**PRIST UNIVERSITY**
**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**SEMESTER -VI**

# LECTURE NOTES

# ON

# 11150H63 GRAPHICS AND MULTIMEDIA

**PREPARED BY**

**R.Bhavani**
**AP/CSE**

## DEPARTMENT OF COMPUTER SCIENCE AND ENGG

### 11150H73 – GRAPHICS AND MULTIMEDIA

**L T P C**

**3 1 0 4**

**UNIT I OUTPUT PRIMITIVES 9**

Basic − Line − Curve and ellipse drawing algorithms − Attributes − Two-Dimensional geometric transformations − Two-Dimensional clipping and viewing –Input techniques.

**UNIT II THREE-DIMENSIONAL CONCEPTS 9**

Three-Dimensional object representations − Three-Dimensional geometric andmodeling transformations − Three-Dimensional viewing − Hidden surface elimination− Color models − Animation.

**UNIT III MULTIMEDIA SYSTEMS DESIGN 9**

Multimedia basics − Multimedia applications − Multimedia system architecture −Evolving technologies for multimedia − Defining objects for multimedia systems −Multimedia data interface standards − Multimedia databases.

**UNIT IV MULTIMEDIA FILE HANDLING 9**

Compression and decompression − Data and file format standards − Multimedia I/Otechnologies − Digital voice and audio − Video image and animation − Full motionvideo − Storage and retrieval technologies.

**UNIT V HYPERMEDIA 9**

Multimedia authoring and user interface − Hypermedia messaging − Mobilemessaging − Hypermedia message component − Creating hypermedia message −Integrated multimedia message standards − Integrated document management −Distributed multimedia systems.

**L:45 T:15 Total : 60**

**TEXT BOOKS**

1. Donald Hearn and M. Pauline Baker, “Computer Graphics C Version”,Pearson Education, 2003.
2. Andleigh, P. K and Kiran Thakrar, “Multimedia Systems and Design”, PHI,2003.

**REFERENCES**

1. Judith Jeffcoate, “Multimedia in practice: Technology and Applications”, PHI,1998.
2. Foley, Vandam, Feiner and Huges, “Computer Graphics: Principles and Practice”, 2nd Edition, Pearson Education, 2003.

**UNIT I OUTPUT PRIMITIVES 9**

Basic − Line − Curve and ellipse drawing algorithms − Attributes − Two-Dimensional geometric transformations − Two-Dimensional clipping and viewing –Input techniques.

# 1.1 BASICS

Computer graphics is a sub-field of computer science and is concerned with digitally synthesizing and manipulating visual content. Although the term often refers to three-dimensional computer graphics, it also encompasses two-dimensional graphics and image processing

Today, we find computer graphics used routinely in such diverse areas as science, engineering, medicine, **business,** industry, government, art, entertainment, advertising, education, and training.

## COMPUTER AIDED DESIGN

A major **use** of computer graphics is in design processes, particularly for engineering and architectural systems, but almost all products are now computer designed. Generally referred to as CAD, computer-aided design methods are now routinely used in the design of buildings, automobiles, aircraft, watercraft, spacecraft, computers, textiles, and many, many other products.

## PRESENTATION GRAPHICS

Another major application **area** is presentation graphics, used to produce illustrations for **reports** or to generate **35-mm** slides or transparencies for use with projectors. Presentation graphics is commonly **used** to summarize financial, statistical, mathematical, scientific, and economic data for research reports, managerial reports, consumer information bulletins, and other types of reports. Workstation devices and **service bureaus** exist for converting screen displays into **35-mm** slides or overhead **transparencies** for **use** in presentations.

Typical examples of presentation graphics are bar charts, line graphs, surface graphs, pie **charts,** and other displays showing relationships between multiple parameters.

## COMPUTER ART

Computer graphics methods are widely used in both fine art and commercial art applications. Artists use a variety of computer methods, including special-purpose hardware, artist's paintbrush (such as Lumens), other paint packages (such as Pixelpaint and Superpaint),**.**

**VISUALIZATION**

Scientists, engineers, medical personnel, business analysts, and others often need to analyze large amounts of information or to study the behavior of certain processes. Numerical simulations carried out on supercomputers frequently produce data files containing thousands and even millions of data values..

## GRAPHICAL USER INTERFACES

It is common now for software packages to provide a graphical interface. A major component of a graphical interface is a window manager that allows a user to display multiple-window areas. Each window can contain a different process that can contain graphical or nongraphical displays. To make a particular window active, we simply click in that window using an interactive pointing device.Interfaces also display menus and icons for fast selection of processing options or parameter values.

## GRAPHIC SYSTEMS

## VIDEO DISPLAY DEVICES

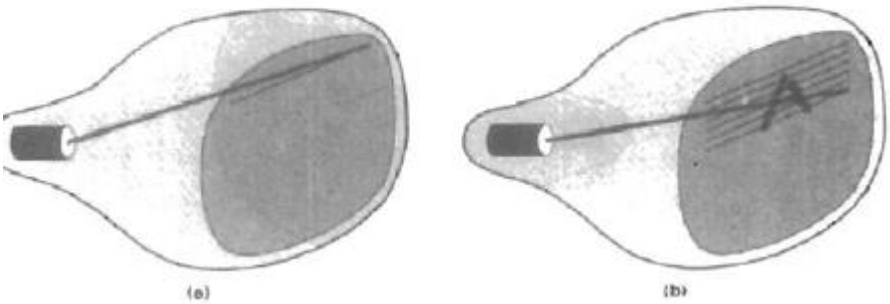
Typically, the primary output device in a graphics system is a video monitor The operation of most video monitors is based on the standard cathode-ray tube (CRT) design.

the basic operation of a CRT  A beam of electrons (cathode rays), emitted by an electron gun, passes through focusing and deflection systems that direct the beam toward specified positions on the phosphomted screen. The phosphor then emits a small spot of light at each position contacted by the electron beam. Because the light emitted by the phosphor fades very rapidly, some method is needed for maintaining the screen picture. One way to keep the phosphor glowing is to redraw the picture repeatedly by quickly directing the electron beam back over the same points. This type of display is called a refresh
CRT.

## Raster-Scan Displays

The most common type of graphics monitor employing a CRT is the raster-scan display, based on television technology.

In a raster-scan system, the electron beam is swept across the screen, one row at a time from top to bottom. As the electron beam moves across each row, the beam intensity is turned on and off to create a pattern of illuminated spots. Picture definition is stored in a memory area called the refresh buffer or frame buffer. This memory area holds the set of intensity values for all the screen points. Stored intensity values are then retrieved from the refresh buffer and "painted" on the screen one row (scan line) at a time .Each screen point is referred to as a pixel or pel (shortened fonns of picture element). The capability of a raster-scan system to store intensity information for each screen point makes it well suited for the realistic displav of scenes containing subtle shading and color patterns.




## Random-Scan Displays

When operated as a random-scan display unit, a CRT has the electron beam directed only to the parts of the screen where a picture is to be drawn. Randomscan monitors draw a picture one line at a time and for this reason are also referred to as vector displays (or stroke-writing or calligraphic diisplays). The component lines of a picture can be drawn and refreshed by a random-scan system in any specified order

Refresh rate on a random-scan system depends on the number of lines to be displayed. Picture definition is now stored as a set of line drawing commands in an area of memory referred to as the refresh display file. Sometimes the refresh display file is called the display list, display program, or simply the refresh buffer.

To display a specified picture, the system cycles through the set of commands in the display file, drawing each component line in turn. After all line drawing commands have been processed, the system cycles back to the first line command in the list. Random-scan displays are designed to draw all the component lines of a picture 30 to 60 times each second. High quality vector systems arecapable of handling approximately 100,000 "short" lines at this refresh rate. When a small set of lines is to be displayed, each rrfresh cycle is delayed to avoid refresh rates greater than 60 frames per second. Otherwise, faster refreshing of the set of lines could bum out the phosphor.

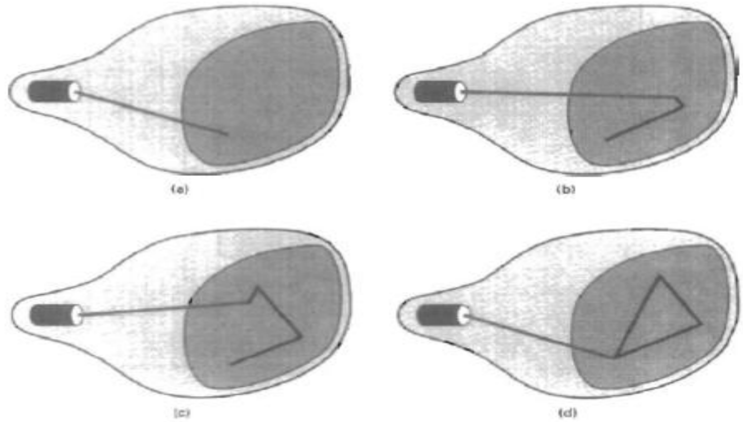



*Figure 2-9*
A random-scan system draws the component lines of an object in any order specified

## OUTPUT PRIMITIVES

1. A picture can be described in several ways. In raster display, a picture is completely specified by the set of intensities for the pixel positions in the display. At the other extreme, we can describe a picture as a set of complex objects, such as trees and terrain or furniture and walls, positioned at specified coordinate locations within the scene.

2. Shapes and colors of the objects can be described internally with pixel arrays or with sets of basic geometric structures, such as straight line segments and polygon color areas. The scene is then displayed either by loading the pixel arrays into the frame buffer or by scan converting the basic geometric-structure specifications into pixel patterns.

## POINTS AND LINES

1 Point plotting is accomplished by converting a single coordinate position furnished by an application program into appropriate operations for the output device in use.

The formula for the point to be plotted is putpixel(x,y,color)

## 1.2 LINE-DRAWING ALGORITHMS

Design of Line and Circle Algorithms
Basic Math Review

**Slope-Intercept Formula For A Line**

Given a third point on the line:

$P = (X, Y)$

$\text{Slope} = (Y - Y_1)/(X - X_1)$

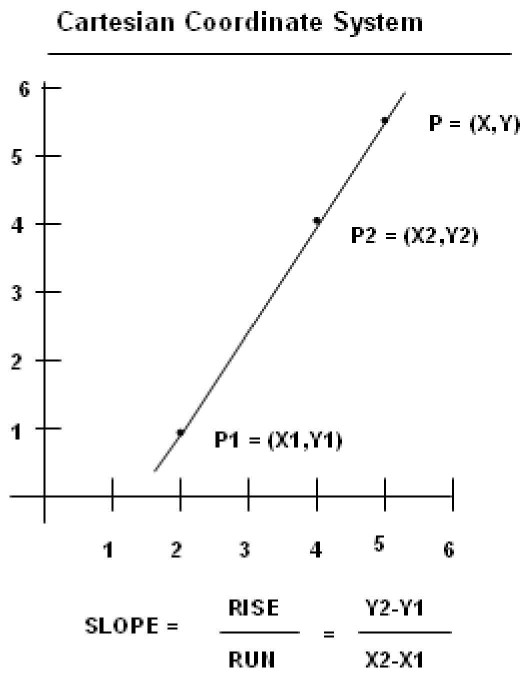
$= (Y_2 - Y_1)/(X_2 - X_1)$

Solving For Y

$Y = [(Y_2 - Y_1)/(X_2 - X_1)]X + [-(Y_2 - Y_1)/(X_2 - X_1)]X_1 + Y_1$

or

$Y = mx + b$




3. Line drawing is accomplished by calculating intermediate positions along the line path between specified end points.

Precise definition of line drawing

4. Given two points P and Q in the plane, both with integer coordinates, determine which pixels on a raster screen should be on in order to make a picture of a unit-width line segment starting from P and ending at Q.

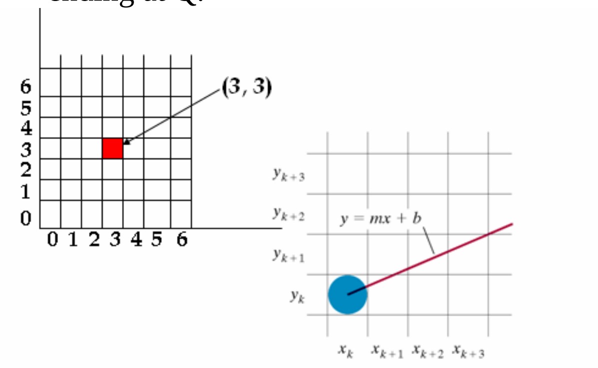



Figure 3-10

A section of the screen showing a pixel in column $x_k$ on scan line $y_k$ that is to be plotted along the path of a line segment with slope $0 < m < 1$.

The thinnest line is of one-pixel wide. We will concentrate on drawing a line of 1 pixel resolution.
The Cartesian slope-intercept equation for a straight line is

$$y = m.\ x + b$$

m is the slope of the line and b is the y intercept.

Given the endpoints of a line segment.

$$m = y2 - y1 / x2 - x1$$
$$b = y1 - m.x1$$

5. Also for any given interval $\Delta x$ along a line, we can compute the corresponding y interval $\Delta y$ from
$$\Delta y = m.\ x$$
6. Similarly we can obtain the x interval $\Delta x$ corresponding to a specified $\Delta y$ as
$$\Delta x = \Delta y / m$$
7. These equations form the basis for determining deflection voltages in analog devices.
8. Also , for any given x interval $\Delta x$ along a line, we can compute the corresponding y interval $\Delta y$ from
$$\Delta y = m.\ \Delta\ x$$
9. These equations form the basis for determining deflection voltages in analog devices.

10. On Raster systems, lines are plotted with pixels, and step sizes in the horizontal and vertical directions are constrained by pixel separations. Hence we ought to “sample” a line at discrete positions and determine the nearest pixel to the line at each sampled position

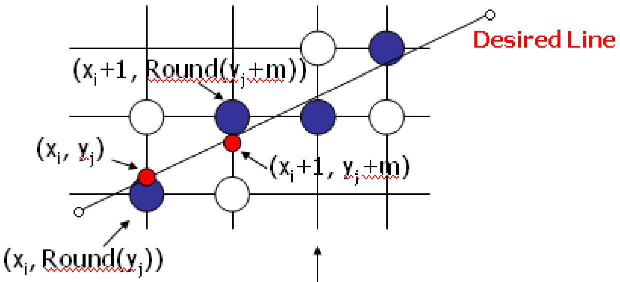
## DDA ALGORITHM

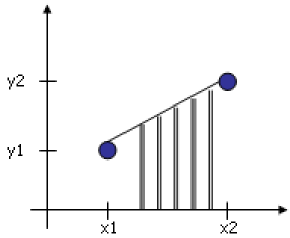
The digital differential analyzer (DDA) samples the line at unit intervals in one coordinate corresponding integer values nearest the line path of the other coordinate.

The following is thus the basic incremental scan-conversion(DDA) algorithm for line drawing

```
for x from x0 to x1
Compute y=mx+b
Draw_fn(x, round(y))
```

Major deficiency in the above approach :
Uses floats
Has rounding operations



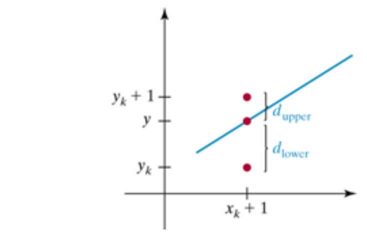

## Bresenham’s Line Algorithm

- An accurate, efficient raster line drawing algorithm developed by Bresenham, scan converts lines using only *incremental integer* calculations that can be adapted to display circles and other curves.

- Keeping in mind the symmetry property of lines, lets derive a more efficient way of drawing a line.

Starting from the left end point (*x*0,*y*0) of a given line , we step to each successive column (x position) and plot the pixel whose scan-line y value closest to the line path

Assuming we have determined that the pixel at (*xk*,*yk*) is to be displayed, we next need to decide which pixel to plot in column *xk*+1.




Choices are*(xk +1, yk)* and *(xk+1, yK+1)*

$d1 = y - yk = m(xk + 1) + b - yk$

$d2 = (yk + 1) - y = yk + 1 - m(xk + 1) - b$

- The difference between these 2 separations is

$d1\text{-}d2 = 2m(xk + 1) - 2\ yk + 2b - 1$

- A decision parameter *pk* for the *kth* step in the line algorithm can be obtained by rearranging above equation so that it involves only *integer calculations*

- Define $Pk = \Delta x\ (d1\text{-}d2) = 2\Delta yxk\text{-}2\ \Delta xyk + c$

- The sign of *P*k is the same as the sign of *d1-d2*, since $\Delta x > 0$.

*Parameter c* is a constant and has the value $2\Delta y + \Delta x(2b\text{-}1)$
(independent of pixel position)

- If *pixel at yk* is closer to line-path than pixel at *yk +1*

*(i.e, if d1 < d2)* then *pk* is negative. We plot lower pixel in such a case. Otherwise , upper pixel will be plotted.

- At step *k + 1*, the decision parameter can be evaluated as,

$pk{+}1 = 2\Delta yxk{+}1 - 2\Delta xyk{+}1 + c$

- Taking the difference of pk+ 1 and pk we get the following.

$pk{+}1 - pk = 2\Delta y(xk{+}1\text{-}\ xk)\text{-}2\Delta x(yk{+}1 - yk)$

- But, *xk+1 = xk +1*, so that

$pk{+}1 = pk + 2\Delta y - 2\ \Delta x(yk{+}1 - yk)$

- Where the term *yk+1-yk* is either 0 or 1, depending on the sign of *parameter pk*

- The first parameter *p0* is directly computed

$p0 = 2\ \Delta yxk - 2\ \Delta xyk + c = 2\ \Delta yxk - 2\ \Delta y + \Delta x\ (2b\text{-}1)$

- Since *(x0,y0)* satisfies the line equation , we also have

$y0 = \Delta y/\ \Delta x * x0 + b$

- Combining the above 2 equations , we will have

$p0 = 2\Delta y - \Delta x$

The constants $2\Delta y$ and $2\Delta y\text{-}2\Delta x$ are calculated once for each time to be scan converted

- So, the arithmetic involves only integer addition and subtraction of 2 constants

## ALGORITHM

Input the two end points and store the left end point in (x0,y0)
Load (x0,y0) into the frame buffer (plot the first point)
Calculate the constants Δx, Δy, 2Δy and 2Δy-2Δx and obtain the starting value for the decision parameter as

$p0 = 2\Delta y\text{-}\ \Delta x$

At each xk along the line, starting at k=0, perform the following test:
If pk < 0 , the next point is (xk+1, yk) and

$p_{k+1} = p_k + 2\Delta y$

Otherwise Point to plot is $(x_k+1, y_k+1)$

$p_{k+1} = p_k + 2\Delta y - 2\Delta x$

Repeat step 4 (above step) $\Delta x$ times

## 1.3 CIRCLE/ CURVE -GENERATING ALGORITHMS

### Properties of a circle:

- A circle is defined as a set of points that are all the given distance *(xc,yc)*. This distance relationship is expressed by the pythagorean theorem in Cartesian coordinates as

  $(x - x_c)^2 + (y - y_c)^2 = r^2$
- We could use this equation to calculate the points on the circle circumference by stepping along x-axis in unit steps from $x_c - r$ to $x_c + r$ and calculate the corresponding y values at each position as

  $y = y_c \pm (r^2 - (x_c - x)^2)^{1/2}$
- This is not the best method:
  - *Considerable amount of computation*
  - *Spacing between plotted pixels is not uniform*

**Polar co-ordinates for a circle**

▪We could use polar coordinates r and θ,

$x = x_c + r\cos\theta \qquad y = y_c + r\sin\theta$

▪A fixed angular step size can be used to plot equally spaced points along the circumference

▪A step size of 1/r can be used to set pixel positions to approximately 1 unit apart for a continuous boundary

▪But, note that circle sections in adjacent octants within one quadrant are symmetric with respect to the 45 deg line dividing the to octants

▪Thus we can generate all pixel positions around a circle by calculating just the points within the sector from x=0 to x=y

▪This method is still computationally expensive

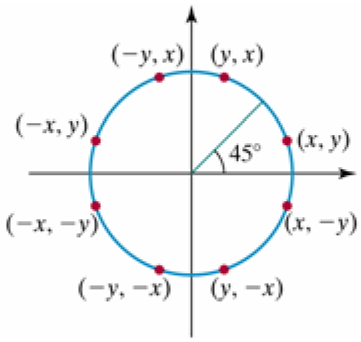



Figure 3-18

Symmetry of a circle. Calculation of a circle point (x, y)in one octant yields the circle points shown for the other seven octants.

