multimedia applications generate and consume a huge amount of data. They produce high demands for the underlying infrastructure and communication architecture:

- Data should be copied directly from adapter to adapter to reduce copy overhead, e.g. from the video board to the network interface by using direct memory access (DMA). With DMA technique, the application itself never really touches the data, it only sets the correct switches for the data flow by connecting sources to sinks
- The data transfer involved by the layered structure of the protocols form a bottleneck, hence other mechanisms must be found
- For error-recovery some protocols use retransmission techniques which impose requirements on buffer space for queues at the expense of larger end-to-end delays
- The synchronous behavior of most multimedia data streams must be mapped onto the asynchronous transfer mode of the underlying networks

Are the "old" transport protocols like TCP and UDP suitable for multimedia transmission and, if not, which new protocols exist or have to be developed?

CH-7 Multimedia Documents

A *multimedia document* then is a document which comprises at least one continuous (time-dependent) medium and one discrete (time-independent) medium

Multimedia document architecture and its elements

Exchanging documents requires that the *document architecture* is known. This requires a definition of an architecture:

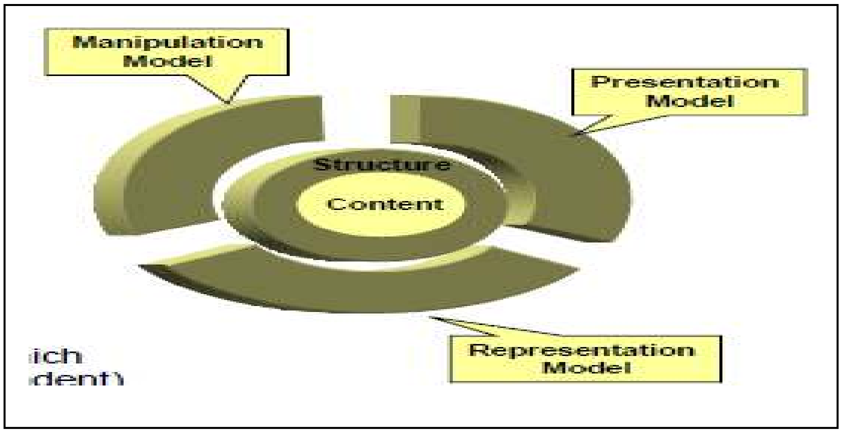
• *Content*: multi-/mono-media information

• *Structure*: spatial and temporal relations between information

• *Manipulation Model*: definition of operations for creation, change and deletion of information

• *Representation Model*: exchange protocol and data format

• *Presentation Model*: rules for document presentation



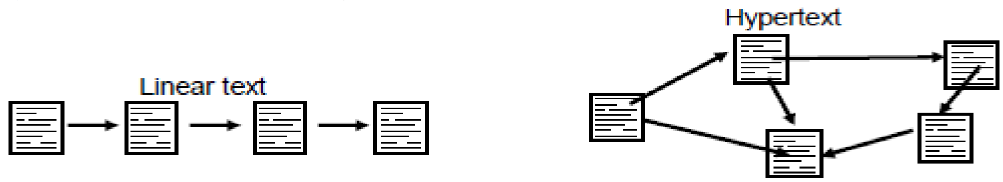
Mono-media Document Example: Hypertext

Hypertext System

• A system that allows to read several related textual documents nonlinearly *Node-Link Hypertext Model*:

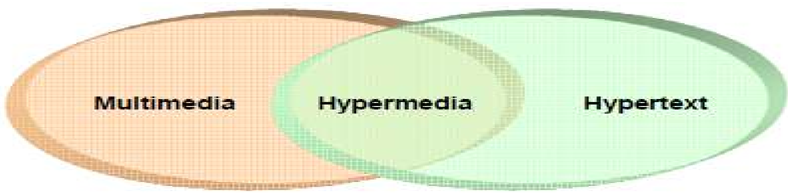
• Individually chunks of (textual) information (also called *nodes*) are set in relation by means of hyperlinks

• *Hyperlink*: relation between two or more nodes, defining a structure
• *Hypertext Document*: distinct set of nodes and links which constitutes a logical entity
• *Hyperweb*: distinct set of hyperlinks



Multimedia Documents: Hypermedia

• *Hypermedia* is the generalization of hypertext to include additional media like graphics, photos, audio clips, video sequences, animations
• Synchronization and linking of these media elements to other elements must be possible
• Hypermedia systems allow interactive, integrated and synchronized presentation of multimedia information



3 Layers of different functionalities:

Presentation and user interface

Based on a given structure and user's display, it is decided:

• which data to present
• how the data are presented

Hypermedia abstract machine

• Determine the structure of the document
• Knowledge about references, data structures, attributes

Database

• Storage of data as objects without semantics or structure definitions
• Storage management: consistency for multi-user access, persistency, fault tolerance, …

Description of hypermedia documents is possible e.g. with SGML, HTML/XML, or MHEG

Standard Generalized Markup Language (SGML)

SGML was evolved from an IBM internal project, and was strongly supported by American publishers:

• Authors define titles, tables, etc. inside a document in a uniform way, without any description of the actual representation
• Publisher determines layout

Basic ideas:

• Author uses *tags (markups)* to mark parts of the text to be e.g. a title or a table
• SGML determines how tags have to look like
• User groups agree on the meaning of the tags
• Formatter generates document layout from tags

SGML defines syntax, not semantics!