### Midpoint Circle Algorithm

- We will first calculate pixel positions for a circle centered around the origin (0,0). Then, each calculated position (x,y) is moved to its proper screen position by adding xc to x and yc to y

- Note that along the circle section from x=0 to x=y in the first octant, the slope of the curve varies from 0 to -1

- Circle function around the origin is given by

  $f_{circle}(x,y) = x^2 + y^2 - r^2$

- Any point (x,y) on the boundary of the circle satisfies the equation and circle function is zero

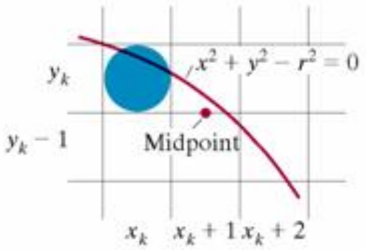
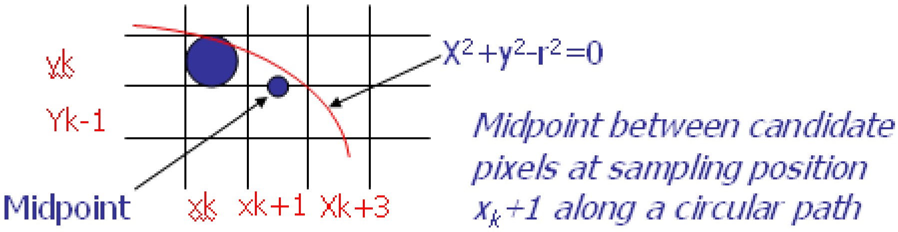



Figure 3-19
Midpoint between candidate pixels at sampling position $x_k + 1$ along a circular path.

## Midpoint Circle Algorithm

- For a point in the interior of the circle, the circle function is negative and for a point outside the circle, the function is positive
- Thus,
  - fcircle(x,y) < 0 if (x,y) is inside the circle boundary
  - fcircle(x,y) = 0 if (x,y) is on the circle boundary
  - fcircle(x,y) > 0 if (x,y) is outside the circle boundary




- Assuming we have just plotted the pixel at *(xk,yk)* , we next need to determine whether the pixel at position *(xk + 1, yk-1)* is closer to the circle
- Our decision parameter is the circle function evaluated at the midpoint between these two pixels

*pk = fcircle (xk +1, yk-1/2) = (xk +1)2 + (yk -1/2)2 – r2*

If *pk < 0* , this midpoint is inside the circle and the pixel on the scan line *yk* is closer to the circle boundary. Otherwise, the
mid position is outside or on the circle boundary, and we select the pixel on the scan line *yk-1*

- Successive decision parameters are obtained using incremental calculations

*Pk+1 = fcircle(xk+1+1, yk+1-1/2)*
*= [(xk+1)+1]2 + (yk+1 -1/2)2 –r2*

**OR**

*Pk+1 = Pk+2(xK+1) + (yK+12 – yk2) – (yk+1- yk)+1*

Where *yk+1* is either *yk* or *yk-1*, depending on the sign of pk

- Increments for obtaining *Pk+1*:

*2xk+1+1 if pk is negative*
*2xk+1+1-2yk+1 otherwise*

- Note that following can also be done incrementally:

*2xk+1 = 2xk +2*
*2 yk+1 = 2yk – 2*

- At the start position *(0,r)* , these two terms have the values 2 and *2r-2* respectively
- Initial decision parameter is obtained by evaluating the circle function at the start position *(x0,y0) = (0,r)*
  *p0 = fcircle(1, r-1/2) = 1+ (r-1/2)2-r2*

OR

*P0 = 5/4 -r*

- If radius r is specified as an integer, we can round p0 to *p0 = 1-r*

**The actual algorithm**

1: Input radius r and circle center (xc,yc) and obtain the first point on the circumference of the circle centered on the origin as

(x0,y0) = (0,r)

2: Calculate the initial value of the decision parameter as

P0 = 5/4 - r

3: At each xk position starting at k = 0 , perform the following test:

If pk < 0 , the next point along the circle centered on (0,0) is (xk+1, yk) and

pk+1 = pk + 2xk+1 + 1

Otherwise the next point along the circle is (xk+1, yk-1) and

pk+1 = pk + 2xk+1 +1 -2yk+1

Where 2xk+1 = 2xk+2 and 2yk+1 = 2yk-2

4: Determine symmetry points in the other seven octants

5: Move each calculated pixel position (x,y) onto the circular path centered on (x,yc) and plot the coordinate values

x = x+ xc , y= y+ yc

6: Repeat steps 3 through 5 until x >= y

## ELLIPSE-GENERATING ALGORITHMS

**Properties of Ellipses**

An ellipse is defined as the set of points such that the sum of the distances from two **fixed** positions (foci) is the **same** for **all** points (Fig. ***b17).***

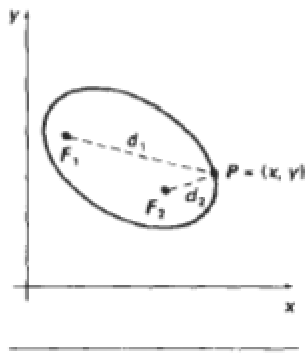



*Figure 3-17*
Ellipse generated about foci

**If** the distances to the two foci from any point P = ***(x,* y)** on the ellipse are labeled **dl** and **d2,** then the general equation of an ellipse can be stated as

***d,*** + ***d,*** = constant (3-32)

Expressing distances $d_1$ and $d_2$ in terms of the focal coordinates $\mathbf{F}_1 = (x_1, y_1)$ and $\mathbf{F}_2 = (x_2, y_2)$, we have

$$\sqrt{(x - x_1)^2 + (y - y_1)^2} + \sqrt{(x - x_2)^2 + (y - y_2)^2} = \text{constant} \qquad (3\text{-}33)$$

By squaring this equation, isolating the remaining radical, and then squaring again,

we can rewrite the general ellipse equation in the form

***Ax2*** + **By2** + ***Cxy*** + ***Dx*** + ***Ey*** + **F** = **0** ***(3-34)***

where the coefficients ***A,*** B, C, ***D, E,*** and F are evaluated l in terms of the focal coordinates and the dimensions of the major and minor axes of the ellipse.
The major axis is the straight line segment extending from one side of the ellipse to the other through the foci.The minor axis spans the shorter dimension of the ellipse, bisecting the major axis at the halfway position (ellipse center) between the twofoci.An interactive method for specifying an ellipse in an arbitrary orientation is to input the two foci and a point on the ellipse boundary.
With these three coordinate positions, we can evaluate the constant in Eq. 3.33.
Then, the coefficients inEq. 3-34 can be evaluated and used to generate pixels along the ellipticalpath.

$$\left(\frac{x - x_c}{r_x}\right)^2 + \left(\frac{y - y_c}{r_y}\right)^2 = 1 \qquad (3\text{-}35)$$

Using polar coordinates r and **0.** we can also describe the ellipse in standard position with the parametric equations:

$$x = x_c + r_x \cos\theta$$
$$y = y_c + r_y \sin\theta \qquad (3\text{-}36)$$

**Midpoint Ellipse Algorithm**

1. Input $r_x$, $r_y$, and ellipse center $(x_c, y_c)$, and obtain the first point on an ellipse centered on the origin as

$$(x_0, y_0) = (0, r_y)$$

2. Calculate the initial value of the decision parameter in region 1 as

$$p1_0 = r_y^2 - r_x^2 r_y + \frac{1}{4} r_x^2$$

3. At each $x_k$ position in region 1, starting at $k = 0$, perform the following test: If $p1_k < 0$, the next point along the ellipse centered on (0, 0) is $(x_{k+1}, y_k)$ and

$$p1_{k+1} = p1_k + 2r_y^2 x_{k+1} + r_y^2$$

Otherwise, the next point along the circle is $(x_k + 1, y_k - 1)$ and

$$p1_{k+1} = p1_k + 2r_y^2 x_{k+1} - 2r_x^2 y_{k+1} + r_y^2$$

with

$$2r_y^2 x_{k+1} = 2r_y^2 x_k + 2r_y^2, \qquad 2r_x^2 y_{k+1} = 2r_x^2 y_k - 2r_x^2$$

and continue until $2r_y^2 x \geq 2r_x^2 y$.

4. Calculate the initial value of the decision parameter in region 2 using the last point $(x_0, y_0)$ calculated in region 1 as

$$p2_0 = r_y^2 \left(x_0 + \frac{1}{2}\right)^2 + r_x^2 (y_0 - 1)^2 - r_x^2 r_y^2$$

5. At each $y_k$ position in region 2, starting at $k = 0$, perform the following test: If $p2_k > 0$, the next point along the ellipse centered on (0, 0) is $(x_k, y_k - 1)$ and

$$p2_{k+1} = p2_k - 2r_x^2 y_{k+1} + r_x^2$$

Otherwise, the next point along the circle is $(x_k + 1, y_k - 1)$ and

$$p2_{k+1} = p2_k + 2r_y^2 x_{k+1} - 2r_x^2 y_{k+1} + r_x^2$$

using the same incremental calculations for $x$ and $y$ as in region 1.

6. Determine symmetry points in the other three quadrants.
7. Move each calculated pixel position $(x, y)$ onto the elliptical path centered on $(x_c, y_c)$ and plot the coordinate values:

$$x = x + x_c, \qquad y = y + y_c$$

8. Repeat the steps for region 1 until $2r_y^2 x \geq 2r_x^2 y$.

## 1.4 ATTRIBUTES

### LINE ATTRIBUTES

Basic attributes of a straight line segment are its type, its width, and its color. In some graphics packages, lines can also be displayed **using** selected pen or brush options.

### Line Type

line-type attribute - solid lines, dashed lines, and dotted lines.

We modify a line drawing algorithm to generate such lines by setting the length and spacing of displayed solid sections along the line path.

A dashed line could be displayed by generating an interdash spacing that is equalto the length of the solid sections. Both the length of the dashes and the interdashspacing are often specified as user options. **A** dotted line can be displayed bygenerating very short dashes with the spacing equal to or greater than the dash size. Similar methods are used to produce other line-type variations.

To set line type attributes in a **PHICS** application program, a user invokes
the function
setLinetype **(It)**

where parameter I t is assigned a positive integer value of 1,2,3, or 4 to generate
lines that are, respectively, solid, dashed, dotted, or dash-dotted.

## Line Width

We set the line-width attribute with the command: Line-width parameter lr. is assigned a positive number to indicate the relative width of the line to be displayed. A value of *1* specifies a standard-width line. On.

For lines with slope magnitude greater than 1, we can plot thick lines withhorizontal spans, alternately picking up pixels to the right and left of the linepath.

Problem with implementing width options using horizontal or vertical pixel spans is that the method produces **lines** whose ends are horizontal or vertical regardless of the slope of the line. **This** effect is more noticeable with very thick lines. We can adjust the shape of the **line** ends to give them a better appearance by adding line caps

One kind of line cap is the butt cap obtained by adjusting the end positions of the component parallel **lines** so that the thick line is displayed with square ends that are perpendicular to the line path. If the specified line has slope m, the square end of the thick line has slope **- l / m .**

Another line cap is the round cap obtained by adding a filled semicircle to eachbutt cap. The circular arcs are centered on the line endpoints and have a diameterequal to the line thickness.

A third type of line cap is the projecting square cap.Here, we simply extend the line and add butt caps that are positioned one-half ofthe line width beyond the specified endpoints.

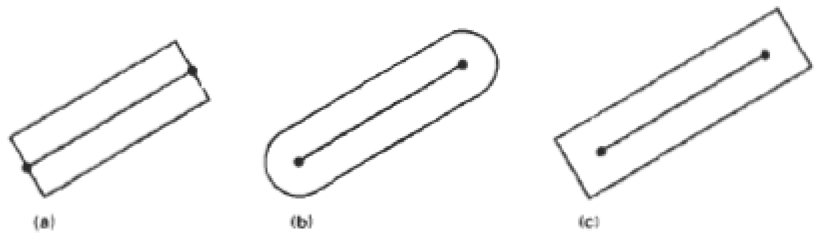



*Figure 4-5*
Thick lines drawn with (a) butt caps, (b) round caps, and (c) projecting square caps.

We can generate thick polylines that are smoothly joined at the cost of additional processing at the segment endpoints.

A miter join is accomplished by extending the outer boundaries of each of the two linesuntil they meet.

A round join is produced by capping the connection between the
two segments with a circular boundary whose diameter is equal to the linewidth.

And a bevel join is generated by displaying the line segments with butt
caps and filling in the triangular gap where the segments meet.

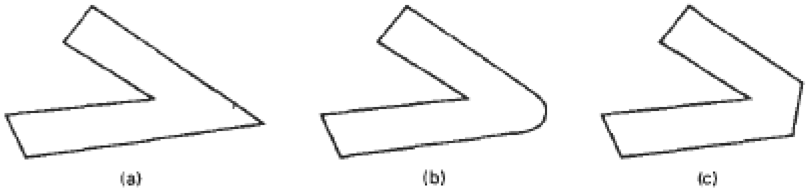



*Figure 4-6*
Thick line segments connected with (a) miter join, (b) round join, and (c) bevel join.

## Pen and Brush Options

lines can be displayed with pen or brush selections. ***Op***tions in this category include shape, size, and pattern.

These shapes can be stored in a pixel mask that identifies the array of pixel positions that are to be set along the line path. Lines generated with pen (or brush) shapes canbe displayed in various widths by changing the size of the mask.

**Line Color**

When a system provides color (or intensity) options, a parameter giving the current
color index is included in the list of system-attribute values. A polyline routinedisplays a line in the current color by setting this color value in the framebuffer at pixel locations along the line path using the **setpixel** procedure.

The number of color choices depends on the number of bits available per pixel in theframe buffer.We set the line color value in **PHlCS** with the function

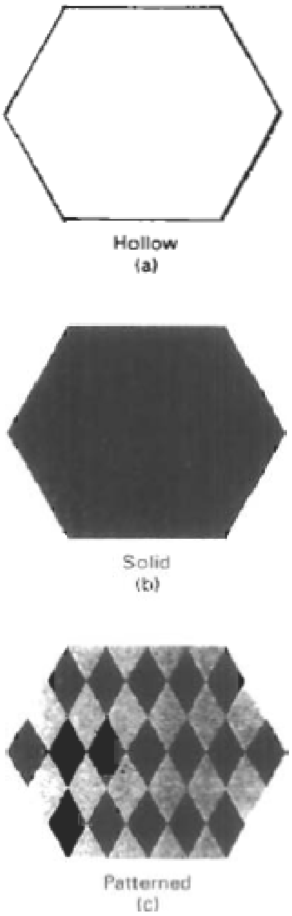
```
setPolylineColourIndex (lc)
```

## **CURVE** ATTRIBUTES

Parameters for curve attributes are the **same** as those for line segments. We can
display curves with varying colors, widths, dotdash patterns, and available pen
or brush options.

## AREA FILL ATTRIBUTES

.




igure 4-18

## 4-4 AREA-FILL ATTRIBUTES

Options for filling a defined region include a choice between a solid color or a patterned fill and choices for the particular colors and patterns. These fill options can be applied to polygon regions or to areas defined with curved boundaries, depending on the capabilities of the available package. In addition, areas can be painted using various brush styles, colors, and transparency parameters.

### Fill Styles

Areas are displayed with three basic fill styles: hollow with a color border, filled with a solid color, or filled with a specified pattern or design. A basic fill style is selected in a PHIGS program with the function

```
setInteriorStyle (fs)
```

Values for the fill-style parameter `fs` include *hollow*, *solid*, and *pattern* (Fig. 4-18). Another value for fill style is *hatch*, which is used to fill an area with selected hatching patterns—parallel lines or crossed lines—as in Fig. 4-19. As with line attributes, a selected fill-style value is recorded in the list of system attributes and applied to fill the interiors of subsequently specified areas. Fill selections for parameter `fs` are normally applied to polygon areas, but they can also be implemented to fill regions with curved boundaries.

Hollow areas are displayed using only the boundary outline, with the interior color the same as the background color. A solid fill is displayed in a single color up to and including the borders of the region. The color for a solid interior or for a hollow area outline is chosen with

```
setInteriorColourIndex (fc)
```

where fill-color parameter `fc` is set to the desired color code. A polygon hollow fill is generated with a linedrawing routine as a closed polyline

### Pattern Fill

We select fill patterns with

```
setInteriorStyleIndex (pi)
```

where pattern index parameter `pi` specifies a table position. For example, the following set of statements would fill the area defined in the `fillArea` command with the second pattern type stored in the pattern table:

```
setInteriorStyle (pattern);
setInteriorStyleIndex (2);
fillArea (n, points);
```

Separate tables are set up for hatch patterns. If we had selected *hatch* fill for the interior style in this program segment, then the value assigned to parameter `pi` is an index to the stored patterns in the hatch table.

For fill style *pattern*, table entries can be created on individual output devices with

```
setPatternRepresentation (ws, pi, nx, ny, cp)
```

**TABLE 4-3**
A WORKSTATION PATTERN TABLE WITH TWO ENTRIES, USING THE COLOR CODES OF TABLE 4-1

| *Index* *(pi)* | *Pattern* *(cp)* |
|---|---|
| 1 | $\begin{bmatrix} 4 & 0 \\ 0 & 4 \end{bmatrix}$ |
| 2 | $\begin{bmatrix} 2 & 1 & 2 \\ 1 & 2 & 1 \\ 2 & 1 & 2 \end{bmatrix}$ |

# CHARACTER ATTRIBUTES

The appearance of displayed characters is controlled by attributes such as font, size, color, and orientation. Attributes can be set for entire character strings (text) and for individual characters defined as marker symbols.

## TEXT ATTRIBUTES

There are a great many text options that can be made available to graphics programmers.

First of **all,** there is the choice of font (or typeface), which is a set of characters with a particular design style such as New York, Courier, Helvetica, London, 'Times Roman, and various special symbol groups.

The width only of text can be set with the function

```
setCharacterExpansionFactor (cw)
```

where the character-width parameter cw is set to a positive real value that scales the body width of characters. Text height is unaffected by this attribute setting. Examples of text displayed with different character expansions is given in Fig. 4-27.

Spacing between characters is controlled separately with

```
setCharacterSpacing (cs)
```

where the character-spacing parameter cs can be assigned any real value. The value assigned to cs determines the spacing between character bodies along print lines. Negative values for cs overlap character bodies; positive values insert space to spread out the displayed characters. Assigning the value 0 to cs causes text to be displayed with no space between character bodies. The amount of spacing to be applied is determined by multiplying the value of cs by the character height (distance between baseline and capline). In Fig. 4-28, a character string is displayed with three different settings for the character-spacing parameter.

width0.5

width 1.0

width 2.0

*Figure 4-27*
The effect of different character-width settings on displayed text.

## ANTIALIASING

- Displayed primitives generated by the raster algorithms have a jagged, or stairstep, appearance because the sampling process digitizes coordinatepoints on an object to discrete integer pixel positions. This distortion of information due to low-frequency sampling (undersampling) is called aliasing.

We can improve the appearance of displayed raster lines by applying antialiasingmethods that compensate for the undersampling process.To avoid losing information from such periodic objects, we need to set the sampling frequency to at least twice that of the highest frequency occurring in the object, referredto as the Nyquist sampling frequency (or Nyquist sampling rate) fs:

$$f_s = 2f_{max}$$

Another way to state this is that the sampling interval should be no larger than one-half the cycle interval (called the Nyquist sampling interval). For x-interval sampling, the Nyquist sampling interval Ax, is

$$\Delta x_s = \frac{\Delta x_{cycle}}{2}$$

where $\Delta x_{cycle} = 1/f_{max}$

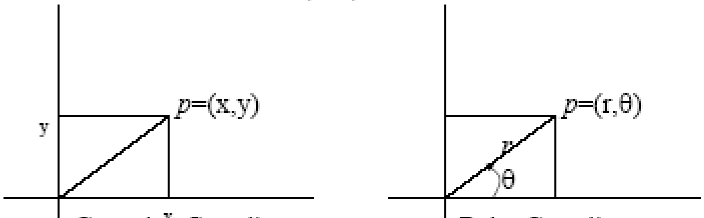
## 1.5 TWO DIMENSIONAL GEOMETRIC TRANSFORMATIONS

### Object Manipulation

- Graphics primitives allows us to draw a variety of pictures, images and graphs
- In many application there is a also need to manipulate the displays
  - Rearranging the display
  - Changing the size of the different objects
- This is accomplished by using geometric transformations

# Coordinate Systems

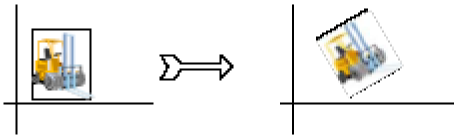
- Objects can be represented by using different coordinate systems
  - Cartesian coordinates (x,y)
  - Polar coordinates =(r,θ)



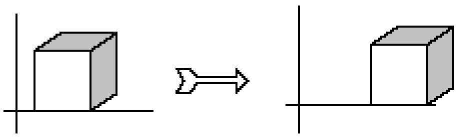

Cartesian Coordinates    Polar Coordinates

# Basic Geometric Transformation

- The basic geometric transformation are:
  - Translation
  - Scaling
  - Rotation

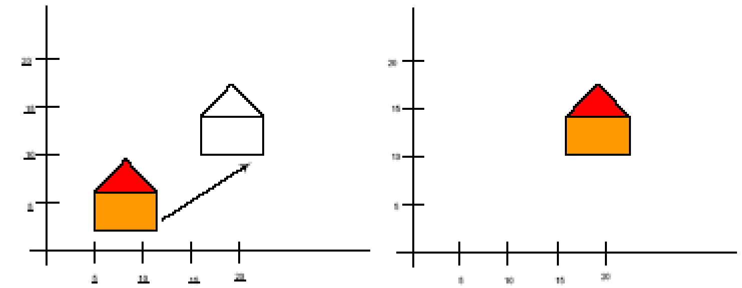


- Other
  - Reflection
  - Shear

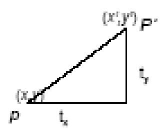


# Translation

- Purpose: to move an object along a line from one location to another location



- We translate a two dimensional point p=(x,y) by a translation distance t=($t_x$,$t_y$) to a new position p'=(x',y') where
  - $x' = x + t_x$
  - $y' = y + t_y$

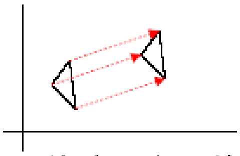



- The translation pair ($t_x$,ty) is called the *translation vector* or *shift vector*

$P = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ $P' = \begin{bmatrix} x_1' \\ x_2' \end{bmatrix}$ $P = \begin{bmatrix} t_x \\ t_y \end{bmatrix}$

Using matrix notation

$P' = P + T$

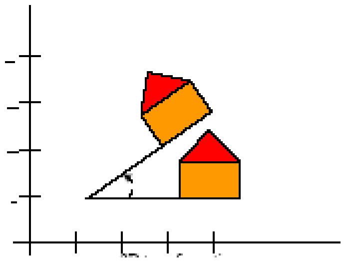


$\Delta' = \{p + t \mid p \in \Delta\}$

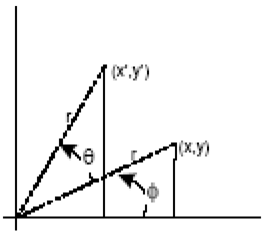
- This is a rigid transformation. Namely, objects are not deformed.
- All points of the objects are moved by the same distance and direction
- Enough to translate the vertices (end points)

# Rotation

- Rotation of an object is repositioning it along a circular path in the *xy* plane.
- A rotation is specified by a rotation angle $\theta$ and a pivot point (rotation point) *p*



- The angle is defined in a counter clockwise order around the point p (counter clockwise is positive angle and clockwise is negative angle)
- The rotation can also be viewed as rotation around an axis line perpendicular to the *xy* plane through the point *p*
- Assuming rotation around the origin (0,0)




$$x' = r\cos(\phi + \theta) = r\cos\phi\cos\theta - r\sin\phi\sin\theta$$
$$y' = r\sin(\phi + \theta) = r\cos\phi\sin\theta + r\sin\phi\cos\theta$$

$$x' = r\cos(\phi + \theta) = r\cos\phi\cos\theta - r\sin\phi\sin\theta$$
$$y' = r\sin(\phi + \theta) = r\cos\phi\sin\theta + r\sin\phi\cos\theta$$

Since

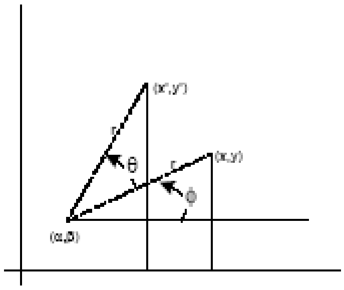
$x = r\cos\phi$ and $y = r\cos\phi$

we obtain

$$x' = x\cos\theta - y\sin\theta$$
$$y' = x\sin\theta + y\cos\theta$$

# Rotation around point $p$

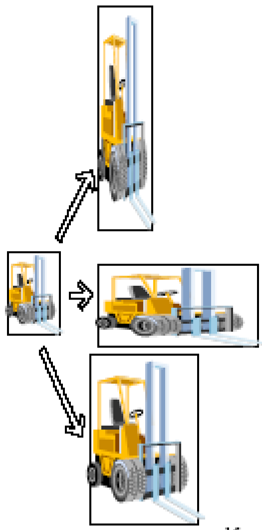
$$x = \alpha + r\cos\phi \quad \Rightarrow r\cos\phi = x\text{-}\alpha$$
$$y = \beta + r\sin\phi \quad \Rightarrow r\sin\phi = y\text{-}\beta$$
$$x' = \alpha + r\cos(\phi+\theta)$$
$$= \alpha + r\cos\phi\cos\theta - r\sin\phi\sin\theta$$
$$= \alpha + (x\text{-}\alpha)\cos\theta - (y\text{-}\beta)\phi\sin\theta$$
$$y' = \beta + (x\text{-}\alpha)\sin\theta + (y\text{-}\beta)\cos\theta$$

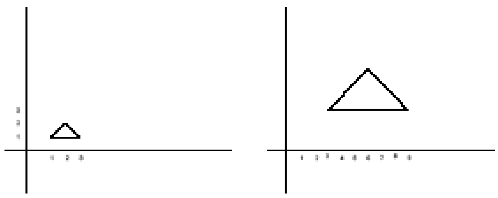



# Scaling

- A scaling transformation alters the size of an object.
- The operation is accomplished by multiplier each coordinate value (x,y) of a vertex v by scaling factors $s_x$, $s_y$:

$$x' = s_x x \quad \text{and} \quad y' = s_y y$$





Scaling using $s_x=s_y=3$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

or

$$P' = SP \quad \text{where } S \text{ is the scaling matrix}$$

- When $s_x$ ($s_y$) > 1 we enlarge the object
- When $s_x$ ($s_y$) < 1 we reduce the object
  - Uniform scaling $s_x=s_y$
  - Differential scaling $s_x \neq s_y$

- Transformed objects using the matrix $S$ are not only scaled but also change their position

- To avoid object repositioning we use a fixed point on the object (similar to a pivot point in rotation)

Assuming the the coordinates of the fixed point are $(x_f, y_f)$

$$x' = x_f + (x - x_f)s_x$$

$$y' = y_f + (y - y_f)s_y$$

rearranging

$$x' = xs_x + x_f(1 - s_x)$$

$$y' = ys_y + y_f(1 - s_y)$$

where $x_f(1 - s_x)$ and $Y_f(1 - s_y)$ are constants

and are added to all points on the object



## Homogenous Coordinates Motivation

- In many cases, e.g., animation, we need to combine a number of basic transformations such as a rotation followed by a translation

$$P' = RP + T$$

- R is a 2x2 matrix while T is a 2x1 matrix
- The rotation operation is multiplication operation while the translation operation is an addition

## Homogenous Computations

- How to speed up the process and to avoid long processing (recall that there can be thousands of vertices in an image)?

- Idea:
  - use a single matrix operation for all types of transformations (rotation, scaling, translation)

- Convert each transformation matrix into a 3x3 matrix
  - Translation matrix $P' = T(t_x, t_y)P$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

  - the inverse matrix is obtained by relacing the parameters $t_x$ and $t_y$ with $-t_x$ and $-t_y$

  - Rotation matrix $P' = R(\theta)P$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

  - Inverse rotation is obtained by replacing the parameters θ with - θ

# Composite Transformation

- What happens when we combine several transformations (concatenation/composition)
  - compute the product of two or more matrices to obtain the final transformation matrix

$$
\begin{aligned}
P' &= (M_1(M_2(M_3P))) \\
&= (M_1(M_2M_3))P \\
&= M_{1-3}P
\end{aligned}
$$

# Concatenation of Similar Transformations

- Translation

$$
\begin{aligned}
P' &= T(t_{x2}, t_{y2})\{T(t_{x1}, t_{y1})P\} \\
&= \{T(t_{x2}, t_{y2})T(t_{x1}, t_{y1})\}P
\end{aligned}
$$

$$
\begin{bmatrix} 1 & 0 & t_{x2} \\ 0 & 1 & t_{y2} \\ 0 & 0 & 1 \end{bmatrix}
\begin{bmatrix} 1 & 0 & t_{x1} \\ 0 & 1 & t_{y1} \\ 0 & 0 & 1 \end{bmatrix}
=
\begin{bmatrix} 1 & 0 & t_{x1}+t_{x2} \\ 0 & 1 & t_{y1}+t_{y2} \\ 0 & 0 & 1 \end{bmatrix}
$$

Hence,

$$T(t_{x2}, t_{y2})T(t_{x1}, t_{y1}) = T(t_{x1}+t_{x2}, t_{y1}+t_{y2})$$



- Scaling

$$
\begin{bmatrix} s_{x2} & 0 & 0 \\ 0 & s_{y2} & 0 \\ 0 & 0 & 1 \end{bmatrix}
\begin{bmatrix} s_{x1} & 0 & 0 \\ 0 & s_{y1} & 0 \\ 0 & 0 & 1 \end{bmatrix}
=
\begin{bmatrix} s_{x2}s_{x1} & 0 & 0 \\ 0 & s_{y2}s_{y1} & 0 \\ 0 & 0 & 1 \end{bmatrix}
$$

Hence,

$$S(s_{x2}, s_{y2})S(s_{x1}, s_{y1}) = S(s_{x1}s_{x2}, s_{y1}s_{y2})$$

- Rotation

$$
\begin{aligned}
P' &= R(\theta_2)\{R(\theta_1)P\} \\
&= \{R(\theta_2)R(\theta_1)\}P
\end{aligned}
$$

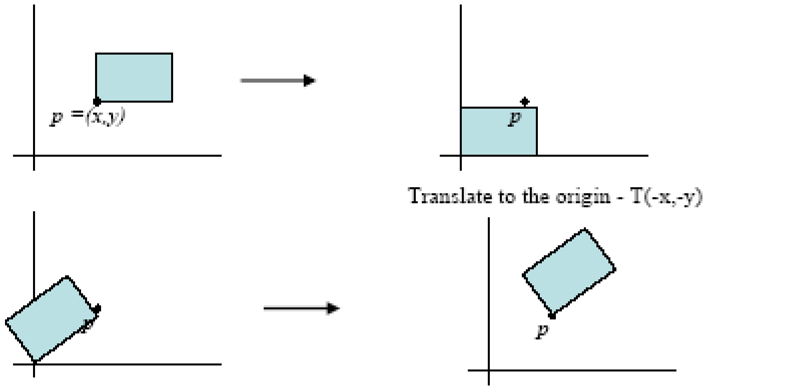
Hence,

$$R(\theta_2)R(\theta_1) = R(\theta_2 + \theta_1)$$

# Pivot Point Rotation

- So far all the transformations assumed that they are done around the origin (0,0)
- This can be generalized to any arbitrary point as follows
  - Translate the object to the origin such that the pivot point is at the origin
  - Transform the object around the origin (e.g., rotation)
  - Use inverse translation used in step 1 to move the object back to its original position

# Example Rotate by 45° around $p$




Translate to the origin - T(-x,-y)

Rotate by 45° - R(45)T(-x,-y)

Translate back to p - T(x,y) R(45)T(-x,-y)

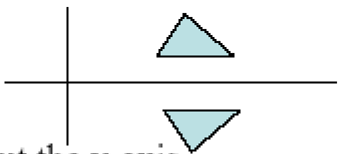
# Other Transformations

- Reflection
  - A transformation that produces a mirror image of an object
  - it is generated by rotating the object 180° around the axis of reflection

# Reflection about the axis (x-axis or y-axis)

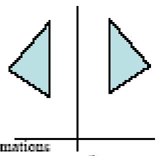
- a reflection about the x-axis

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



- a reflection about the y-axis

$$\begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

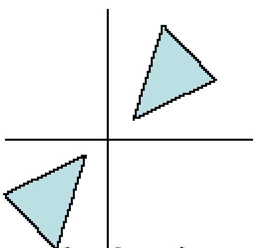




# Reflection about the origin (x-axis and the y-axis)

- Show a picture of reflection about the x-axis and y-axis

$$\begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



- This transformation can also be viewed as a rotation where $\theta$=180°

## Reflection about lines

- Line y=x

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- It can be viewed as
  - a rotation by -45° (CCW 45°)
  - Reflection about the x-axis
  - a rotation by 45° (CW 45°)

## Shear

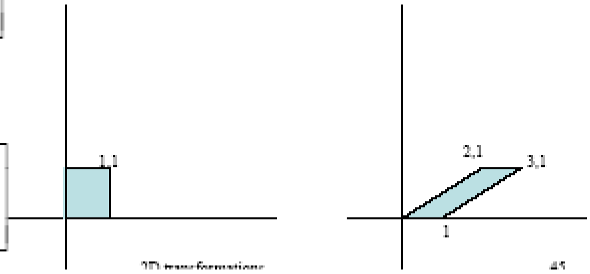
- Shear transformation distorts the shape of the object.
- The effect causes the object to be pushed to one side as if it was constructed of layers that slide on top of each other

- Shearing transformation along the x-axis

$$\begin{bmatrix} 1 & sh_x & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad x' = x + sh_x y \qquad y' = y$$

- example

$$\begin{bmatrix} 1 & 2 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

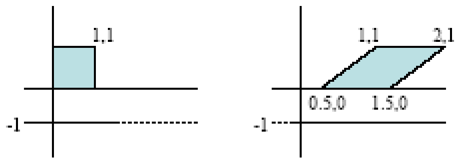



- Shearing transformation along the x-axis using a reference line

$$\begin{bmatrix} 1 & sh_x & -sh_x y_{ref} \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad x' = x + sh_x (y - y_{ref}) \qquad y' = y$$

- example using a shearing along the x-axis of ½ and a reference line y=-1

$$\begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{2} \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$




# 1.6 TWO DIMENSIONAL CLIPPING AND VIEWING

## The Viewing Pipeline

•**Window**

•A world-coordinate area selected for display. defines *what* is to be viewed

•**Viewport**

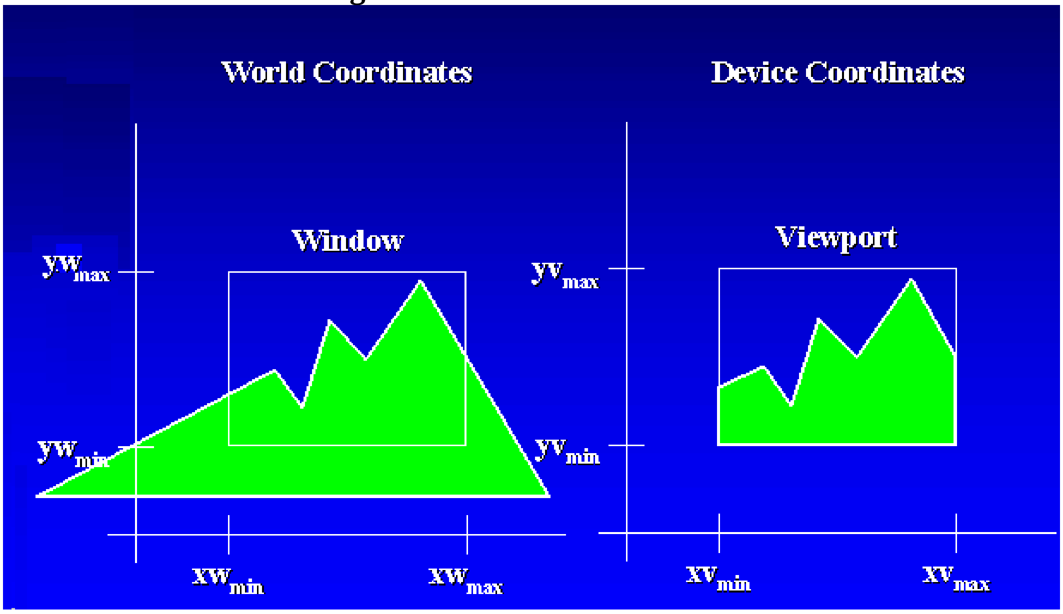
•An area on a display device to which a window is mapped. defines *where* it is to be displayed

**•Viewing transformation**

•The mapping of a part of a world-coordinate scene to device coordinates.

**•A window could be a rectangle to have any orientation.**

**Two-Dimensional Viewing**




**The Viewing Pipeline**

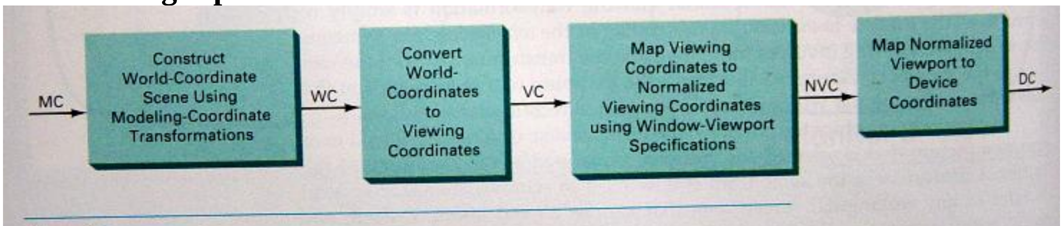


*Figure 6-2*
The two-dimensional viewing-transformation pipeline.

## Viewing Effects

**•Zooming effects**

Successively mapping different-sized windows on a fixed-sized viewports.

**•Panning effects**

Moving a fixed-sized window across the various objects in a scene.

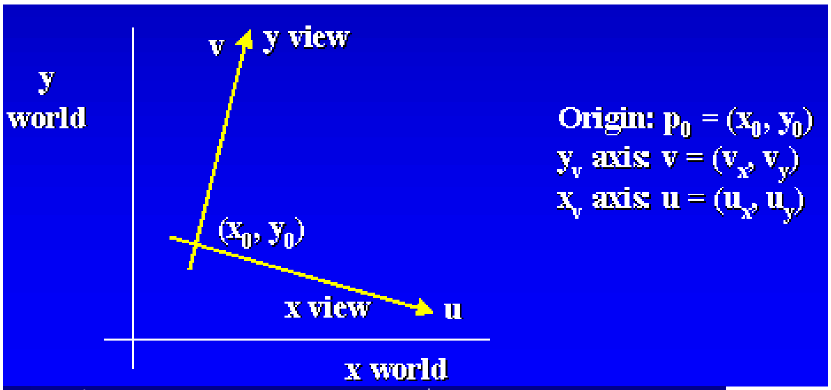
**•Device independent**

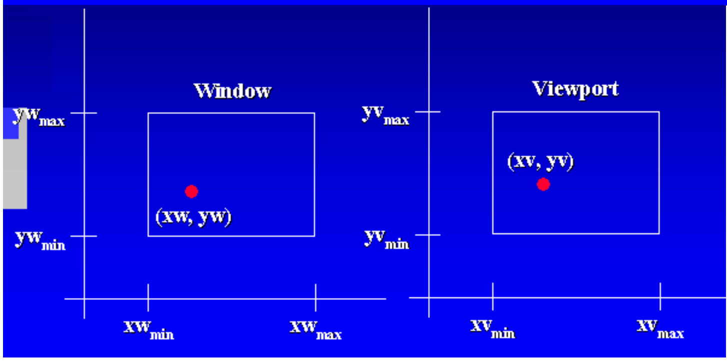
Viewports are typically defined within the unit square (normalized coordinates)

**Viewing Coordinate Reference Frame** **The reference frame for specifying the world-coordinate window.**

**•Viewing-coordinate origin: $P_0 = (x_0, y_0)$**

**•View up vector V: Define the viewing $y_v$ direction**




**Window-to-Viewport Coordinate Transformation**

$$\frac{xv - xv_{min}}{xv_{max} - xv_{min}} = \frac{xw - xw_{min}}{xw_{max} - xw_{min}} \qquad \frac{yv - yv_{min}}{yv_{max} - yv_{min}} = \frac{yw - yw_{min}}{yw_{max} - yw_{min}}$$

$$xv = xv_{min} + (xw - xw_{min})sx$$

$$yv = yv_{min} + (yw - yw_{min})sy$$

$$sx = \frac{xv_{max} - xv_{min}}{xw_{max} - xw_{min}}$$

$$sy = \frac{yv_{max} - yv_{min}}{yw_{max} - yw_{min}}$$

Clipping Operations

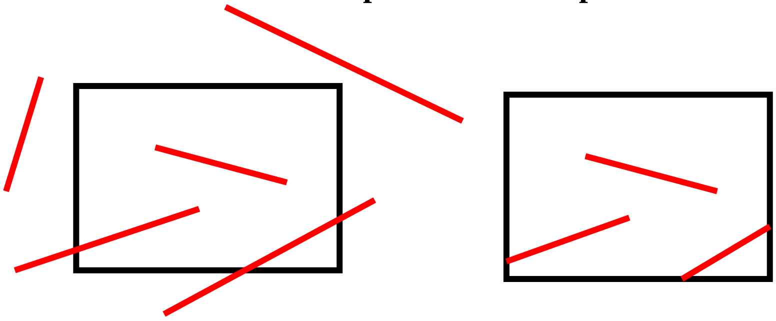
- **Clipping**
  - Identify those portions of a picture that are either inside or outside of a specified region of space.
- **Clip window**
  - The region against which an object is to be clipped.
  - The shape of clip window
- Applications of clipping
- World-coordinate clipping