Relationship between:

• Document
_ Data content
_ Tags (markups)

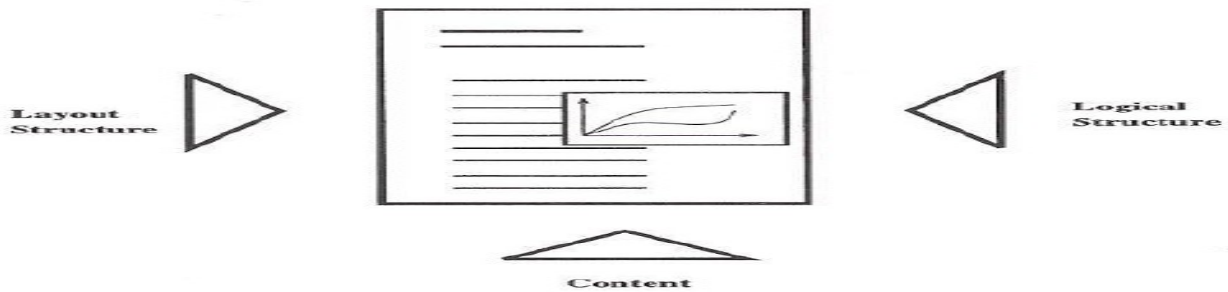
• Document Type Definition (DTD):
  _ Set of markup declarations, define
    • Element types
    • Attributes of elements
    • Hierarchical relationships between elements
• Procedures
  _ Specify the document processing
  _ Correspond to functions of the formatter

Open Document Architecture ODA

The *Open Document Architecture (ODA)* was initially called the Office Document Architecture because it supports mostly office-oriented applications. The main goal of this document architecture is to support the exchange, processing and presentation of documents in open systems. ODA has been endorsed mainly by the computer industry, especially in Europe.

Details of ODA

The main property of ODA is the distinction among content, logical structure and layout structure. This is in contrast to SGML where only a logical structure and the contents are defined. ODA also defines semantics.



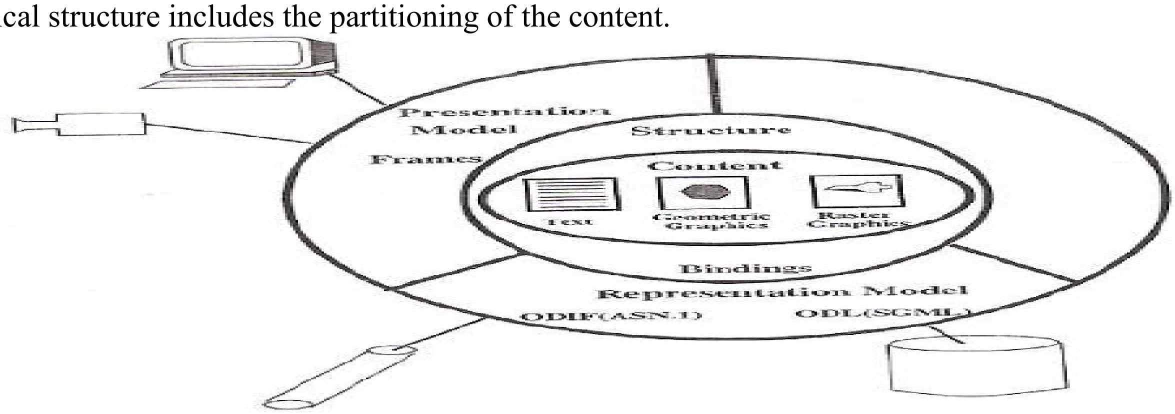
ODA : Content layout and logical view

A content architecture describes for each medium:
(1) the specification of the elements,
(2) the possible access functions and,
(3) the data coding.

ODA has content architectures for media *text, geometrical graphics* and *raster graphics*.
Contents of the medium text are defined through the *Character Content Architecture*. The *Geometric Graphics Content Architecture* allows a content description of still images.

Layout structure and Logical Structure

The Structure and presentation models describe-according to the information architecture-the cooperation of information units. These kinds of meta information distinguish layout and logical structure.
The layout structure specifies mainly the representation of a document. It is related to a two dimensional representation with respect to a screen or paper. The presentation model is a tree. Using frames the position and size of individual layout elements is established. For example, the page size and type style are also determined.
The logical structure includes the partitioning of the content.



Multimedia and Hypermedia Information Coding Expert Group (MHEG)

MHEG is an "umbrella standard" for multimedia objects. It provides a language for control of delivery of multimedia objects:

- Exchange format and representation for composed (time-dependent) multimedia information
- Provides a structures for interactive, spatial and temporal related information
- Suitable for real-time (distributed) multimedia/hypermedia applications (no complex parsing and interpretation required)
- Platform independent description
- Typical application: devices with small resources like point-of-sales terminals, video on demand set-top boxes (MHEG is e.g. used in DVB), ...

MHEG Family

Outdated standards:

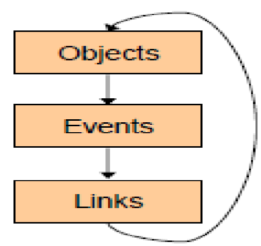
- MHEG-1: MHEG Object Representation, Base Notation (ASN.1)
- MHEG-2: MHEG Alternate Notation (SGML)
- MHEG-3: MHEG Extension for Scripting Language Support
- MHEG-4: Registration Procedures for Format Identifiers

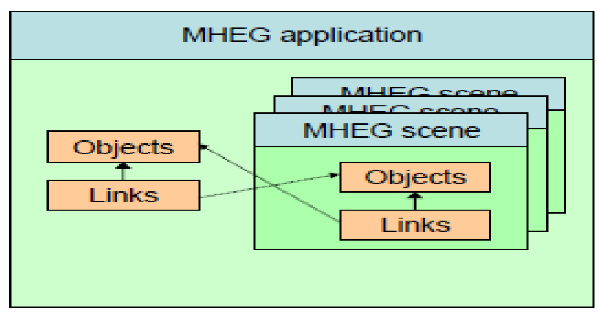
Current standards:

- **MHEG-5: Support for Base-Level Interactive Applications**
- MHEG-6: Support for Enhanced Interactive Applications: JavaVM
- MHEG-7: Conformance Testing
- MHEG-8: XML