## Clipping Operations

- Viewport clipping
  - It can reduce calculations by allowing concatenation of viewing and geometric transformation matrices.
- Types of clipping
  - Point clipping
  - Line clipping
  - Area (Polygon) clipping
  - Curve clipping
  - Text clipping
- Point clipping (Rectangular clip window)

**Line Clipping**

- **Possible relationships between line positions and a standard rectangular clipping region**



Before clipping after clipping

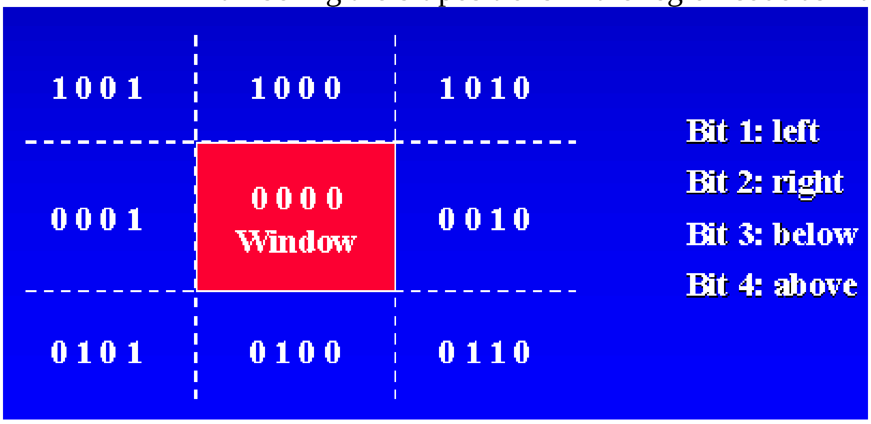
- Possible relationships
  - Completely inside the clipping window
  - Completely outside the window
  - Partially inside the window
- Parametric representation of a line

$$x = x1 + u(x2 - x1) \qquad y = y1 + u(y2 - y1)$$

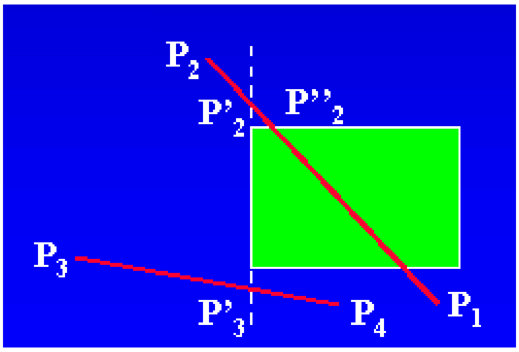
- The value of u for an intersection with a rectangle boundary edge
  - Outside the range 0 to 1
  - Within the range from 0 to 1

## Cohen-Sutherland Line Clipping

- **Region code**
  - A four-digit binary code assigned to every line endpoint in a picture.
  - Numbering the bit positions in the region code as 1 through 4 from right to left.




- Bit values in the region code
  - Determined by comparing endpoint coordinates to the clip boundaries
    - A value of 1 in any bit position: The point is in that relative position.
- Determined by the following steps:
  - Calculate differences between endpoint coordinates and clipping boundaries.
  - Use the resultant sign bit of each difference calculation to set the corresponding bit value.
- The possible relationships:
  - Completely contained within the window
    - 0000 for both endpoints.
  - Completely outside the window
    - Logical and the region codes of both endpoints, its result is not 0000.
  - Partially




**Liang-Barsky Line Clipping**
**Rewrite the line parametric equation as follows:**

- **Rewrite the line parametric equation as follows:**

$$x = x_1 + u\Delta x$$
$$y = y_1 + u\Delta y \qquad 0 \le u \le 1$$

- **Point-clipping conditions:**

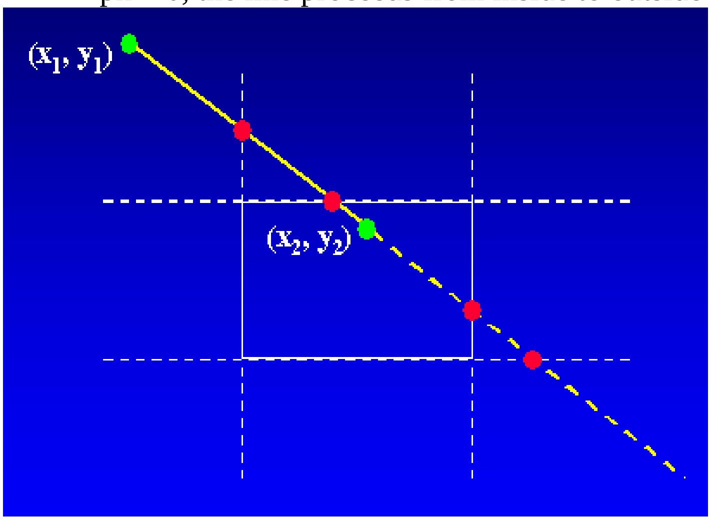
$$xw_{min} \le x_1 + u\Delta x \le xw_{max}$$
$$yw_{min} \le y_1 + u\Delta y \le yw_{max}$$

- **Each of these four inequalities can be expressed as:** $up_k \le q_k \qquad k = 1,2,3,4$

- **Parameters p and q are defined as**

$$p_1 = -\Delta x \qquad q_1 = x_1 - xw_{min}$$
$$p_2 = \Delta x \qquad q_2 = xw_{max} - x_1$$
$$p_3 = -\Delta y \qquad q_3 = y_1 - yw_{min}$$
$$p_4 = \Delta y \qquad q_4 = yw_{max} - y_1$$

- pk = 0, parallel to one of the clipping boundary
  - qk < 0, outside the boundary
  - qk >= 0, inside the parallel clipping boundary
- pk < 0, the line proceeds from outside to the inside
- pk > 0, the line proceeds from inside to outside



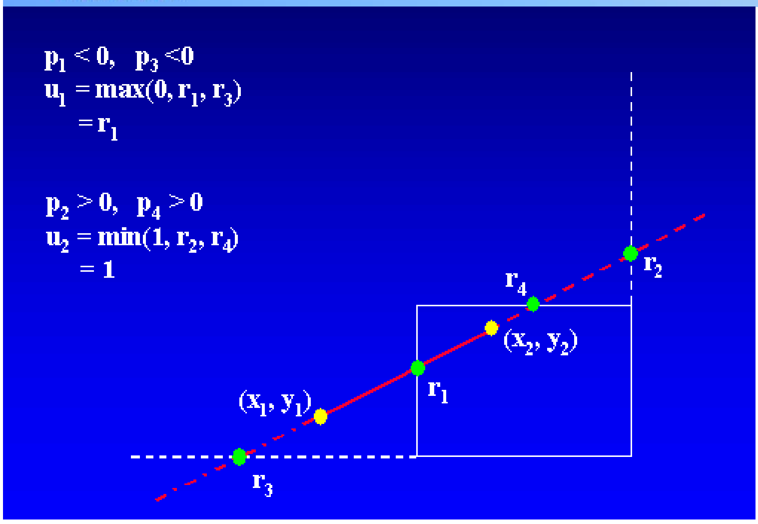
- **Parameters $u_1$ and $u_2$ that define that part of the line lies within the clip rectangle**
- **The value of $u_1$ : From outside to inside ($p_k < 0$)**

  $$u_1 = \max(0, r_k's) \qquad r_k = \frac{q_k}{p_k}$$

- **The value of $u_2$ : From inside to outside ($p_k > 0$)**

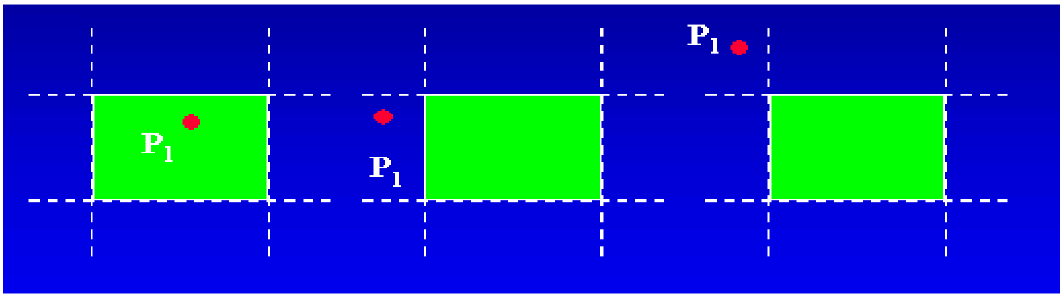
  $$u_2 = \min(1, r_k's) \qquad r_k = \frac{q_k}{p_k}$$

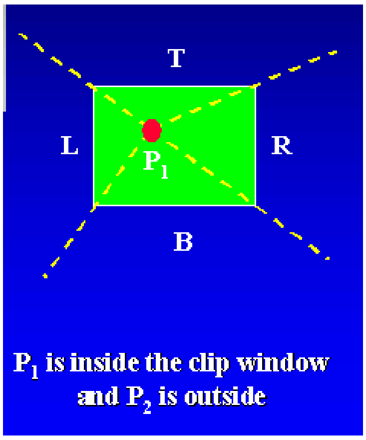
- **If $u_1 > u_2$, the line is completely outside the clip window.**

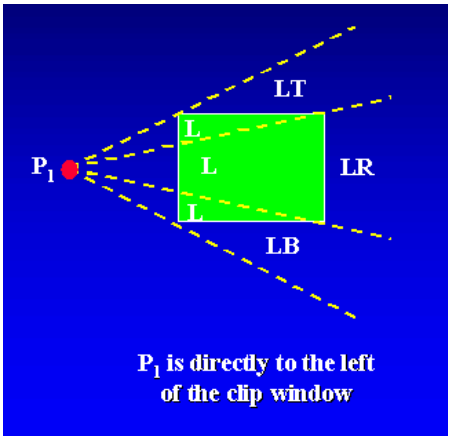


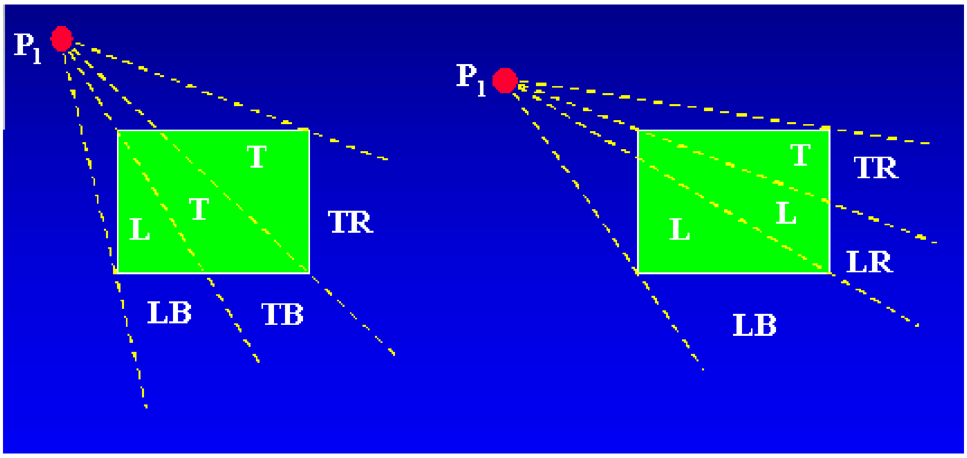

## Nicholl-Lee-Nicholl Line Clipping

- Compared to C-S and L-B algorithms
  - NLN algorithm performs fewer comparisons and divisions.
  - NLN can only be applied to 2D clipping.
- The NLN algorithm
  - Clip a line with endpoints P1 and P2
  - First determine the position of P1 for the nine possible regions.
    - Only three regions need be considered
    - The other regions using a symmetry transformation
  - Next determine the position of  P2 relative to P1.



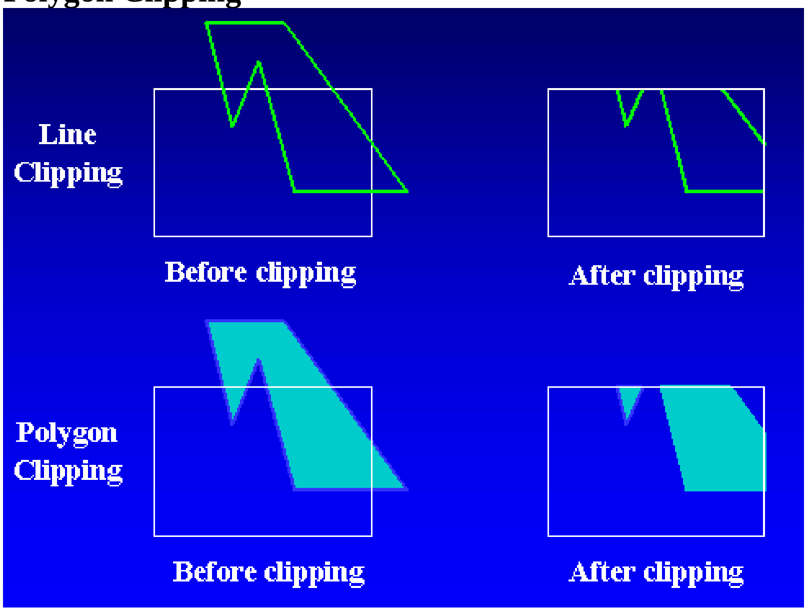


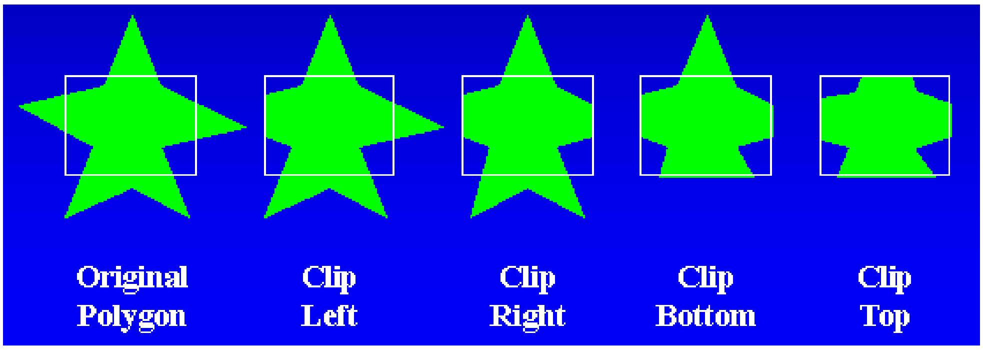

- To determine the region in which P2 is located
  - Compare the slope of the line to the slopes of the boundaries of the clip region.
  - Example: P1 is left of the clipping rectangle, P2 is in region LT.

**Polygon Clipping**

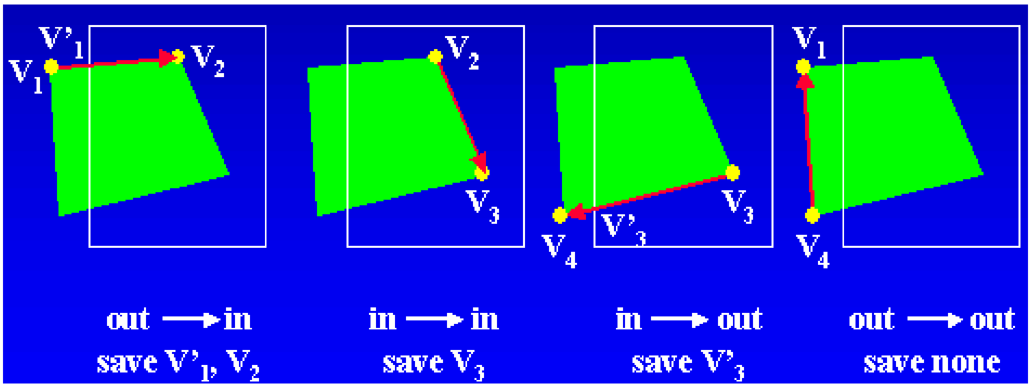



**Sutherland-Hodgeman Polygon Clipping**

- Processing the polygon boundary as a whole against each window edge
  - Processing all polygon vertices against each clip rectangle boundary in turn



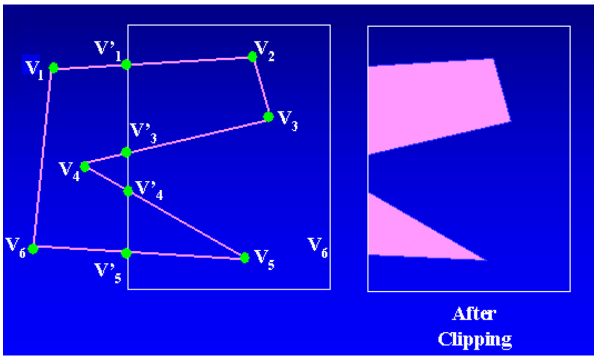

- Pass each pair of adjacent polygon vertices to a window boundary clipper
  - There are four cases:




- Intermediate output vertex list
  - Once all vertices have been processed for one clip window boundary, it is generated.
  - The output list of vertices is clipped against the next window boundary.
  - It can be eliminated by a pipeline of clipping routine.
- Convex polygons are correctly clipped.
- If the clipped polygon is concave
  - Split the concave polygon

## Weiler-Atherton Polygon Clipping

- Developed as a method for identifying visible surfaces
  - It can be applied with arbitrary polygon-clipping region.
  - Not always proceeding around polygon edges
  - Sometimes follows the window boundaries
- For clockwise processing of polygon vertices
  - For an outside-to-inside pair of vertices, follow the polygon boundary.
  - For an inside-to-outside pair of vertices, follow the window boundary in clockwise direction.




- Curve clipping
  - Use bounding rectangle to test for overlap with a rectangular clip window.
- Text clipping
  - All-or-none string-clipping
  - All-or-none character-clipping
  - Clip the components of individual characters

## 1.7 Input Techniques

### INPUT OF GRAPHICAL DATA

Two possible ways to see input devices:

- as a physical device – keyboard, mouse, trackball, etc.
- as a logical device – from a programmer perspective – with specified functionality, in graphics more complex
- the separation of physical and logical levels enable us to make programs more flexible, independent from the actual physical device

**Physical input devices:**

- pointing device – allows to indicate position & send signals/interrupts to the computer – relative/absolute positioning
- keyboard device – almost physical keyboard – returns character codes to a program

**Absolute positioning:**

- data tablets
- light pen
- joystick – variable-sensitivity device & haptic device
- spaceball – up-down, left-right, front-back & 3 independent twists

**Logical Input Devices**

Some APIs (PHIGS, GKS, Direct xx) supports 6 classes of logical input devices – OpenGL does not take this approach

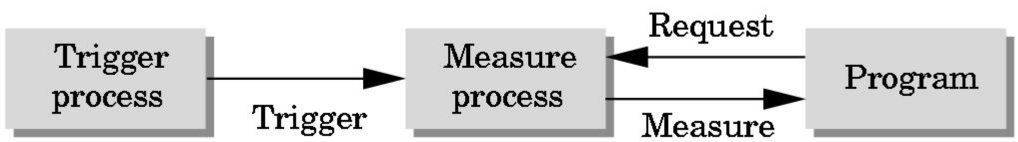
- Two older APIs (GKS, PHIGS) defined six types of logical input
  - **Locator**: return a position
  - **Pick**: return ID of an object
  - **Keyboard**: return strings of characters
  - **Stroke**: return array of positions
  - **Valuator**: return floating point number
  - **Choice**: return one of n items

**Input Modes**

- Input devices contain a *trigger* which can be used to send a signal to the operating system
  - Button on mouse
  - Pressing or releasing a key
- When triggered, input devices return information (their *measure*) to the system
  - Mouse returns position information
  - Keyboard returns ASCII code

**Request Mode**

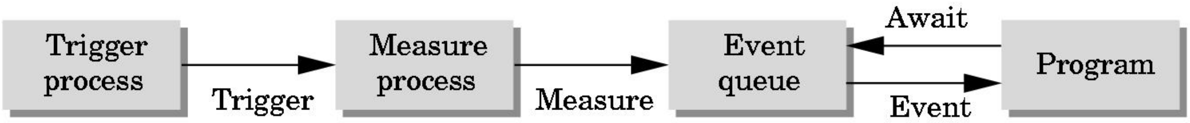
- Input provided to program only when user triggers the device
- Typical of keyboard input
  - Can erase (backspace), edit, correct until enter (return) key (the trigger) is depressed




**Event Mode**

- Most systems have more than one input device, each of which can be triggered at an arbitrary time by a user

- Each trigger generates an *event* whose measure is put in an *event queue* which can be examined by the user program




**Event Types**

- Window: resize, expose, iconify
- Mouse: click one or more buttons
- Motion: move mouse
- Keyboard: press or release a key
- Idle: nonevent

  Define what should be done if no other event is in queue

# UNIT II THREE-DIMENSIONAL CONCEPTS

**UNIT II THREE-DIMENSIONAL CONCEPTS 9**

Three-Dimensional object representations – Three-Dimensional geometric andmodeling transformations – Three-Dimensional viewing – Hidden surface elimination– Color models – Animation.

## THREE DIMENSIONAL DISPLAY METHODS

To obtain display of a three-dimensional scene that has been modeled in world coordinates. we must first set up a coordinate reference for the "camera". This coordinate reference defines the position and orientation for the plane of the carnera film which is the plane we want to us to display a view of the objects in the scene. Object descriptions are then transferred to the camera reference coordinates and projected onto the selected display plane. We can then display the objects in wireframe (outline) form, or we **can** apply lighting surface rendering techniques to shade the visible surfaces.

- **PARALLEL PROJECTION**

In a parallel projection, parallel lines in the world-coordinate scene projected into parallel lines on the two-dimensional display plane.

- **Perspective Projection**

Another method for generating a view of a three-dimensional scene is to project points to the display plane along converging paths. This causes objects farther from the viewing position to be displayed smaller than objects of the same size that are nearer to the viewing position.In a perspective projection, parallel lines in a scene that are not parallel to the display plane are projected into converging lines

- DEPTH CUEING

A simple method for indicating depth with wireframe displays is to vary the intensity of objects according to their distance from the viewing position. The viewing position are displayed with the highest intensities, and lines farther away are displayed with decreasing intensities.

### Visible Line and Surface Identification

We can also clarify depth relation ships in a wireframe display by identifying visible lines in some way. The simplest method is to highlight the visible lines or to display them in a different color. Another technique, commonly used for engineering drawings, is to display the nonvisible lines as dashed lines. Another approach is to simply remove the nonvisible lines

### Surface Rendering

Added realism is attained in displays by setting the surface intensity of objects according to the lighting conditions in the scene and according to assigned surface characteristics. Lighting specifications include the intensity and positions of light sources and the general background illumination required for a scene. Surface properties of objects include degree of transparency and how rough or smooth the surfaces are to be. Procedures can then be applied to generate the correct illumination and shadow regions for the scene.

- Exploded and Cutaway View

  Exploded and cutaway views of such objects can then be used to show the internal structure and relationship of the object Parts

  Three-Dimensional and Stereoscopic View

- Three-dimensional views can be obtained by reflecting a raster image from a vibrating flexible mirror. The vibrations of the mirror are synchronized with the display of the scene on the CRT. As the mirror vibrates, the focal length varies so that each point in the scene is projected to a position corresponding to its depth.
- Stereoscopic devices present two views of a scene: one for the left eye and the other for the right eye.

**THREE DIMENSIONAL OBJECT REPRESENTATIONS**

Representation schemes for solid objects are often divided into two broad categories

- **Boundary representations (B-reps)** describe a three-dimensional object as a set of surfaces that separate the object interior from the environment.Typical examples of boundary representations are polygon facets and spline patches.

- **Space-partitioning representations** are used to describe interior properties, by partitioning the spatial region containing an object into a set of small, nonoverlapping, contiguous solids (usually **cubes).**

**POLYGON SURFACES**

The most commonly used boundary representation for a three-dimensional graphics object is a set of surface polygons that enclose the object interior. Many graphics systems store all object descriptions as sets of surface polygons. This simplifies and speeds up the surface rendering and display of objects, since all surfaces are described with linear equations. For this reason, polygon descriptions are often referred to as "standard graphics objects."

## Polygon Tables

We specify a polygon surface with a set of vertex coordinates and associated attribute parameters. As information for each polygon is input, the data are placed into tables that are to be used in the subsequent' processing, display, and manipulation of the objects in a scene.

Polygon data tables can be organized into two groups:

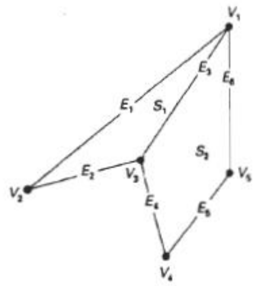
**geometric tables - attribute tables.**

Geometric data tables contain vertex coordinates and parameters to identify the spatial orientation of the polygon surfaces.

Attribute information for an object includes parameters specifying the degree of transparency of the object and its surface reflectivity and texture characteristics.

A convenient organization for storing geometric data is to create three lists: **a vertex table, an edge table, and a polygon table.**

Coordinate values for each vertex in the object are stored in the vertex table. The edge table contains pointers back into the vertex table to identify the vertices for each polygon edge. And thepolygon table contains pointers back into the edge table to identify the edges for each polygon.




| VERTEX TABLE |
|---|
| $V_1$: $x_1, y_1, z_1$ |
| $V_2$: $x_2, y_2, z_2$ |
| $V_3$: $x_3, y_3, z_3$ |
| $V_4$: $x_4, y_4, z_4$ |
| $V_5$: $x_5, y_5, z_5$ |

| EDGE TABLE |
|---|
| $E_1$: $V_1, V_2$ |
| $E_2$: $V_2, V_3$ |
| $E_3$: $V_3, V_1$ |
| $E_4$: $V_3, V_4$ |
| $E_5$: $V_4, V_5$ |
| $E_6$: $V_5, V_1$ |

| POLYGON-SURFACE TABLE |
|---|
| $S_1$: $E_1, E_2, E_3$ |
| $S_2$: $E_3, E_4, E_5, E_6$ |

## Plane Equations

To produce a display of a three-dimensional object, we must process the input data representation for the object through several procedures.

These processing steps include transformation of the modeling and world-coordinate descriptions to viewing coordinates, then to device coordinates; identification of visible surfaces; and the application of surface-rendering procedures.

The equation for 'I plane surface can be expressed In the form

Ax + By + Cz + D =0

where (r, y, z ) is any point on the plane, and the coefficients A, B, C, and D are constants describing the spatial objects of the plane

we select three successive polygon vertices (x1, y1,z1), (x2, y2,z2), (x3, y3,z3),

and solve thc following set of simultaneous linear plane equation5 for the ratios AID, B/D,and ClD

$(A/D)x_k - (B/D)y_k + (C/D)z_k = -1,$ where k =1,2,3…

The solution for this set of equations can be obtained in determinant form, using Cramer's rule, as

$$A = \begin{vmatrix} 1 & y_1 & z_1 \\ 1 & y_2 & z_2 \\ 1 & y_3 & z_3 \end{vmatrix} \qquad B = \begin{vmatrix} x_1 & 1 & z_1 \\ x_2 & 1 & z_2 \\ x_3 & 1 & z_3 \end{vmatrix}$$

$$C = \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix} \qquad D = -\begin{vmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{vmatrix} \qquad (10\text{-}3)$$

Expanding thc determinants, we can write the calculations for the plane coefficients in the form

$$A = y_1(z_2 - z_3) + y_2(z_3 - z_1) + y_3(z_1 - z_2)$$

$$B = z_1(x_2 - x_3) + z_2(x_3 - x_1) + z_3(x_1 - x_2)$$

$$C = x_1(y_2 - y_3) + x_2(y_3 - y_1) + x_3(y_1 - y_2)$$

$$D = -x_1(y_2z_3 - y_3z_2) - x_2(y_3z_1 - y_1z_3) - x_3(y_1z_2 - y_2z_1)$$

As vertex values and other information are entered into the polygon data structure,values tor ***A, B***, *C*. and D are computed for each polygon and stored with the other polygon data.

Plane equations are used also to identify the position of spatial points relativeto the plane surfaces of an object. For any point (x, y, **z)** not on a plane withparameters ***A, B,*** C, ***D,*** we have Ax + By+ Cz+ D != 0

We **can** identify the point as either inside or outside the plane surface accordingto the sign (negative or positive) of ***Ax*** + ***By*** + Cz + D:

if $Ax + By + Cz + D < 0$, the point $(x, y, z)$ is inside the surface

if $Ax + By + Cz + D > 0$, the point $(x, y, z)$ is outside the surface

**CURVED LINES AND SURFACES**

Curve and surface equations can be expressed in either a parametric or anonparametric form.

**QUADRIC SURFACES**

A frequently used class of objects is the quadric surfaces, which are described
with second-degree equations (quadratics). They include spheres, ellipsoids, tori,
paraboloids, and hyperboloids. Quadric surfaces, particularly spheres and ellipsoids, are common elements of graphics scenes.

## Sphere

In Cartesian coordinates, a spherical surface with radius r centered on the coordi-nate origin is defined as the set of points (x, y, **z)** that satisfy the equation

$$x^2 + y^2 + z^2 = r^2$$

We can also describe the spherical surface in parametric form, using latitude and longitude angles (Fig. 10-8):

$$\begin{aligned} x &= r\cos\phi\cos\theta, & -\pi/2 \le \phi \le \pi/2 \\ y &= r\cos\phi\sin\theta, & -\pi \le \theta \le \pi \qquad (10\text{-}8) \\ z &= r\sin\phi \end{aligned}$$

**Torus**

**A** torus is a doughnut-shaped object, as shown in Fig. It can be generated by rotating a circle or other conic about a specified axis. The Cartesian representation for points over the surface of a torus can be written in the form

$$\left[r - \sqrt{\left(\frac{x}{r_x}\right)^2 + \left(\frac{y}{r_y}\right)^2}\,\right]^2 + \left(\frac{z}{r_z}\right)^2 = 1$$

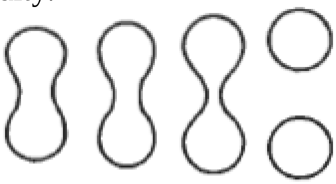
where **r** is any given offset value. Parametric representations for a torus are similar to those for an ellipse, except that angle **d** extends over 360". Using latitude and longitude angles , we can describe the toms surface as the set of points that satisfy

$$x = r_x(r + \cos\phi)\cos\theta, \qquad -\pi \le \phi \le \pi$$
$$y = r_y(r + \cos\phi)\sin\theta, \qquad -\pi \le \theta \le \pi$$
$$z = r_z \sin\phi$$

**BLOBBY OBJECTS**

Some objects do not maintain a fixed shape, but change their surface characteristics in certain motions or when in proximity to other opts.

Examples in this class of objects include molecular structures, water droplets and other liquid effects, melting objects, and muscle shapes in the human body. **These** objects can bedescribed as exhibiting "blobbiness" and are often simply referred to as blobby objects, since their shapes show a certain degree of fluidity.

Eg: 

Surface function is then defined as

$$f(x, y, z) = \sum_k b_k e^{-a_k r_k^2} - T = 0$$

where $r_k^2 = \sqrt{x_k^2 + y_k^2 + z_k^2}$, parameter $T$ is some specified threshold

And a& b – to adjust the amount of blobbiness of an object.

**SPLINE REPRESENTATIONS**

In drafting terminology, a spline is a flexible strip **used** to produce a smoothcurve through a designated **set** of points. Several small weights are distributed along the length of the strip to hold it in position on the drafting table as the curve is drawn.

**INTERPOLATION AND APPROXIMATION SPLINES**

We specify a spline curver by giving a set of coordinate positions, called control points, which indicates the general shape of the curve. These control points are then fitted with piece wise continuous parametric poly nomial functions in one of two ways.

- When polynomial sections are fitted so that the curve passes through
each control point, the resulting curve is said to interpolate the set of control points.
- On the other hand, when the polynomials are fitted to the general control-point path without necessarily passing through any control point, the resulting curve is said to approximate the set of control points

## Spline Specifications

There are three equivalent methods for specifying a particular spline representation:

(1) We can state the set of boundary conditions that are imposed on the spline;

*Figure 10-19*
A set of six control points interpolated with piecewise continuous polynomial sections.

*Figure 10-20*
A set of six control points approximated with piecewise continuous polynomial sections

(2) we can state the matrix that characterizes the spline; or

(3) we can state the set of blending functions (or basis functions) that determine how specified geometric constraints on the curve are combined to calculate positions along the curve path

**we** have the following parametric cubic polynomial representation for the **x** coordinate along the path of a spline section:

$$x(u) = a_x u^3 + b_x u^2 + c_x u + d_x, \qquad 0 \le u \le 1$$

Boundary conditions for this curve might be set, for example, on the endpoint coordinates x(0) and x(l) and on the parametric first derivatives at the endpoints x'(0) and x' ( 1 ) . These four boundary conditions are sufficient to determine the values of the four coefficients ak, bk, ck, and dk.

$$x(u) = [u^3\ u^2\ u\ 1]\begin{bmatrix} a_x \\ b_x \\ c_x \\ d_x \end{bmatrix}$$

$$= \mathbf{U} \cdot \mathbf{C}$$ Matrix product

Where U – row matrix of powers of parameter u

C – coefficient column matrix

1. To obtain a polynomial representation for coordinate **x** in terms ***of*** the geometric constraint parameters

$$x(u) = \sum_{k=0}^{3} g_k \cdot BF_k(u)$$

where ***gk*** are the constraint parameters, such as the control-point coordinates and slope of the curve at the control points, and ***BFk(u)*** are the polynomial blending functions.

**BEZIER CURVES AND SURFACES**

This spline approximation method was developed by the French engineer Pierre Mzier for use in the design of Renault automobile bodies. **Bezier** splines have a number of properties that make them highly useful and convenient for curve and surface design.

## Bezier Curves

In general, a Bezier curve section can be fitted to any number of control points.

The number of control points to be approximated and their relative position determine the degree of the BCzier polynomial. As with the interpolation splines, a bezier curve can be specified with boundary conditions, with a characterizing matrix, or with blending functions.

For general Bezier curves, the blending-function specification is the most convenient.

Suppose we are given ***n*** + 1 control-point positions: **pk** = (xk, ***yk,*** zk), with ***k*** varying from **0** to n. These coordinate points can **be** blended to produce the following position vector ***P(u),*** which describes the path of an approximating Bezier polynomial function between **p1,** and $p_n$.

$$\mathbf{P}(u) = \sum_{k=0}^{n} \mathbf{p}_k BEZ_{k,n}(u), \qquad 0 \le u \le 1$$

- The Bezier blending functions ***$BEZ_{k,n}$,(u)*** are the ***Bemstein polynomials:***

$$BEZ_{k,n}(u) = C(n,k)u^k(1-u)^{n-k}$$

where the ***C(n, k)*** are the binomial coefficients:

$$C(n,k) = \frac{n!}{k!(n-k)!}$$

we can define Bezier blending functions with the recursive calculation

$$BEZ_{k,n}(u) = (1-u)\,BEZ_{k,n-1}(u) + u\,BEZ_{k-1,n-1}(u), \qquad n > k \ge 1$$

with $BEZ_{k,k} = u^k$, and $BEZ_{0,k} = (1 - u)^k$

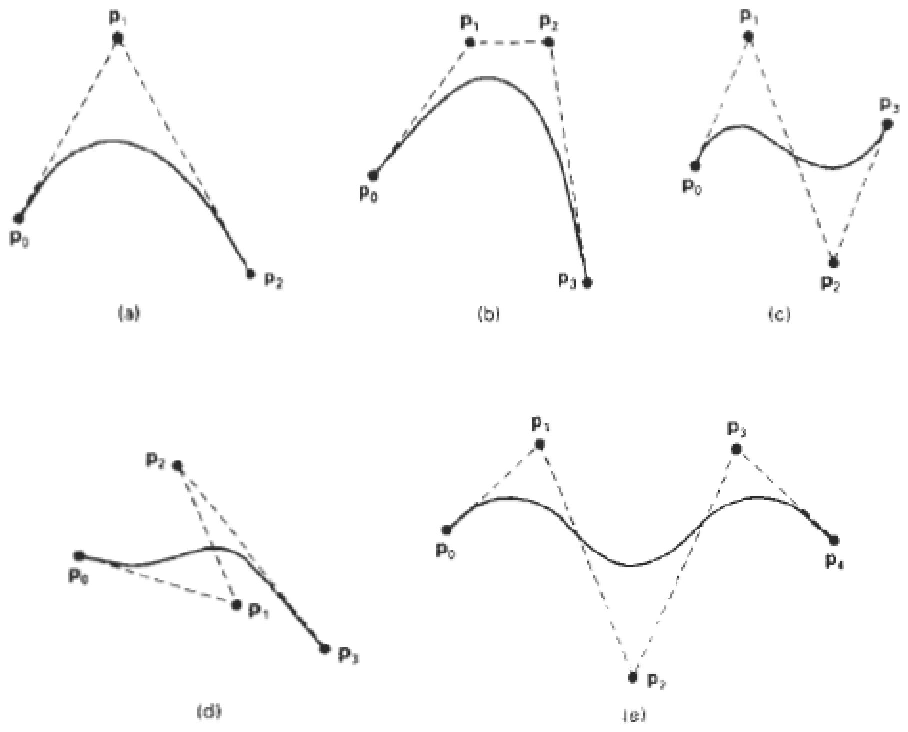
vector equation represents a set **of** three parametric equations for the individual curve coordinates

$$x(u) = \sum_{k=0}^{n} x_k\, BEZ_{k,n}(u)$$

$$y(u) = \sum_{k=0}^{n} y_k\, BEZ_{k,n}(u)$$

$$z(u) = \sum_{k=0}^{n} z_k\, BEZ_{k,n}(u)$$

a Bezier curve is a polynomial of degree one less than the number of control points used: Three points generate a parabola, four points a cubiccurve, and so forth. For Example




## Properties of Bezier Curves

1. A very useful property of a Bezier curve is that it always passes through the first and last control points. That is, the boundary conditions at the two ends of the curve are

$$\mathbf{P}(0) = \mathbf{p}_0$$

$$\mathbf{P}(1) = \mathbf{p}_n$$

2. Values of the parametric first derivatives of a Bezier curve at the endpoints can be calculated from control point coordinates as

$$\mathbf{P}'(0) = -n\mathbf{p}_0 + n\mathbf{p}_1$$

$$\mathbf{P}'(1) = -n\mathbf{p}_{n-1} + n\mathbf{p}_n$$

Thus, the slope at the beginning of the curve is along the line joining the first two control points, and the slope at the end of the curve is along the line joining the last two endpoints.

3. Similarly, the parametric second derivatives o f a Bezier curve at the endpoints are calculated as

$$\mathbf{P}''(0) = n(n-1)[(\mathbf{p}_2 - \mathbf{p}_1) - (\mathbf{p}_1 - \mathbf{p}_0)]$$

$$\mathbf{P}''(1) = n(n-1)[(\mathbf{p}_{n-2} - \mathbf{p}_{n-1}) - (\mathbf{p}_{n-1} - \mathbf{p}_n)]$$

Another important property of any Bezier curve is that it lies within the convex hull (convex polygon boundary) of the control points. This follows from the properties of Bezier blending functions:

They are all positive and is always one.

$$\sum_{k=0}^{n} BEZ_{k,n}(u) = 1$$

where the Bezier matrix is

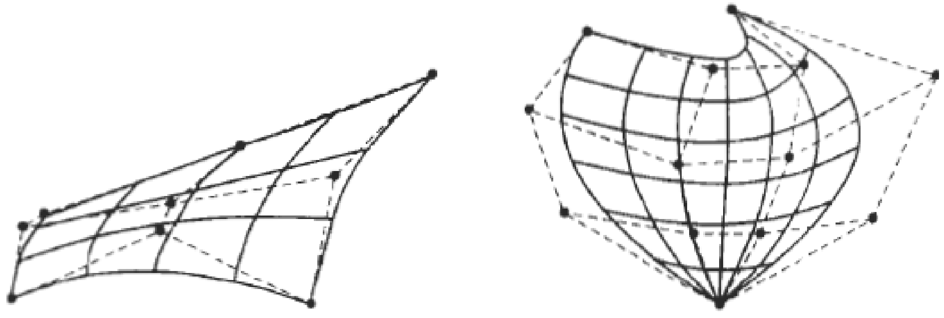
$$\mathbf{M}_{Bez} = \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

## Bezier Surtaces

Two sets of orthogonal Bezier curves can be used to design an object surface by specifying by an input mesh of control points. The parametric vector function for the Bezier surface is formed as the Cartesian product of Bezier blending functions:

$$\mathbf{P}(u,v)=\sum_{j=0}^{m}\sum_{k=0}^{n}\mathbf{p}_{j,k}BEZ_{j,m}(v)BEZ_{k,n}(u)$$

with P**j,k** specifying the location of the ***(m*** **+ 1)** by ***(n*** **+ I )** control points.