MHEG-5: Objects, Events, Links

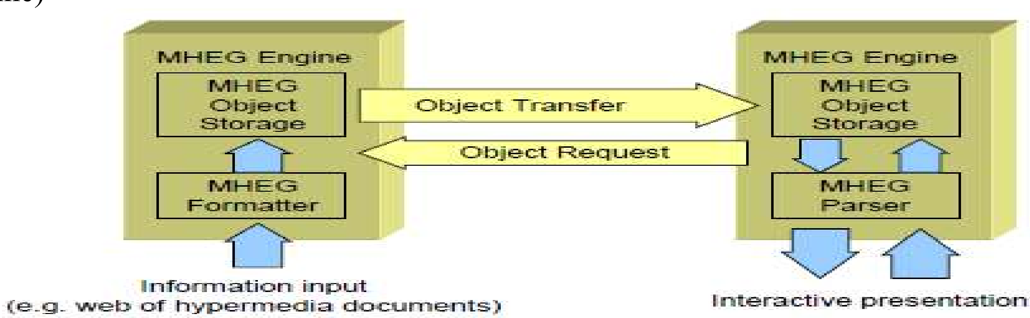
Multimedia applications can be described using MHEG as collection of scenes. Both contain *objects* and *links*; on the happening of certain *events* for an object (mouse movement, clicks, …) the effects of the corresponding link are processed.



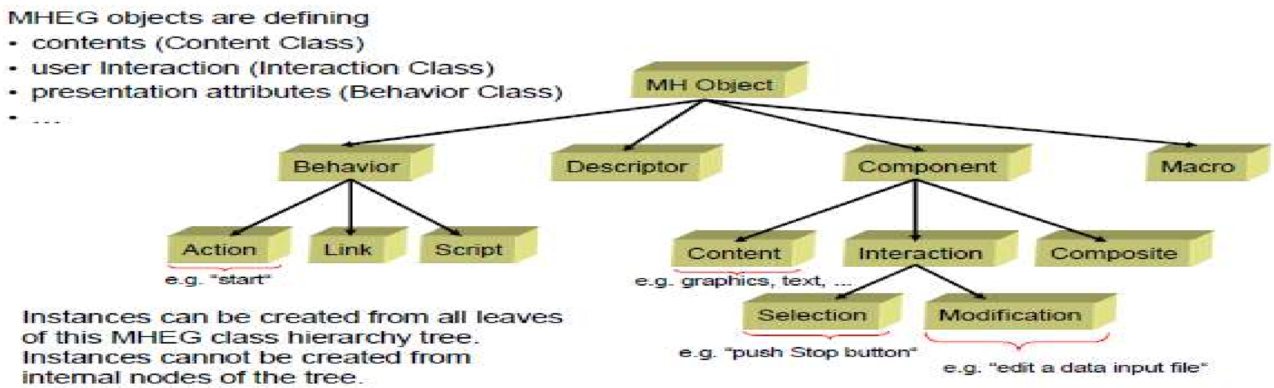


MHEG Objects

MHEG standard: data structures and binary representation of MHEG objects (not: retrieval protocol or MHEG Engine)



MHEG Object Classes



CH-8 Multimedia User Interfaces:

- Computer interfaces that enable human-computer interaction using multiple media
- Media determine how human-computer interaction occurs
- But main question is not how – it is *how good* (decisive for user acceptance)

Architectural Issues:

- Information characteristics for presentation
- Effective human-computer interaction
- Video and audio at the user interface
- User-friendliness of interfaces (primary goal!)
  → Development goes towards new user interfaces:
    1. Virtual Environments
    2. Ubiquitous Computing

General Design Issues

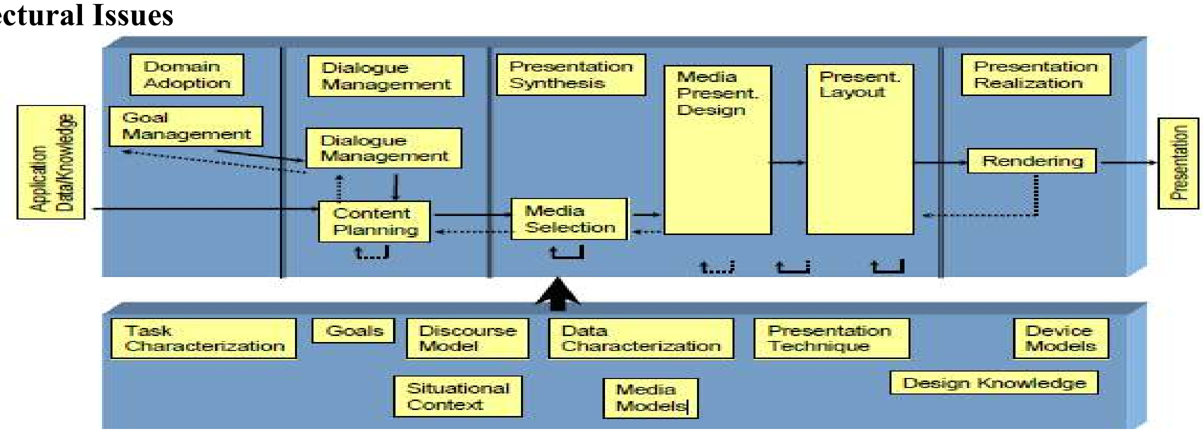
Main emphasis in design of Multimedia User Interfaces (MUI) is multimedia *presentation*.
The general issues to be considered are:

- To determine the appropriate information content to be communicated
- To represent the essential characteristics of the information
- To choose the proper media for information presentation
- To coordinate different media and assembling techniques within a presentation
- To provide interactive exploration of the information presented
- Keep care of
  - set of functionality vs. easy usability
  - professional vs. ad-hoc user
  - technical feasibility vs. futuristic visions

→*Appropriateness Principle*

"The surface presentation used by the artifact should allow the person to work with exactly the information acceptable to the task: neither more nor less" (Norman, "Cognitive Artifacts", 1991)

Architectural Issues



Information Characteristics for Presentation

Types (ordering information)

- Coordinates vs. amount (specify points in time, space or other domains)
- Intervals vs. ratio (suggests the type of comparisons meaningful among elements of coordinate and amount data types)