(a) (b)

The control points are connected by dashed lines, and the solid lines show curves of constant ***u*** and constant v. Each curve of constant ***u*** is plotted by varying v over the interval from 0to **1,** with ***u*** fixed at one of the values in this unit interval. Curves of constant v are plotted similarly

## B-SPLINE CURVES AND SURFACES

B-splines have two advantages over Bezier splines:

**(1)** the degree of a B-spline polynomial can be set independently of the number of control points (with certain limitations),

*(2)* B-splines allow local control over the shape of a spline curve or surface

## B-Spline Curves

We can write a general expression for the calculation of coordinate positions along a B-spline curve in a blending-function formulation as

$$\mathbf{P}(u)=\sum_{k=0}^{n}\mathbf{p}_k B_{k,d}(u), \qquad u_{min}\le u\le u_{max}, \qquad 2\le d\le n+1$$

where the **pk** are an input set of **n** + 1 control points.

Blending functions for B-spline curves are defined by the Cox-deBoor recursion formulas:

$$B_{k,1}(u)=\begin{cases}1, & \text{if } u_k\le u<u_{k+1}\\ 0, & \text{otherwise}\end{cases}$$

$$B_{k,d}(u)=\frac{u-u_k}{u_{k+d-1}-u_k}B_{k,d-1}(u)+\frac{u_{k+d}-u}{u_{k+d}-u_{k+1}}B_{k+1,d-1}(u)$$

where each blendjng function is defined over d subintervals of the total range of ***u.***

The selected set of subinterval endpoints u, is referred to as a **knot** vector.

B-spline curves have the following **properties**.

- The polynomial curve has degree d - 1 and $C^{d-2}$ continuity over the rangeof ***u.***
- For n + 1 control points, the curve is described with n + 1 blending functions.
- Each blending function $B_{k,d}$, is defined over d subintervals of the total range of u, starting at knot value ***u1.***
- The range of parameter u 1s divided into ***n*** + d subintervals by the **n** + ***d*** +1 values specified in the knot vector.
- With knot values labeled as ***[u1,*** u2, . . . , ***un,],*** the resulting B-spline curve is defined only in the interval from knot value $u_{d-1}$ , up to knot value $u_{n+1}$.
- Each section of the spline curve (between two successive knot values) is influenced by d control points.
- Any one control point can affect the shape of at most d curve sections.

B-splines are tightly bound to the input positions. For any value of ***u*** in the interval from knot value $u_{d-1}$ to $u_{n+1}$ the sum over all basis functions is 1:

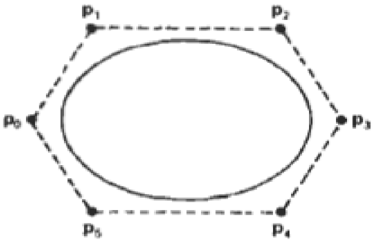
$$\sum_{k=0}^{n}B_{k,d}(u)=1$$

- **Uniform, Periodic B-Splines**

When the spacing between knot values is constant, the resulting curve is called a uniform B-spline.

- **Cubic, Periodic B-Splines**

Since cubic, periodic 8-splines are commonly used in graphics packages, we consider the fornlulation for this class of splines. Periodic splines are particularly useful for generating certain closed curves




A closed, periodic, piecewise, cubic B-spline constructed with cyclic specification of the six control points.

- **Open Uniform B-Splines**

This class of B-splines is a cross between uniform B-splines and nonuniform Bsplines. Sometimes it is treated as a special type of uniform 8-spline, and sometimes it is considered to be in the nonuniform B-splines classification.

For the open uniform B-splines, or simply open B-splines, the knot spacing is uniform except at the ends where knot values are repeated d times.

For any values of parameters ***d*** and n, we can generate an open uniform knot vector with integer values using the calculations

$$u_j = \begin{cases} 0, & \text{for } 0 \le j < d \\ j - d + 1, & \text{for } d \le j \le n \\ n - d + 2, & \text{for } j > n \end{cases}$$

for values of] ranging from 0 to ***n*** + ***d.*** With this assignment, the first d knots are assigned the value 0, and the last ***d*** knots have the value ***n*** - d + *2.*

- **Non Uniform B-Splines**

For this class of splines, we can specify any values and intervals for the knot vector. With nonuniform B-splines, we can choose multiple internal knot values and unequal spacing between the knot values.

## B-Spline Surfaces

We can obtain a vector point function over a B-spline surface using the Cartesian product ofB-spline blending functions in the form

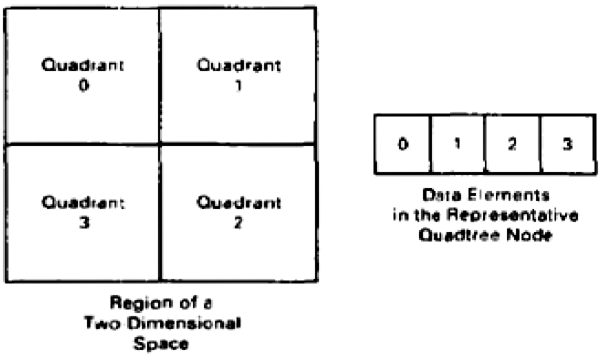
$$\mathbf{P}(u, v) = \sum_{k_1=0}^{n1} \sum_{k_2=0}^{n2} \mathbf{p}_{k_1,k_2} B_{k_1,d_1}(u) B_{k_2,d_2}(v)$$

where the vector values for $\mathbf{p}_{k_1,k_2}$ specify positions of the $(n_1 + 1)$ by $(n_2 + 1)$ control points.

## OCTREES

Hierarchical tree structures, called octrees, are used to represent solid objects in some graphics systems.The tree structure is organized so that each node corresponds to a region of three-dimensional space. This representation for solids takes advantage of spatial coherence to reduce storage requirements for three-dimensional objects. It also provides a convenient representation for storing information about object interiors.

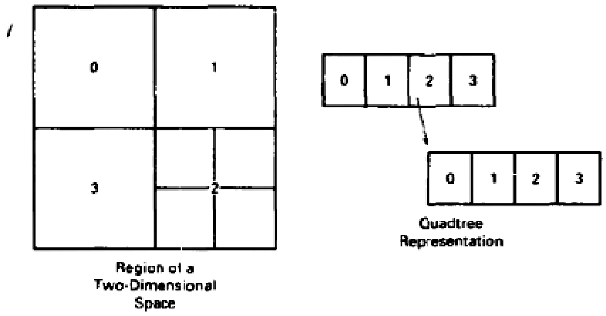
The octree encoding procedure for a three-dimensional space is an extensionof an encoding scheme for two-dimensional space, called quadtree encoding.Quadtrees are generated by successively dividing a two-dimensional region (usually a square) into quadrants. Each node in the quadtree has four data elements,one for each of the quadrants in the region




If all pixels within a quadrant have the same color (a homogeneous quadrant), the corresponding data element in the node stores that color.

In addition, a flag is set in the data element to indicate that the quadrant is homogeneous. Suppose all pixels in quadrant are found to be red. The color code for red is then placed in data element 2 of the node. Otherwise, the quadrant is said to be heterogeneous, and that quadrant is itself divided into quadrants

An octree encoding scheme divides regions of three-dimensional space (usually cubes) into octants and stores eight data elements in each node of the tree




- Individual elements of a three-dimensional space are called **volume elements,** or **voxels.**
- When all voxels in an octant are of the same type this type value is stored in the corresponding data element of the node.
- Empty regions of space are represented by voxel type "void."
- Any heterogeneous octant is subdivided into octants, and the corresponding data element in the node points to the next node in the octree.

**BSP TREES**

This representation scheme is similar to **octree** encoding, except **we** now divide space into two partitions instead of eight at each step.

With a binary space-partitioning **(BSP)** tree, we subdivide a scene into two sections at each step with aplane that can be at any position and orientation.

In an octree encoding, the scene is subdivided at each step with three mutually perpendicular planes aligned with the Cartesian coordinate planes.

## 2.2 Three-Dimensional Geometric and Modeling Transformations

Methods for geometric transformations and object modeling in three dimensions are extended from two-dimensional methods by including considerations for the ***z*** coordinate.

We now translate an object by specifying a three-dimensional translation vector, which determines how much the object is tobe moved in each of the three coordinate directions. Similarly, we scale an object with three coordinate scaling factors.

**TRANSLATION**

In a three-dimensional homogeneous coordinate representation, a point is translated
(Fig. **17-1)** from position P = ***(x,* y, *z)*** to position P' = ***(x',* y',** z') with the matrix
Operation

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \qquad P' = T \cdot P$$

Parameters $t_x$, $t_y$, and $t_z$, specifying translation distances for the coordinate directions $x$, $y$, and $z$, are assigned any real values. The matrix representation in Eq. 11-1 is equivalent to the three equations

$$x' = x + t_x, \qquad y' = y + t_y, \qquad z' = z + t_z \tag{11-3}$$

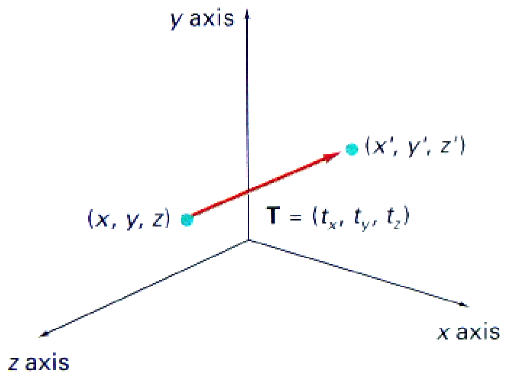



*Figure 11-1*
Translating a point with translation vector $\mathbf{T} = (t_x, t_y, t_z)$.

An object is translated in three dimensions by transforming each of the defining points of the object.

ROTATION

To generate a rotation transformation for an object, we must designate an axis of rotation (about which the object is to be rotated) and the amount of angular rotation.

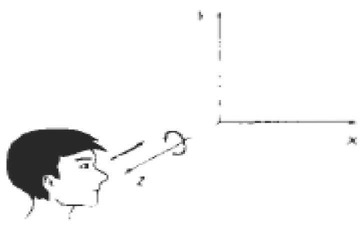
Unlike two-dimensional applications, where all transformations are carried out in the **xy** plane, a three-dimensional rotation can be specified around any line in space.

The easiest rotation axes to handle are those that are parallel to the coordinate axes. Also, we can use combinations of coordinate axis rotations (along with appropriate translations) to specify any general rotation.

By convention, positive rotation angles produce counterclockwise rotations about a coordinate axis, if we are looking along the positive half of the axis toward the coordinate origin.

## Coordinate-Axes Rotations

The two-dimensional z-axis rotation equations are easily extended to three dimensions:



$$\begin{aligned} x' &= x\cos\theta - y\sin\theta \\ y' &= x\sin\theta + y\cos\theta \\ z' &= z \end{aligned} \tag{11-4}$$

Parameter $\theta$ specifies the rotation angle. In homogeneous coordinate form, the three-dimensional z-axis rotation equations are expressed as

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{11-5}$$

which we can write more compactly as

$$\mathbf{P}' = \mathbf{R}_z(\theta) \cdot \mathbf{P}$$

or

$$\mathbf{P}' = \mathbf{R}_x(\theta) \cdot \mathbf{P} \qquad (11\text{-}10)$$

Rotation of an object around the $x$ axis is demonstrated in Fig. 11.6.

Cyclically permuting coordinates in Eqs. 11-8 give us the transformation equations for a **$y$-axis rotation**:

$$\begin{aligned} z' &= z\cos\theta - x\sin\theta \\ x' &= z\sin\theta + x\cos\theta \\ y' &= y \end{aligned} \qquad (11\text{-}11)$$

The matrix representation for $y$-axis rotation is

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \qquad (11\text{-}12)$$

or

$$\mathbf{P}' = \mathbf{R}_y(\theta) \cdot \mathbf{P} \qquad (11\text{-}13)$$

## General Three-Dimensional Rotations

**A** rotation matrix for any axis that does not coincide with a coordinate axis can be set up as a composite transformation involving combinations of translations and the coordinate-axes rotations. .

Any coordinate position **P** on the object in this figure is transformed with the sequence shown as

$$\mathbf{P}' = \mathbf{T}^{-1} \cdot \mathbf{R}_x(\theta) \cdot \mathbf{T} \cdot \mathbf{P}$$

where the composite matrix for the transformation is

$$\mathbf{R}(\theta) = \mathbf{T}^{-1} \cdot \mathbf{R}_x(\theta) \cdot \mathbf{T}$$

Given the specifications f or the rotation axis and the rotation angle, we can accomplish the required rotation in five s t e p

1 Translate the object so that the rotation axis pass= through the coordinate origin.

**2.** Rotate the object so that the axis of rotation coincides with one of the coordinate axes.

3. Perform the specified rotation about that coordinate axis.

4. Apply inverse rotations to bring the rotation axis back to original orientation.

5. Apply the inverse translation to bring the rotation axis back to its original position

A rotation axis can be defined with coordinate positions, as in fig or with one coordinate point and direction angles between the rotation axis and two of the coordinate axes. We will assume that the rotation axis is defined by two points and that the direction of rotation axes is to be counter clockwise when looking along the axis from $P_1$ to $P_2$

axis vector is then defined by the two points as

$$\begin{aligned} \mathbf{V} &= \mathbf{P}_2 - \mathbf{P}_1 \\ &= (x_2 - x_1, y_2 - y_1, z_2 - z_1) \end{aligned} \qquad (11\text{-}14)$$

A unit vector $\mathbf{u}$ is then defined along the rotation axis as

$$\mathbf{u} = \frac{\mathbf{V}}{|\mathbf{V}|} = (a, b, c) \qquad (11\text{-}15)$$

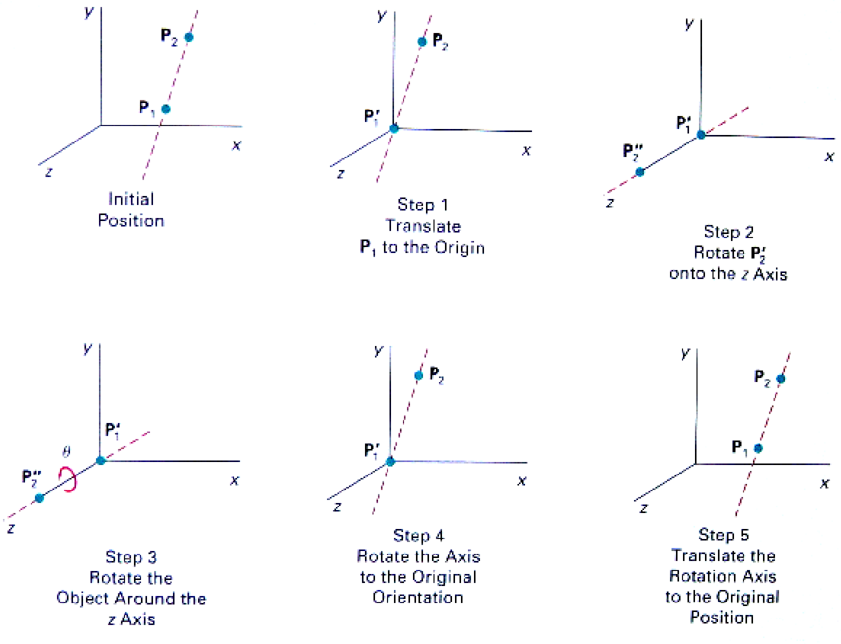



*Figure 11-9*
Five transformation steps for obtaining a composite matrix for rotation about an arbitrary axis, with the rotation axis projected onto the z axis.

## SCALING

The matrix expression tor the scaling transformation of a position P = **(x,** y, *z)* relative to the coordinate origin can be written as

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \qquad (11\text{-}42)$$

$$\mathbf{P}' = \mathbf{S} \cdot \mathbf{P} \qquad (11\text{-}43)$$

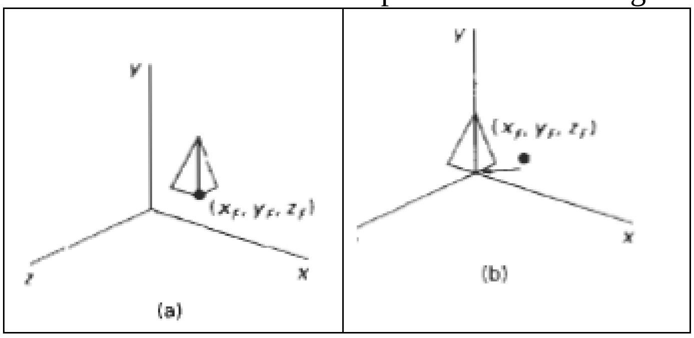
where scaling parameters $s_x$, $s_y$, and $s_z$, are assigned any positive values. Explicit expressions for the coordinate transformations for scaling relative to the origin are

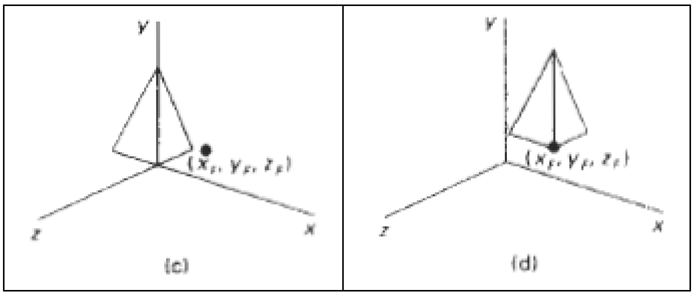
$$x' = x \cdot s_x, \qquad y' = y \cdot s_y, \qquad z' = z \cdot s_z$$

Scaling an object with transformation changes the size of the object and repositions the object relative to the coordinate origin. Also, if the transformation parameters are not all equal, relative dimensions in the object are changed:

Scaling with respect to a selected fixed position **(x,** y, z,*)* can be represented with the following transformation sequence:

1. Translate the fixed point to the origin.2. Scale the object relative to the coordinate origin using Eq. **11-42.**3. Translate the fixed point back to its original position.






This sequence of transformations is demonstrated in Fig. 11-18. The matrix representation for an arbitrary fixed-point scaling can then be expressed as the concatenation of these translate-scale-translate transformations as

$$\mathbf{T}(x_f, y_f, z_f) \cdot \mathbf{S}(s_x, s_y, s_z) \cdot \mathbf{T}(-x_f, -y_f, -z_f) = \begin{bmatrix} s_x & 0 & 0 & (1-s_x)x_f \\ 0 & s_y & 0 & (1-s_y)y_f \\ 0 & 0 & s_z & (1-s_z)z_f \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (11\text{-}45)$$

**Reflections**

**A** three-dimensional reflection can be performed relative to a selected ***reflection axis*** or with respect to a selected ***reception plane.*** In general, three-dimensional reflection matrices are set up similarly to those for two dimensions.

Reflections relative to a given axis are equivalent to 180 rotations about that axis. Reflections with respect to a plane are equivalent to 160' rotations in four-dimensional space.

When the reflection plane is a coordinate plane (either **xy,** ***xz, or*** yz), we can think of the transformation as a conversion between Left-handed and right-handed systems.

An example of a reflection that converts coordinate specifications from aright-handed system to a left-handed system (or vice versa)

This transformation changes the sign of the z coordinates, leaving the ***x*** and y-coordinate values unchanged. The matrix representation for this reflection of points relative to the **xy** plane is

$$RF_z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

## 2.3 THREE-DIMENSIONAL VIEWING

For three-dimensional applications, First of all, we can view an object from any spatial position: from the front, from above, or from the back. ***Or*** we could generate a view of what we would **see** if we were standing in the middle of a group of objects or inside a single object, such as a building. Additionally, three-dimensional descriptions of objects must be projected onto the flat viewing surface of the output device.

## VIEWING COORDINATES

Generating a view of an object in three dimensions is similar to photographing the object. We can walk around and take its picture from any angle, at various distances, and with varying camera orientations. Whatever appears in the viewfinder is projected onto the flat film surface. The type and size of the camera lens determines which parts of the scene appear in the final picture.

These ideas are incorporated into three dimensional graphics packages so that views of a scene can be generated, given the spatial position, orientation, and aperture size of the "camera".

**Specifying the View Plane**

1. We choose a particular view for a scene by first establishing the viewing-coordinate **system,** also called the view reference coordinate **system**. **A** view plane, or projection plane, is then set up perpendicular to the viewing ***z,*** axis.World-coordinate positions in the scene are transformed to viewing coordinates, then viewing coordinates are projected onto the view plane.To establish the viewing-coordinate reference frame, we first pick a world coordinate position called the view reference point.

## PROJECTIONS

Once world-coordinate descriptions of the objects in a scene are converted to viewing coordinates, we can project the three-dimensional objects onto the triodimensional view plane.
There are two basic projection methods .
In a parallel projection, coordinate positions are transformed to the view plane along parallel lines
For a perspective projection object positions are transformed to the view plane a!ong lines that converge to a point called the projection reference point (or center of projection).
The projected view of the object is determined by the interjection ofthe projection lines with the view z. plane.

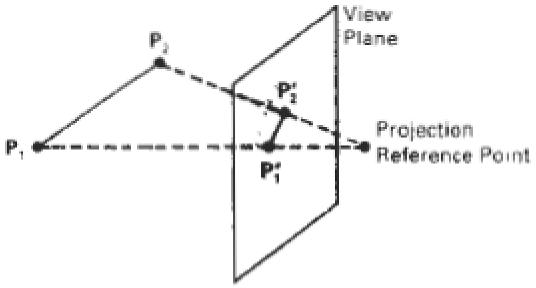



Figure 12-15
Perspective projection of an object to the view plane.

## Parallel Projections

We can specify a parallel projection with a projection vector that defines the directionfor the projection lines. When the projection is perpendicular to the view plane, we have an orthographic parallel projection. Otherwise, we have a oblique parallel projection.
Orthographic projections are most often used to produce the front, side, and top views of an object
**Perspective projection**

- The center of projection is located at a finite point in three space.
- A distant line is displayed smaller than a nearer line of the same length.

- In three-dimensional homogeneous-coordinate representation

$$\begin{bmatrix} x_h \\ y_h \\ z_h \\ h \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & z_{vp}/d_p & -z_{vp}(z_{prp}/d_p) \\ 0 & 0 & 1/d_p & -z_{prp}/d_p \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

When a three-dimensional obpct is projected onto a view plane using per- hree-Dimensional Viewing spective transformation equations, any set of parallel lines in the object that are not parallel to the plane are projected into converging lines.
Parallel Lines that are parallel to the view plane will be projected as parallel lines.

## vanishing point:

- The point at which a set of projected parallel lines appears to converge is called a vanishing point.
- Each such set of projected parallel lines will have a separate vanishing point;
- And in general, a scene can have any number of vanishing points, depending on howmany sets of parallel lines there are in the scene.
- The vanishing point for any set of lines that are parallel to one of the principal axes of an object is referred to as a principal vanishing point.
- We control the number of principal vanishing points (one, two, or three) with the orientation of the projection plane, and perspective projections are accordingly classified as one-point, two-point, or three-point projections.

**THREE-DIMENSIONAL VIEWING FUNCTIONS**
Several procedures are usually provided in a three-dimensional graphics library to enable an application program to set the parameters for viewing transfonnations.

With parameters spenfied in world coordinates, elements of the matrix for transforming worldcoordinate descriptions to the viewing reference frame are calculated using the function

***evaluateViewOrientationMatrix3 (x0, y0,' Z0, xN, yN, zN, xv, yv, zV, error.,viewMatrix)***

1. This function creates the viewMatrix from input coordinates defining the viewingsystem, 2. Parameters xo, yo, and z0 specify the origin (view reference point) of the viewing system. 3. World-coordinate vector defines the normal to the view plane and the direction of the positive, Three-Dimensional Viewing **viewi**ng axis. 4. And world-coordinate vector (xV, yv, zv) gives the elements of the view-up vector. The projection of this vector perpendicular to (xN, yN, zN) estab lishes the direction for the positive y, axis of the viewing system. 6. An integer error code is generated in parameter error if input values are not specified correctlyFor example, an error will be generated if we set (XV, YV, ZV) parallel to (xN, YN, zN)

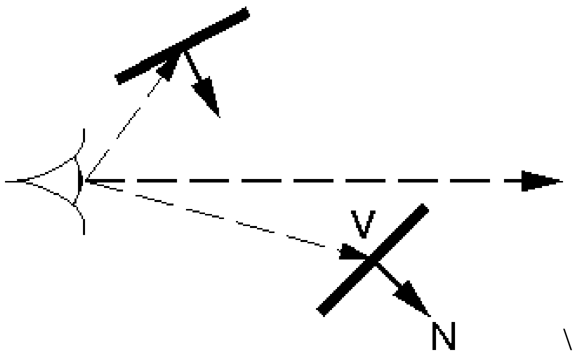
To specify a second viewing-coordinate system, we can redefine some or allof the coordinate parameters and invoke evaluatevieworientationMatrix3 with a new matrix designation. In this way, we can set up any number of world-to-viewingcoordinate matrix transformations.

## 2.4 HIDDEN SURFACE ELIMINATION

# Hidden Surface Removal

- When drawing lots of polygons, we want to draw only those ``visible" to viewer.
- There are a variety of algorithms with different strong points.
- Issues:
  - Online
  - Device independent
  - Fast
  - Memory requirements
  - Easy to implement in hardware

Backface CullingA simple way to perform hidden surface is to remove all ``backfacing" polygons. The observation is that if polygon normal is facing away from the viewer then it is ``backfacing." For solid objects, this means the polygon will not be seen by the viewer.




\

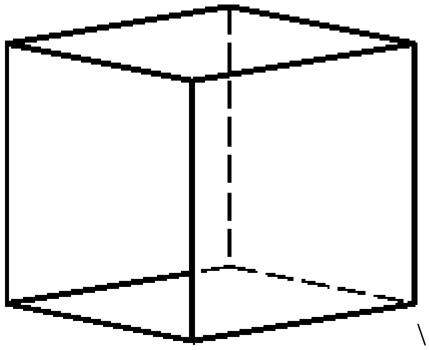
- Thus, if $N \cdot V > 0$, then cull polygon.
- Note that *V* is vector from eye to point on polygon

  You cannot use the view direction for this.

### Backface Culling

Not a complete solution

- If objects not convex, need to do more work.
- If polygons two sided (i.e., they do not enclose a volume) then we can't use it.

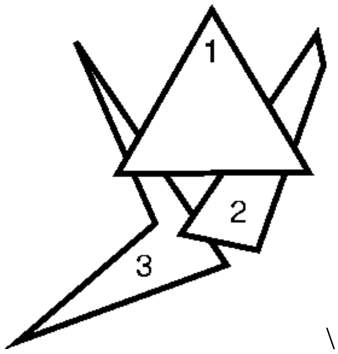


\

- A HUGE speed advantage if we can use it since the test is cheap and we expect at least half the polygons will be discarded.
- Usually performed in conjunction with a more complete hidden surface algorithm.
- Easy to integrate into hardware (and usually improves performance by a factor of 2).

## Painter's Algorithm

- Idea: Draw polygons as an oil painter might: The farthest one first.
  - Sort polygons on farthest *z*
  - Resolve ambiguities where *z*'s overlap
  - Scan convert from largest *z* to smallest *z*

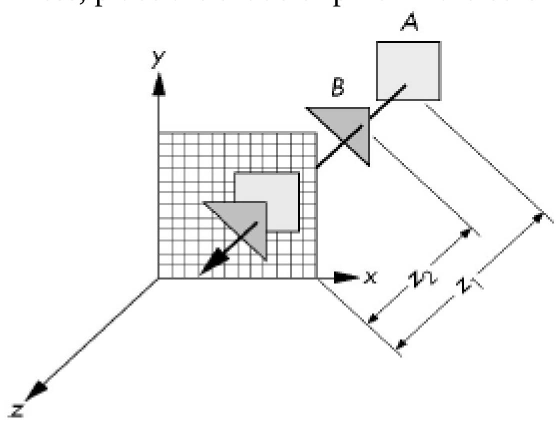



\

- Since closest drawn last, it will be on top (and therefore it will be seen).
- Need all polygons at once in order to sort.

## z-Buffer Algorithm

•The z or depth buffer–stores the depth of the closest object at each pixel
found so far As we render each polygon, compare the depth ofeach pixel to depth in z buffer
–If less, place the shade of pixel in the color buffer and update z buffer



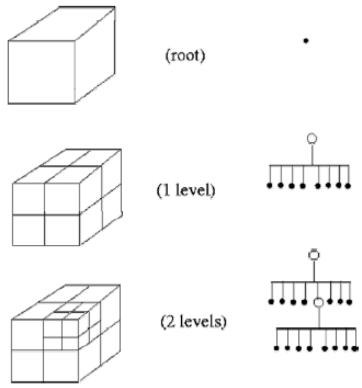

**Function** setpixel(int i, int j, rgb c, real z)
**If** z > z-buffer(i, j) **then**
z-buffer(i, j) = z
screen(i,j) = c

## Space Partitioning

•Avoid rendering an object when it’s unnecessary

–In many real-time applications, we want to eliminateas many objects as possible within the application.

Octree

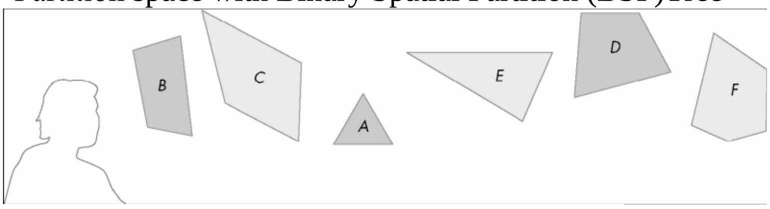


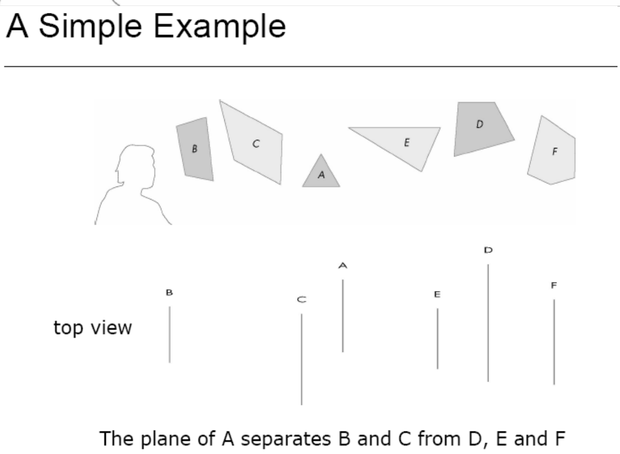

**Why do we use BSP trees?**

•Hidden surface removal

–A back-to-front painter’s algorithm

•Partition space with Binary Spatial Partition (BSP)Tree




## Binary Space Partitioning Tree

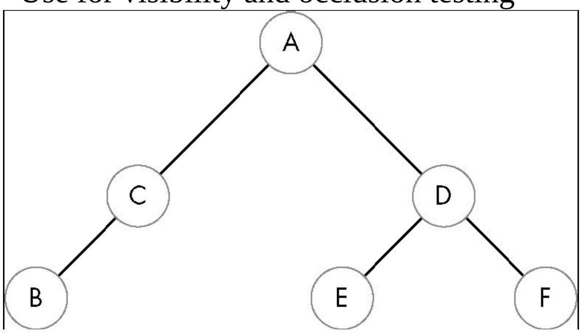
•Can continue recursively

–Plane of C separates B from A

–Plane of D separates E and F

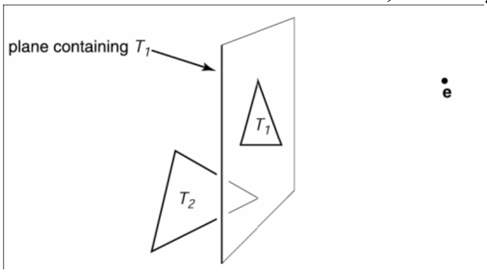
•Can put this information in a BSP tree

–Use for visibility and occlusion testing

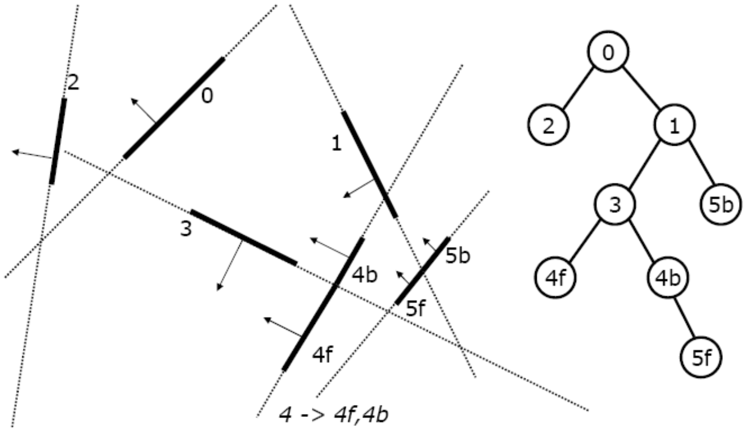



**Key Idea of BSP**

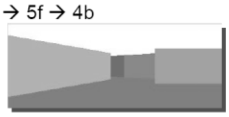
•Assume T2 not cross the plane of T1
•If e and T2 on the same side, T1 won't block
•If e and T2 on different sides, T2 may block

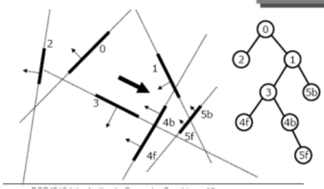



**Creating a BSP tree**




2 → 0 → 5b →1 → 4f → 3 → 5f → 4b





## 2.5 Color models

### 1. RGB COLOR MODEL

- Based on the tri stimulus theory of vision, our eyes perceive color through the stimulationof three visual pigments in the cones of the retina. These visual pigments have a peak sensitivity at wavelengths of about 630 run (red), 530 nm (green), and 450 nm (blue).
- By comparing intensities in a light source, we perceive the color of the light. This theory of vision is the basis for displaying color output on a video monitor using the three color primaries, red, green, and blue, referred to as the RGB color model.
- We can represent this model with the unit cube defined on R,G, and B axes,
  as shown in Fig. 15-11. The origin represents black, and the vertex with coordinates(1; 1,l) is white. Vertices of the cube on the axes represent the primary colors,and the remaining vertices represent the complementary color for each of the primary colors.
- As with the XYZ color system, the RGB color scheme is an additive model.
  Intensities of the primary colors are added to produce other colors. Each color point within the bounds of the cube can be represented as the triple (R, G, B), where values for R, G, and B are assigned in the range from 0 to 1. Thus, a color C, is expressed in RGB components as

$$C_\lambda = R\mathbf{R} + G\mathbf{G} + B\mathbf{B} \qquad (15\text{-}5)$$

- The magenta vertex is obtained by adding red and blue to produce the triple (1,0,l). and white at (1,l. 1) is thesum of the red, green, and blue vertices. Shadesof gray are represented along the main diagonal of the cube from the origin(black) to the white vertex.
- Each point along this diagonal has an equal contributionfrom each primary color, so that a gray shade halfway between black and white is represented as (0.5, 0.5, 0.5).

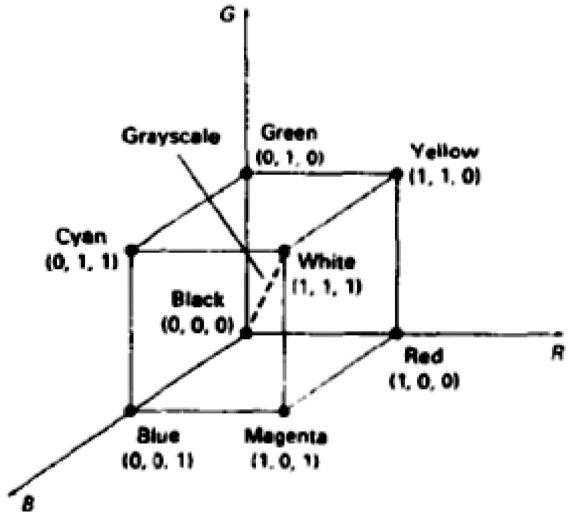



Figure 15-11
The RGB color model, defining colors with an additive process within the unit cube.

**2. YIQ COLOR MODEL**

. The National Television System Committee (NTSC) color model for forming the composite video signal is the YIQ model, which is based on concepts in the CIE XYZ model.

In the YI Q color model, parameter Y is the same as in the XYZ model. Luminance (brightness) information is contained in the Y parameter, while chromaticity information (hue and purity) is incorporated into the 1 and Q parameters.

A combination of red, green, and blue intensities is chosen for the Y parameter to yield the standard luminosity curve. Since Y contains the luminance information, black-and-white television monitors use only the Y signal.

The largest bandwidth in the NTSC video signal (about 4 MHz) is assigned to the Y information. Parameter I contains orange-cyan hue information that provides the flesh-tone shading, and occupies a bandwidth of approximately 1.5 MHz. Parameter Q carries green-magenta hue information in a bandwidth of about 0.6 MHz.

An RGB signal can be converted to a television signal using an NTSC encoder, which converts RGB values to YIQ values, then modulates and superimposes the I and Q information on the Y signal. The conversion from RGB values to YIQ values is accomplished with the transformation

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.144 \\ 0.596 & -0.275 & -0.321 \\ 0.212 & -0.528 & 0.311 \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix} \qquad (15\text{-}6)$$

This transformation is based on the NTSC standard RGB phosphor, whose chromaticity coordinates were given in the preceding section. The larger proportions of red and green assigned to parameter Y indicate the relative importance of these hues in determining brightness, compared to blue.

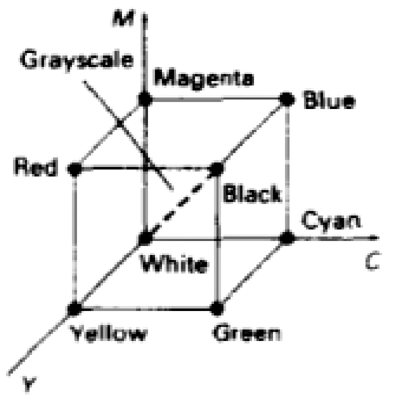
An NTSC video signal can be converted to an RGB signaul sing an NTSC decoder, which separates the video signal into the YlQ components,'then converts to RGB values. We convert from YIQ space to RGB space with the inverse matrix transformation from Eq. 15-6:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.000 & 0.956 & 0.620 \\ 1.000 & -0.272 & -0.647 \\ 1.000 & -1.108 & 1.705 \end{bmatrix} \cdot \begin{bmatrix} Y \\ I \\ Q \end{bmatrix} \qquad (15\text{-}7)$$

**3. CMY COLOR MODEL**

1 A color model defined with the primary colors cyan, magenta, and yellow (CMY) is useful for describing color output to hard-copy devices.

2. Unlike video monitors, which produce a color pattern by combining light from the screen phosphors, hard-copy devices such as plotters produce a color picture by coating a paper with color pigments. We see the colors by reflected light, a subtractive process.




In the CMY model, point (1, 1, 1) represents black, because all components of the incident light are subtracted. The origin represents white light.

Equal amounts of each of the primary colors produce grays, along the main diagonal of the cube. A combination of cyan and magenta ink produces blue light, becausethe red and green components of the incident light are absorbed. Other color combinations are obtained by a similar subtractive process

We can express the conversion from an RGB representation to a CMY representation with the matrix transformation

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

where the white is represented in the RGB system iis the unit column vector. Similarly,we convert from a CMY color representation to an RGB representationwith the matrix transformation where black is represented In the CMY system as the unit column vector.

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} C \\ M \\ Y \end{bmatrix}$$

**4. HSV COLOR MODEL**

lnstead of a set of color primaries, the HSV model uses color descriptions that have a more intuitive appeal to a user. To give a color specification, a user selects a spectral color and the amounts of white and black that are to be added to ohtain differens shades, tints, and tones. Color parameters in this model are hue ( H ) , saturation(S) and value(V)

The three dimensional representation of the HSV model is derived from the RGB cube. If we imagine viewing the cube along the diagonal from the white vertex to the origin(black), we see an outline of the cube that has the hexagon shape shown in fig. The boundary of the hexagon represents the various hues, and it is used as the top of the HSV hexcone. In the hexcone, saturation is measured along a horizontal; axis and value is along a vertical axis through the center of the hexcone.

Hue is represented as an angle about the vertical axis ranging from 0 degree at red through 360 degrees. Vertices of the hexagon are separated by $60^0$ intervals. Yellow is at $60^0$, green at $120^0$ and cyan opposite e red at $H = 180^0$ complementary colors are 1800 apart

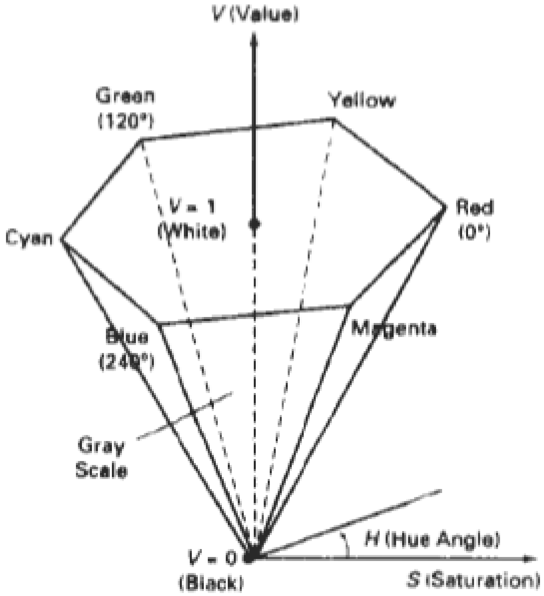



*Figure 15-16*
The HSV hexcone.