Relational structures

- Functional dependencies (e.g. bar chart)
- Non-functional dependencies (e.g. entry in a relational database)

Multi-domain relations

- Multiple attributes of a single object set (e.g. position, colors, ...)
- Multiple object sets (e.g. graphical symbols on a map)
- Multiple displays (e.g. multiple windows)

Large data sets

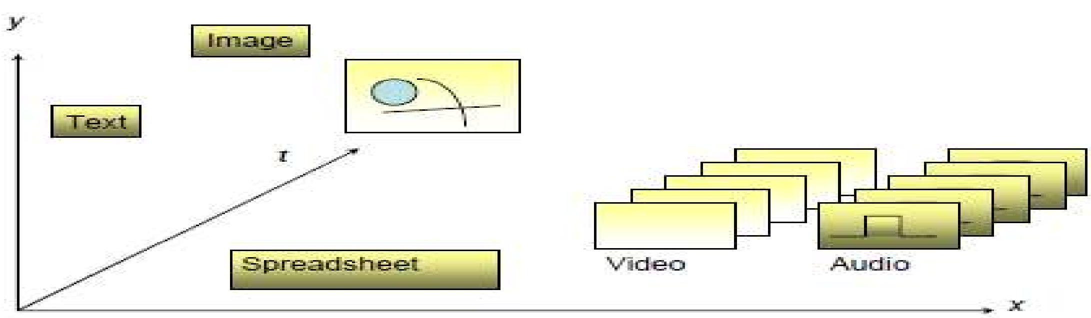
- Numerous attributes of collections of heterogeneous objects (e.g. presentation of semantic networks)

Effective Human-Computer Interaction

- For presentation design the following topics must be considered:
  - _ Content selection
  - _ Media selection
  - _ Coordination
- One of the most important issues regarding MUI is *user-friendliness*. Therefore during the design of a MUI the following main issues must be considered:
  - _ Context
  - _ Linkage to the world
  - _ Evaluation of the interface
  - _ Interactive capabilities
- Some guidelines for good interface design:
  - _ Grouping of logically related functions
  - _ Graphical symbols or sequences instead of text (associative, recognizable)
  - _ Definite and up-to date information about system state (e.g. adaptive cursors)
  - _ Immediate reaction (e.g. progress indicators)

Extension through Video and Audio

Time as a new presentation dimension in a User Interfaces: “Illusion of continuity“ created by presentation of a “sequence of static elements”



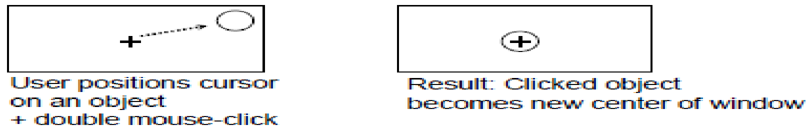
Video at the User Interface

- Video is implemented through a continuous sequence of individual images (video frames at frame rate of 25 frames/sec and higher)
- Hardware, software and heterogeneous solutions are available for coding and decoding of the frames with more than 30 frames/sec
- Example: remote camera control application
  - _ A camera is remotely controlled by a computer. The camera receives control information from the computer and sends the video data back.
  - _ User Interface: camera control through:

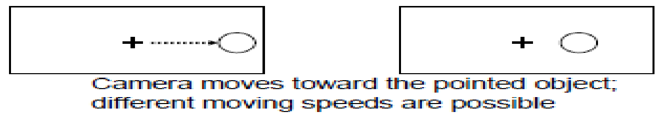
• Keyboard
• Buttons in window system
• Scroll bars
• Mouse, joystick, etc.

Direct Manipulation of the Video Window:

- *Absolute Positioning*



- *Relative Positioning*



Audio at the User Interface

Speech Analysis
- • Speaker-dependent (training of the system needed, more words recognized)
- • Speaker-independent (no training needed, limited set of words recognized)

Dimension of Space
- • Monophony (all audio sources have the same spatial location)
- • Stereophony (allows bilateral listening to hear lower intensity sounds)
- • Quadrophony → concept of two or more separate channels

Audio Windows
- • Audio windows as the graphical representation of audio locations
- • One audio window per audio source
- • Changing the position of the audio window on the desktop changes the location of the audio source

User-Friendliness

User-friendliness is the main property of a good user interface – but requirements of applications differ. Generally applicable criteria are:

- Easy to learn and remember instructions
- Context-sensitive help functions
- Effective instructions
  - ➢ Present logically connected functions together
  - ➢ Use graphics instead of text
  - ➢ Quick activation of actions
  - ➢ Useful for both professional and sporadic users
- Aesthetics
- Effective implementation support
- Entry elements
- Meaningful location of functions
- Presentation
- Dialogue boxes
- Additional design criteria, e.g. show progress in time intensive tasks

- alphabetically ordered
or
- logically grouped

CH-10 Synchronization

- Synchronization in refers to *temporal relations between media objects* in the multimedia systems
- Media objects
  - – Time dependent
  - – Time independent

Synchronization Issues

- Handling inherent dependencies of the media objects involved
- Dependencies include content, spatial & temporal relations between media objects

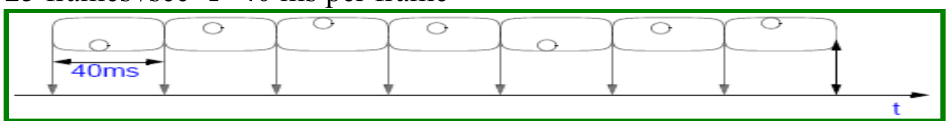
Content relations
- – Two graphics depend on same data (in spreadsheet), give different statistical representations

Temporal relations
- Temporal dependencies between media objects

Spatial Relations
- Known as layout relationships
- Layout frames used in desktop publishing

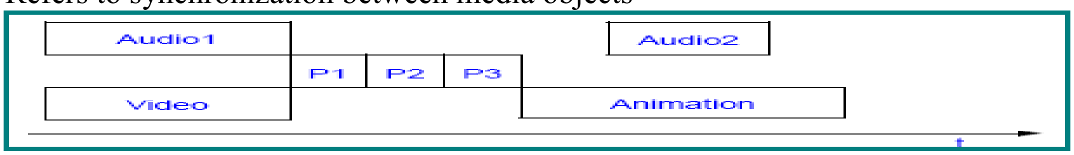
Intra & Inter-Object Synchronization

- <u>Intra-object synchronization</u>
  - Time relations between various presentation units of one time-dependent media object
  - Time relations between single frames of a vide sequence
    - 25 frames /sec ➔ 40 ms per frame



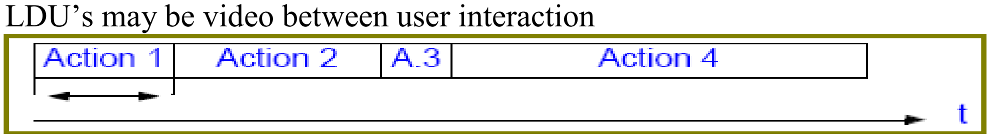
Video Sequence showing bouncing ball (Intra-object synchronization)

- Inter-Object Synchronization
  - Refers to synchronization between media objects



Temporal relationships between media objects

- Time dependent presentation units
  - Consisting of sequence of information units, known as Logical Data Units (LDU)
  - Various Granularity levels
    - Samples → Notes → Movements → Symphony
    - Frames → scenes → video
  - Selection of LDU's dependent on application
    - For a play application, symphony / movements can form LDU
  - Granularity implies a hierarchical decomposition of media objects
- For video, frames serves as LDU
- For Audio, group of samples (usually 512) serve as LDU
- For user generated media objects, duration of LDU's can be selected by user
- When LDU's vary in duration, streams become complex
  - Recording users interactions with computer
  - LDU's may be video between user interaction



- <u>Open LDU</u>
  - The LDU duration is not known in advance, because the duration of interaction is not known in advance

	Duration defined by	
	Capturing	User
Fixed Duration of LDU	Audio, Video	Animation, Timer
Variable or unknown Duration of LDU	Recorded interaction	User interaction

Types of LDU's

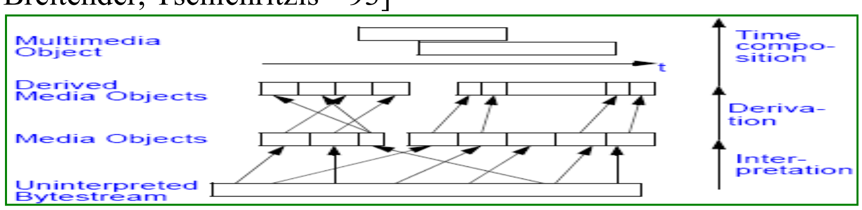
- *Classification of LDU* –<u>overview</u>
  - Open vs. closed
  - Fixed vs. variable duration
  - User defined vs. capturing related
  - Physical vs. Blocked

Live & Synthetic Synchronization

- <u>Live Synchronization</u>
  - At presentation, *reproduce the temporal relations as they existed in the capturing process*
  - Ex: Video conferencing application
- <u>Synthetic Synchronization</u>
  - *Temporal relations are artificially specified*
  - Ex: Online virtual tours
    - Media objects created independently are synchronized
    - A model for specification & manipulation of temporal synchronization conditions & operations is necessary
- Synthetic synchronization
  - *Specification Phase*
    - Define temporal relations between media objects
  - *Presentation Phase*
    - Run-time system presents data in a synchronized mode
- In Live synchronization,
  - Synchronization specification is implicitly specified during data capturing
  - User interactions are possible only at capturing

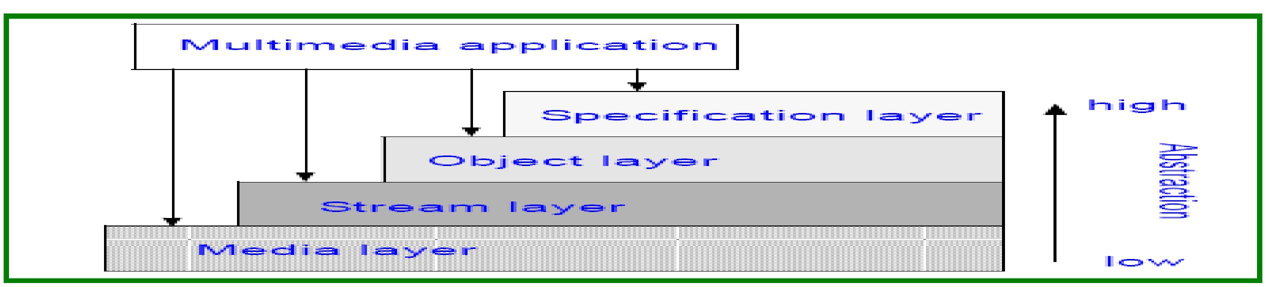
Synchronization Reference Model

- A reference model is needed
  - To understand various requirements for multimedia synchronization
  - Identify & Structure run-time mechanisms that support execution of synchronization
  - Identify interfaces between runtime mechanisms
  - Compare system solutions for multimedia synchronization systems
- Existing classification approaches
  - Physical Level, System Level & human level [Little &Ghafoor-' 90]
  - Intra-Stream & Inter-Stream [Little &Ghafoor-' 90]
  - Live-Stream & Synthetic Stream [Little &Ghafoor-' 90]
  - Mapping between synchronized multimedia object to uninterpretted byte stream [Gibbs, Breitender, Tschichritzis-' 93]



Mapping

- *Layered model*
  - Each layer implements synchronization mechanisms
  - Each layer is provided with appropriate interfaces
  - Each interface defines services, a means for the user to define his/her requirements
  - Each interface can be used by applications or next higher layer
  - Higher layers offers higher Programming & QoS Abstractions



Four layer reference model

Media Layer

- An application operates on a single continuous media stream, which is treated as a sequence of LDU's
- Abstraction offered at this layer is *device independent interface*, with operations like
  - *read(devicehandle, LDU), write(devicehandle, LDU)*
- *Application is responsible for intra-stream synchronization* by using flow-control mechanisms between a producing & consuming device
- Handling is difficult with multiple streams, distributed environment
- Inter-stream synchronization like, Lip-sync would be provided by interleaving audio & visual information in a LDU
- Media layer implementations can be
  - Simple
  - Interleaved

<u>Stream Layer</u>

- Operates on continuous media streams, as well as groups of media streams
  - In a group, all streams are presented in parallel
- Abstraction offered is the *notion of streams with timing parameters concerning QoS* for intra-stream & inter-stream synchronization between streams of a group
- Individual LDU's are not visible here
- Streams are executed in RTE by applications invokes stream layer services in NRTE
- Typical NRTE functions to manage streams/ groups are
  - Start(stream)
  - Stop(stream)
  - Create_group(stream)
  - Start(group)
  - Stop(group)
- The interaction of time-independent media or user interactions with continuous streams is performed by the attachment of events
  - *setcuepoint(stream/group, at, event)*
- Application using stream layer is responsible for starting, stopping & grouping the streams with the definition of required QoS

<u>Object Layer</u>

- Operates on all types of media & hides the difference between discrete & continuous media
- The abstraction offered to the application is that of *a complete, synchronized presentation*
  - Takes synchronization specification as input & is responsible for the correct schedule of the overall presentation
- Functions located at this layers are
  - Compute & execute compete presentation schedules
  - Initiate preparation actions, necessary for achieving correctly synchronized presentations
  - For inter-stream & intra-stream synchronization, it uses stream layer

<u>Specification Layer</u>

- Open layer
  - Does not contain an explicit interfaces
  - Contain applications & tools that allow to create synchronization specifications
  - Contain tools for converting specifications into object layer formats
  - Maps QoS requirements of user to object layer requirement

Layer	Interface Abstraction	Tasks
Specification	The tools performing the tasks of this layer have interfaces, the layer itself has no upper interface	to edit to format to map the quality of service values to the object layer
Object	Objects that hide types of enclosed media Synchronization Specification Media-oriented quality of service (In terms like allowed skew, allowed jitter)	Plan and coordinate presentation execution Initiate presentation of time-dependent media objects Initiate presentation of time-indepen-dent media objects
Stream	Streams and groups of streams Guarantees for intra-stream synchronization Guarantees for inter-stream synchroni-zation of streams in a group	Resource Reservation and scheduling of LDU processing
Media	Device independent access to LDUs. Guarantees for single LDU processing	File and device access

Presentation Requirements-

- Difficult to provide an objective measurement at users viewpoint on synchronization
  - Human perception varies from person to person
- Presentation requirements consists
  - Intra-object synchronization
  - Inter-object synchronization
- Blocking & Gap Problems !
  - Restricted blocking

Lip Synchronization

- Temporal relationship between audio & video for the case of humans speaking
  - Time difference between related audio & video LDU's known as *skew*

- Present the recorded presentation with artificially included skews, in steps of 40ms

Pointer Synchronization

Skew Values

Media		Mode, Application	QoS
Video	Animation	correlated	+/- 120 ms
	Audio	lip synchronization	+/- 80 ms
	Image	overlay	+/- 240 ms
		non-overlay	+/-500 ms
	Text	overlay	+/- 240 ms
		non-overlay	+/-500 ms
Audio	Animation	event correlation (e.g., dancing)	+/- 80 ms
	Audio	tightly coupled (stereo)	+/- 11 µs
		loosely coupled (dialogue mode with various participants)	+/- 120 ms
		loosely coupled (e.g., background music)	+/- 500 ms
	Image	tightly coupled (e.g., music with notes)	+/- 5 ms
		loosely coupled (e.g., slide show)	+/- 500 ms
	Text	Anmerkungen zu Text	+/- 240 ms
	Pointer	Audio Related to the Itlem	-500ms +750 ms

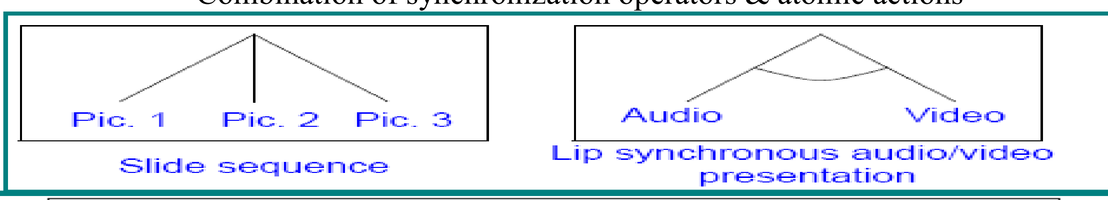
Synchronization Specification

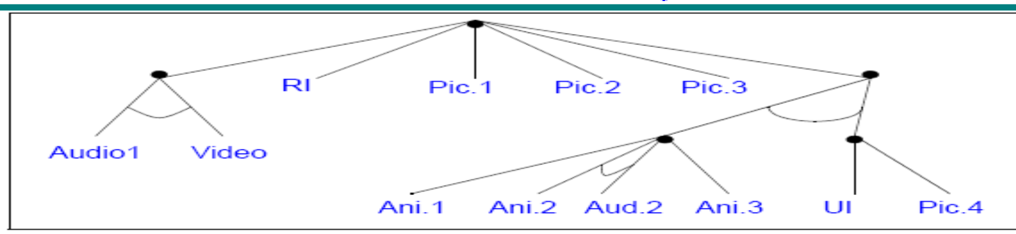
- Synchronization specification of multimedia object *describes all temporal dependencies of the included objects in the multimedia project*

- Synchronization specification comprised of
  - Intra-object synchronization specifications
  - QoS descriptions for intra-object synchronization
  - Inter-object synchronization specifications
  - QoS descriptions for inter-object synchronization
- Requirements of a specification method are
  - Shall support object consistency & maintenance of synchronization specification
  - Media objects should be kept as a single logical unit in the specification
  - All types of synchronization relations may be describable
  - Support of integration of time independent & dependent media objects
  - QoS requirements must be describable
  - Support of hierarchical levels of synchronization

Basic hierarchical specification

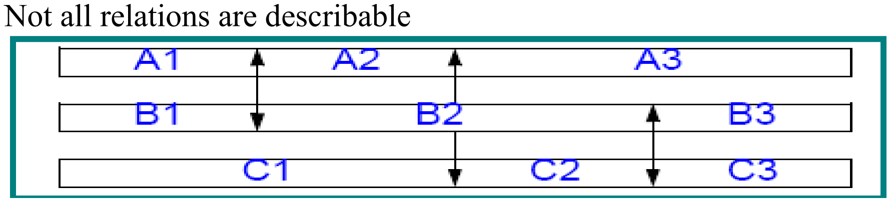
- Two main synchronization operations
  - Serial synchronization of actions
  - Parallel synchronization of operations
- Multimedia objects are regarded as a tree, nodes representing serial / parallel operations
- An action can be
  - Atomic
    - Handles presentation of a single media object, user input or delay
  - Compound
    - Combination of synchronization operators & atomic actions





Hierarchical specification of the application example (RI = Recorded Interaction, Pic. = Picture, Aud. = Audio, Ani. = Animation, UI = User Interaction)

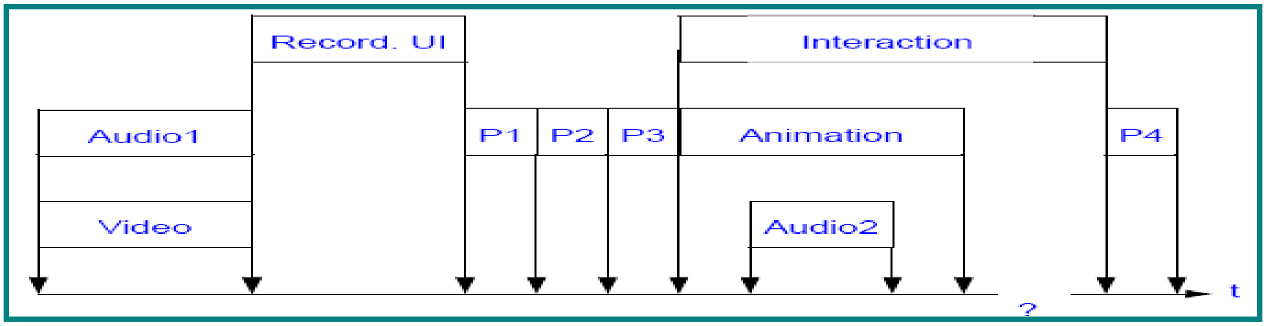
- Not all relations are describable



- No adequate abstractions for media object contents

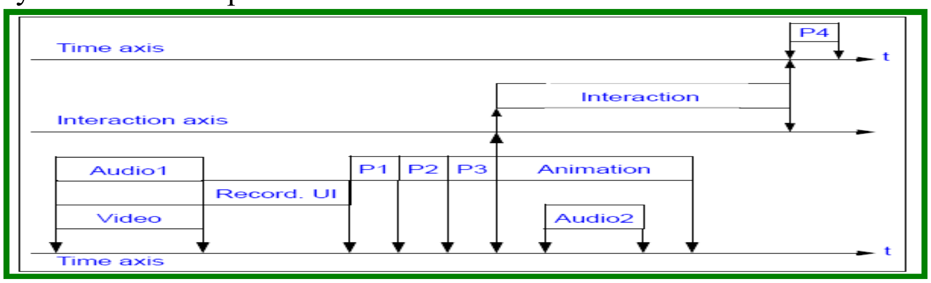
Based on Global Timer

- All single medium objects are attached to time axis that represents an abstraction of real time
- A world time is maintained
  - Each object maps it to it's local time
- Provides better abstractions from the internal structure of single medium objects & nested multimedia objects
  - Needs no knowledge about the frames to synchronize it with corresponding audio



Based on Virtual Axis

- Generalization of time axis approach
  - Possible to specify coordinate systems with user define units
  - Synchronization specification is based on these axes



Based on Reference points

- Time dependent single medium objects are regarded as sequences of closed LDUs
  - Start & stop of the presentation of media objects, start time of all sub units of the presentation are called *reference points*
- Synchronization is defined by connecting reference points
  - Set of connected reference points are called *synchronization point*

Event Based

- Presentation actions are initiated by synchronization events
  - Actions
    - Start a presentation
    - Stop a presentation
    - Prepare a presentation
  - Events
    - External
      - Generated by a timer
    - Internal
      - Generated by a time-dependent media object that reaches a specific LDU

Interval Based

- Enhanced interval based method used interval relations
  - 10 operators
- A slide show with slides Slide$_i$(i from 1 to n) & an audio object Audio can be specified by this model by

Slide $_1$ cobegin(0) Audio

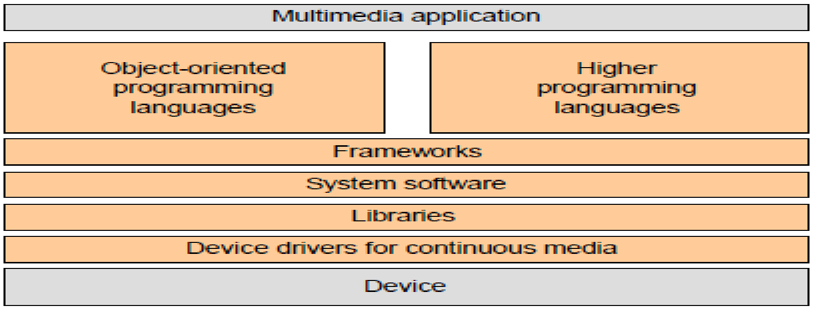
Slide $_i$ before(0) Slide $_{i+1}$ (for i from 1 to n-1)

This model does not handle skew specifications !!!

CH-10 Abstraction Levels

- Common operating system extensions try to solve this problem

• Different programming possibilities for accessing and representing multimedia data:



Libraries

• Processing of continuous media based on functions embedded in libraries
• Libraries differ in their degree of abstraction
• Example from IBM's early Audio Visual Connection (AVC):

```
acb.channel = AAPI_CHNA
acb.mode = AAPI_PLAY
...
aud_init(&acb) /* acb is the audio control block */
...
audrc = fab_open(AudioFullFileName,AAFB_OPEN,AAFB_EXNO, 0,&fab,0,0,0,0);
fork(START in PARALLEL)
aud_strt(&acb)
displayPosition(RelativeStarttime, Duration)
```

...

Libraries – OpenGL

2D and 3D graphics API developed by Silicon Graphics

• Basic idea: "write applications once, deploy across many platforms":
 _ PCs
 _ Workstations
 _ Super Computers
• Benefits:
 _ Stable
 _ Reliable and Portable
 _ Evolving
 _ Scalable (Features like Zoom, Rectangle handling ...)
 _ Well documented and easy to use
• Integrated with:
 _ Windows 95/NT/2000/XP
 _ UNIX X Window System

System Software

Device access becomes part of the operating system:

• Data as *time capsules* (file extensions)
 _ Each Logical Data Unit (LDU) carries in its time capsule its data type, actual value and valid life span
 _ Useful concept for video, where each frame has a valid life span of 40ms (rate of read access during a normal presentation)
 _ Presentation rate is changed for VCR (Video Cassette Recorder) functions like fast forward, slow forward or fast rewind by
  • Changing the presentation life span of a LDU
  • Skipping of LDUs or repetition of LDUs
• Data as *streams*
 _ A stream denotes the continuous flow of audio and video data between a source and a sink

_ Prior to the flow the stream is established equivalent to the setup of a connection in a networked environment

Programming Language Requirements

The high-level language should support parallel processing, because the processing of continuous data is

• controlled by the language through pure asynchronous instructions
• an integral part of a program through the identification of media

Different processes must be able to communicate through an inter-process communication mechanism, which must be able to:

• Understand a priori and/or implicitly specified time requirements (QoS parameters or
• Transmit the continuous data according to the requirements
• Initiate the processing of the received continuous process on time

Object-Oriented Approaches

Basic ideas of object-oriented programming is data encapsulation in connection with class and object definitions

• Abstract Type Definition (definition of data types through abstract interfaces)
• Class (implementation of a abstract data type)
• Object (instance of a class)

Other important properties of object-oriented systems are:

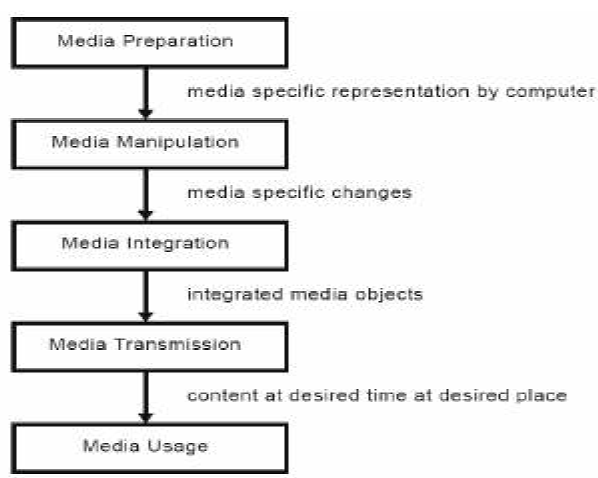
• Inheritance
• Polymorphism

Processing units as classes:

• Three main objects:
  _ Source objects
  _ Destination objects
  _ Combined source-destination objects allows the creation of data flow paths through connection of objects
• Multimedia object
  _ Basic Multimedia Classes (BMCs) /
    Basic Multimedia Objects (BMOs)
  _ Compound Multimedia Classes (CMCs) /
    Compound Multimedia Objects (CMO), which are compound of BMCs / BMOs and other CMCs/CMOs
  _ BMOs and CMOs can be distributed over different computer nodes

CH-11 Classification of Applications

• Applications are of central significance: interact and create, modify and use of multimedia data
• No general approach for applications – requirements differ with data
• Only possible: classification by series of transforms:



“*Media Food Chain*”

Preparation
Composition
Integration
Communication
Consumption
Entertainment

Media Preparation

Performed by Multimedia I/O devices & supporting hardware

Multimedia I/O hardware & software
Audio Support
Video support
Scanner devices
Recognition Devices
Tracking Devices

Media Composition

- Involves editing single media
- Editors available for
  - Text
  - Graphics
  - Image
  - Animation
  - Sound
  - Video

Media Integration

- Specifies relationship between various media elements to represent & manipulate a multimedia object
  - Multimedia Editors
  - Hypermedia / Hypertext editors
    - Apples Hypercard
  - Authoring tools

Concept → Design → Content collection →Assembly →Testing

Media Communication

Major Categories:
• Interactive Services
_ Conversation Services
_ News Transmission Services
_ Inquiry Services
_ Tele-Action Services
• Distribution Services
_ Pay-per-View
_ Near Video-on-Demand
_ True Video-on-Demand
Strong relationship to Communication /QoS
• Bandwidth
• Response Time
• Symmetric vs. Asymmetric Channels
• ...

Media Consumption

- Act of viewing, listening or feeling multimedia information
  - Browsing, Navigation, Displaying, Annotation
  - Books, Proceedings, News Paper
  - Kiosks

– Tele Shopping

Entertainment

- Virtual Reality
- Interactive TV
- VOD
- Games
- Tele Games

Media Usage

Large variety of applications, e.g.

- Electronic Newspapers and Books
- Kiosk Systems
- Tele Shopping
- Entertainment
    - _ Virtual Reality
    - _ Interactive Video
    - _ Interactive Audio
    - _ Computer Games

Thank you