Saturation S varies from 0 to 1. It is represented in this model as the ratio of the purity of a selected hue to its maximum purity at S = 1. A selected hue is said to be one-quarter pure at the value S = 0.25. At S = 0, we have the gray scale.
Value V varies from 0 at the apex of the hexcone to I at the top. The apex represents black. At the top of the hexcone, colors have their maximum intensity. When V = 1 and S = 1, we have the "pure" hues. White is the point at V = 1 andS =o

## 2.6 ANIMATION

### DESIGN OF ANIMATION SEQUENCES

In general, an animation sequence is designed with the tollowing steps:

1. Storyboard layout
2. Object definitions
3. Key-frame specifications
4. Generation of in-between frames

This standard approach for animated cartoons is applied to other animation applications as well, although there are many special application that do not follow this sequence.
**The Storyboard layout** is an outline of the action. It defines the motion sequence as a
set of basic events that are to take place. Depending on the type of animation to be produced, the storyboard could consist of a set of rough sketches or it could be a list of the basic ideas for the motion.
**An object definition** is given for each participant in the action. Objects can bedefined interms of basic shapes, such as polygons or splines. In addition, the associatedmovements for each object are speeded along with the shape.
**A keyframe** is a detailed drawing of the scene at a certain time in the animation sequence.
Within each key frame, each object is positioned according to the time for that frame. Some key frames are chosen at extreme positions in the action; others are spaced so that the time interval between key frames is not to great. More key frames are specified for intricate motions than for simple, slowly varing motions.
**In-between frames** are the intermediate frames between the key frames.
The nurnber of in-betweens needed is determined by the media to be used to display theanimation. Film requires 24 frames per second, and graphics terminals are refreshedat the rate of 30 to 60 frames per second. Typically, time intervals for themotion are set up so that there arr from three to five in-betweens for each pair ofkey frames.

**GENERAL COMPUTER-ANIMATION FUNCTIONS**

Some steps in the development of an animation sequence are well-suited to computer solution. These include object manipulations and rendering, camera motions, and the generation of in-betweens. Animation packages, such as Wavefront,
One function available in animation packages is provided to store and manage the object database. Object shapes and associated parameters are stored and updated in the database. Other object functions

include those for motion generation and those for object rendering. Motions can be generated according to specified constraints using two-dimensional or three-dimensional transformations.

**RASTER ANIMATIONS**

On raster systems, we can generate real-time animation in limited applications using raster operations. Sequences of raster operations can be executed to produce real-time animation of either two-dimensional or three-dimensional objects, as long as we restrict the animation to motions in the projection plane. Then no viewing or visible- surface algorithms need be invoked.The animation is then accomplished by changing the color-table values so that the object is "on"at successively positions along the animation path as the preceding position is set-to the background intensity

**COMPUTER-ANIMATION LANGUAGES**

Design and control of animation sequences are handled with a set of animationroutines. A general-purpose language, such as C, Lisp, Pascal, or FORTRAN, is often used to program the animation functions, but several specialized animation languages have been developed. Animation functions include

- a graphics editor, a
- key-frame generator,
- an in-between generator,
- and standard graphics routines.

The graphics editor allows us to design and modify object shapes, using spline surfaces, constructive solid-geometry methods, or other representation schemes.

A typical task in an animation specification is scene description. This includes the positioning of objects and light sources, defining the photometric parameters (light-source intensities and surface-illumination properties), and setting the camera parameters (position, orientation, and lens characteristics).

Another standard function is action specification. This involves the layout of motion paths for the objects and camera. And we need the usual graphics routines: viewing and perspective transformations, geometric transformations to generate object movements as a function of accelerations or kinematics path specifications, visible-surface identification, and the surface-rendering operations.

**KEY FRAME SYSTEMS**

We generate each set of in-betweens from the specification of two (or more) keyframes. Motion paths can be given with a kinematic a s a set of spline curves, or the motions can be physicdly bnscd by specifying the for acting on the objects to be animated.For complex scenes, we can separate the frames into individual components or objects called cels celluloid transparencies)

**MORPHING**

Transformation of object shapes from one form to another is called morphing,which is a shortened form of metamorphosis. Morphing methods can he applied to any motion or transition involving a change in shape.Givcn two key frames for an object transformation, we first adjust the object specification in one of the frames so that the number of polygon edges (or the number of vertices) is the same for the two frames.

We can state rules for equalizing key frames in terms of either the number of edges or the number of vertices to be added to a key frame. Suppose we equalize the edge count, and parameters $L_k$ and $L_{k+1}$ denote the number of line segments in two consecutive frames. We then define

$$L_{max} = \max(L_k, L_{k+1}), \qquad L_{min} = \min(L_k, L_{k+1}) \qquad (16\text{-}1)$$

**MOTION SPECIFICATIONS** There are several ways in which the motions of objects can be specified in an animation system.

**Direct Motion Specification**The most straightforward method for defining a motion sequence is direct specification of the motion parameters. Here, we explicitly give the rotation angles and translation vectors. Then the geometric transformation matrices are applied to transform coordinate positions. Alternatively, we could use an approximating equation to specify certain kinds of motions.

$$y(x) = A\,|\sin(\omega x + \theta_0)|\,e^{-kx}$$

where A is the initial amplitude, w is the angular frequence, 0, is the phase angle,and k is the damping constant. These methods can be used for simple user-programmedanimation sequences.

**Goal-Directed Systems** At the opposite extreme, we can specify the motions that are to take place in general terms that abstractly describe the actions. These systems are referred to as goal directed because they determine specific motion parameters given the goalsof the animation.

**UNIT III MULTIMEDIA SYSTEMS DESIGN**

Multimedia basics − Multimedia applications − Multimedia system architecture −Evolving technologies for multimedia − Defining objects for multimedia systems −Multimedia data interface standards − Multimedia databases.

# 3.1 Multimedia Basics

Multimedia is a combination of text, graphic art, and sound, animation and video elements.

The IBM dictionary of computing describes multimedia as "comprehensive material, presented in a combination of text, graphics, video, animation and sound. Any system that is capable of presenting multimedia, is called a multimedia system".

A multimedia application accepts input from the user by means of a keyboard, voice or pointing device.

Multimedia applications involve using multimedia teclmology for business, education and entertainment.

Multimedia is now available on standard computer platforms. It is the best way to gain attention of users and is widely used in many fields as follows:

* Business - In any business enterprise, multimedia exists in the form of advertisements, presentations, video conferencing, voice mail, etc.

- Schools - Multimedia tools for learning are widely used these days. People of all ages learn easily and quickly when they are presented information with the visual treat.
- Home PCs equipped with CD-ROMs and game machines hooked up with TV screens have brought home entertainment to new levels. These multimedia titles viewed at home would probably be available on the multimedia highway soon.
- Public places - Interactive maps at public places like libraries, museums, airports and the stand-alone terminal
- Virtual Reality (VR) - This technology helps us feel a 'real life-like' experience. Games using virtual reality effect is very popular

**Multimedia Elements**

High-impact multimedia applications, such as presentations, training and messaging, require the use of moving images such as video and image animation, as well as sound (from the video images as well as overlaid sound by a narrator) intermixed with document images and graphical text displays. Multimedia applications require dynamic handling of data consisting of a mix of text, voice, audio components, video components, and image animation. Integrated multimedia applications allow the user to cut sections of all or any of these components and paste them in a new document or in another application such as an animated sequence of events, a. desktop publishing system, or a spreadsheet.

The components that fall under our definition of multimedia are: '

## Data elements for Multimedia Systems

**Facsimile**Facsimile transmissions were the first practical means of transmitting document images over telephone lines. The basic technology, now widely used, has evolved to allow higher scanning density for better-quality fax

**Document images** Document images are used for storing business documents that must be retained for long periods oftime or may need to be accessed by a large number of people. Providing multimedia access to such documents removes the need far making several copies ofthe original for storage or *distribution*

**Photographic images** Photographic images are used for a wide range of applications . such as employee records for instant identification at a security desk, real estates systems with photographs of houses in the database containing the description of houses, medical case histories, and so on.

**Geographic information systems map (GIS)**

Map created in a GIS system are being used wildly for natural resources and wild life management as well as urban planning. These systems store the geographical information of the map along with a database containing information relating highlighted map elements with statistical or item information such as wild life statistics or details of the floors and rooms and workers in an office building

**Voice commands and voice synthesis**Voice commands and voice synthesis are used for hands-free operations of a computer program. Voice synthbsis is used for presenting the results of an action to the user in a synthesized voice. Applications such as a patient monitoring system in a surgical theatre will be prime beneficiaries of these capabilities. Voice commands allow the user to direct computer operation by spoken commands

**Audio message**Annotated voice mail already uses audio or voice message as attachments to memos and documents such as maintenance manuals.

**Video messages**Video messages are being used in a manner similar to annotated voice mail.

**Holographic images**

All of the technologies so for essentially present a flat view of information. Holographic images extend the concept of virtual reality by allowing the user to get "inside" a part, such as, an engine and view its operation from the inside.

**Fractals**

Fractals started as a technology in the early 1980s but have received serious attention only recently. This technology is based on synthesizing and storing algorithms that describes the information.

## 3.2 MULTIMEDIA APPLICATIONS

The first widely used application of multimedia is document image management. It is primarily intended for scanning documents and retaining their images.

Another application is image processing. It is also known as Image recognition. It is intended for recognizing objects by analyzing their raster images. Applications that present a view of generic multimedia applications are:

**1. Document Imaging**

The fundamental concepts of storage, compression and decompression, and display technologies used for multimedia systems were developed for document image management. Organizations such as insurance agencies law offices,country and state governments, and the federal government manage large volumes of documents.

Document image technology is adopted by Department of Defence for applications ranging from military personnel records to maintenance manuals and high-speed printing systems. Almost all document image system use workflows that are customized for the purpose for which they are being used. The workflow defines the sequence for scanning images, performing data *entry* based on the contents of the lmages, indexing them and storing them on optical media.

### Document Image Hardware requirements:

Realtime image decompression and display place an important role on image processing hardware.

Image decompression and display hardware supports 4 to 8 planes. 4 planes provide 16 colors and 8 planes provide 256 colors. The image planes are also called bit planes, because, they are addressed by a bit in a bytes. Images must be processed at the rate of tens to hundreds of pixels per nano-second. For high-resolution images, processing of the order of 10 pixels/ ns is enough for monochrome still images.

Gray scale images consist of pixels that have shades of gray ranging from 16 to 256. Color images feature color hues instead of shades of gray. Most high-resolution monitors support 16 to 256 colors display capability. The number of colors that can be depicted depends on the number of bits used to define the palette.

### Image processing and Image Recognition

Image processing involves image recognition, Image enhancement,image synthesis, and image reconstruction.

An image processing system may actually alter the contents of the image itself. Image processing systems employ the compression and decompression techniques, a wide range of algorithm for object recognition, comparing images of objects with pre-defined objects, extrapolating finer details to view edges more clearly, gray-scale balancing and gray-scale and color adjustments.

Let us briefly review the various aspects of inlage processing and recognition.

Image enhancement: Most image display systems feature some level of image adjustment.
Increasing the sensitivity and contrast makes the picture darker by making bonderline pixels black or increasing the gray-scale level of pixels.
Capabilities built in the compression boards might include the following

* **Image calibration**: The overall image density is calibrated, and the image pixels are adjusted to a predefined level. * **Real time alignment**: The image is aligned in real-time for skewing caused by improper feeding of paper. * **Gray-Scale normalization**: The overall gray level of an image or picture is evaluated to determine if it is skewed in one direction and if it needs cOlTection. * **RGB hue intensity adjustment**: Too much color makes picture garish and fuzzy. Automatic hue intensity adjustment brings the hue intensity within pre-defined ranges. * **Color Separation**: A picture with very little color contrast can be dull and may not bring out the details. The hardware used can detect and adjust the range of color separation. * **Frame averaging**: The intensity level of the frame is averaged to overcome the effects of very dark or very light areas by adjusting the middle tones.

## IMAGE ANIMATION

Computers-created or scanned images can be displayed sequentially at controlled display speeds to provide image animation that simulates real processes.
The basic concept of displaying successive images at short intervals to give the perception of motion is being used successfully in designing moving parts such as automobile engines.

**Image annotation**

Image annotation can be performed in one of two ways: as a text file stored along with the image or as a small image stored with the original image. The annotation is overlayed over the original image for display purposes. It requires tracking multiple image components associated with a single page, decompressing all of them, and ensuring correct spatial alignment they are overlayed.

## Optical Character Recognition

Data entry is the most expensive component of data processing, because it requires extensive clerical staff work to enter data.
Automating data entry, both typed and handwritten, is a significant application that can provide high returns. Optical Character Recognition (OCR) technology is used for data entry by scanning typed or printed words in a form.
Initially, people used dedicated OCR scanners. Now, OCR Technology is available in software. OCR technology, used as a means of data entry, may be used for capturing entire paragraphs of text. The capturing text is almost a\ways entered as a field in a database or in an editable document

## Handwriting recognition

Research for Handwriting recognition was performed for *CADI* CAM systems for command recognition. Pen-based systems are designed to allow the user to write· commands on an electronic tablet.
Handwriting recognition engines use complex algorithms designed to capture data in real time as it is being input or from an image displayed in a window, depending on the application. Two factors are important for handwriting recognition. They are the strokes or shapes being entered, and the velocity of input or the vectoring that is taking place. The strokes are parsed and processed by a shape recognizer that tries to determine the geometry and topology of the strokes. It attempts to compared it to existing shapes, such as predefined characters. The stroke is compare with the prototype character set until a match is found or all pre-defined prototypes have been checked without a match.
Multimedia system will use handwriting recognition as another means of user input.

## Non-Textual Image Recognition

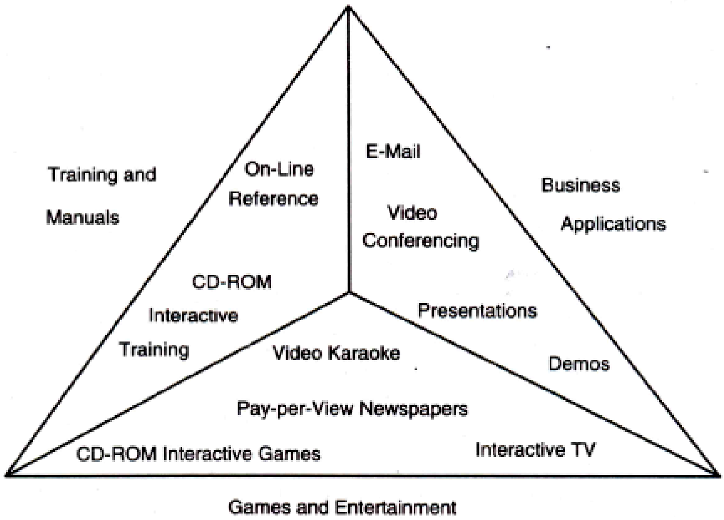
Image recognition is a major technology component in designing, medical and manufacturing fields. Let us review the basic concepts of image recognition architecture.
For example, a general Image recognition system,- the Image Understanding Architecture has the design which calls for three processing layers.

(i) 512 x 512 array of custom pixel processors that extract basic features such as lines and object boundaries. (ii) The features of an object extracted by the first layer are tracked by the DSP array, and that information is fed into 5l2-M byte RAM. (iii) At the highest level, sophisticated AI algorithms perform the difficult task of object and scene recognition. .

## Full motion Digital video Applications

Full motion video has applications in the games industry and training, as well as the business world. Full motion video is the most complex and most demanding component of multimedia applications.

For business applications, some core requirements are needed.
(i) Full-motion video clips should be sharable but should have only one sharable copy.
(ii) It should be possible to attach full-motion video clips to other documents such as memos, chapter text, presentation, and so on.




The following features should be available:
(a) Features, of a VCR metaphor, such as, rewind, fast-forward, play, and search.(b) Ability to move and resize the window displaying the video clip. (c) Ability to view the same clip on a variety of display terminal types with varying resolution capabilities without the need for storing multiple copies in different form (d) Ability to adjust the contrast and brightness of the video clip. (e) Ability to adjust the volume of the associated sound. (I) It should enable the users to place their own indexing marks to locate segments in video clip.

## A Universal Multimedia Application

It is an application that works on universal data type. This means that the application manipulates datatypes that can be combined in a document, displayed 'on a screen, or printed, with no special manipulations that the user needs to perform. The application is truly distributed in nature.
An important consideration for such a universal application is the methodology for dissemination of the information on a network.

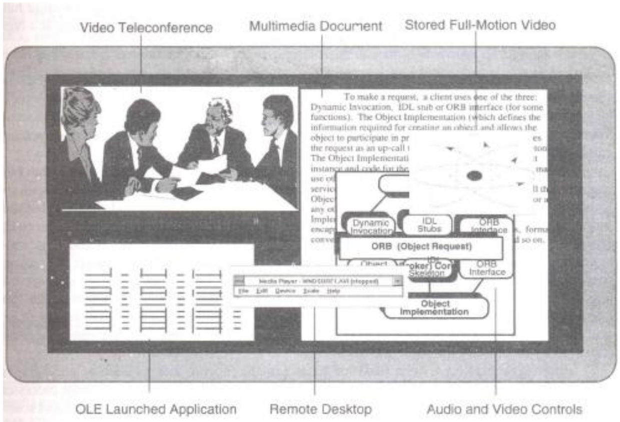



**Figure** describes the user screen for a universal multimedia application. **In** this screen, mix of windows for displaying still video and document images, a video conference window with a live session in progress, a remote live desk top, and a couple of other windows for applications such as electronic mail and desk top publishing.
To maintain all of these windows requires a substantial amount of CPU power. Digital Signal Processing assistance is needed to manage the multiple simultaneous decompressions for JPEG, MPEG and windows applications.

## **3.** Full-Motion Video Messages

In addition to textual messages, electronic mail capability allows embedding of voice messages and video messages. Video messages may consist of video snapshots or live video with full-motion picture and sound.
Two technological concepts at play in the implementation of full motion video messages:

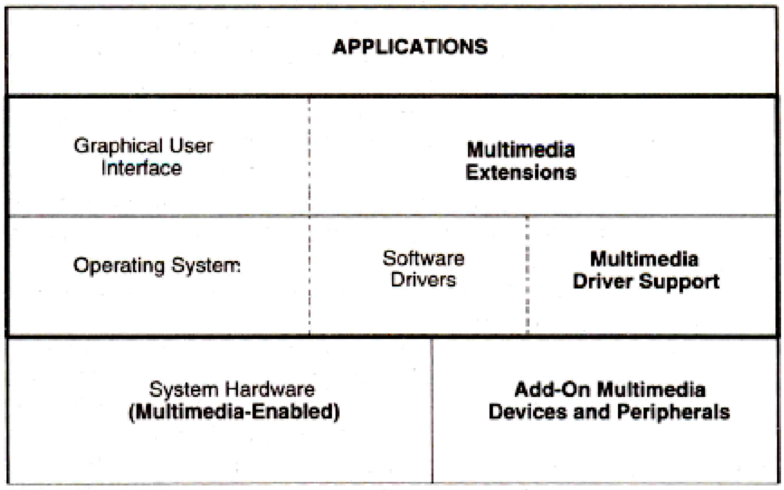
**(i)** The storage and transmitted of a very large volume of data at a high rate,
**(ii)** Decompression of that data to present a continuous play back

## Audio and Video Indexing.

Indexing is an important and complex subject for multimedia design. Marking a position is called Indexing. Audio and video indexing are used in full-motion video in a manner similar to any video sequence, i.e.,. just as it would in a home movie, taped performance and so on.
The needs of the application must be a strong consideration for the type of indexing provideq with the system.
Key points for indexing of stored video clips:
* Indexing is useful only if the video is stored, indexing information is lost.
* When sound and video are decompressed and managed separately, synchronization is very important.
* Depending on the application, indexing information must be maintained separately for sound and video components of a video clip.

## 3.3 Multimedia Systems Architecture

Multimedia encompasses a large variety of technologies and integration of multiple architectures interacting in real time. All of these multimedia capabilities must integrate with the standard user interfaces such as Microsoft Windows.
**The** following figure describes the architecture of a multimedia workstation environment.
In this diagram.




The right side shows the new architectural entities required for supporting multimedia applications.
For each special devices such as scanners, video cameras, VCRs and sound equipment-, a software device driver is need to provide the interface from an application to the device. The GUI require control extensions to support applications such as full motion video

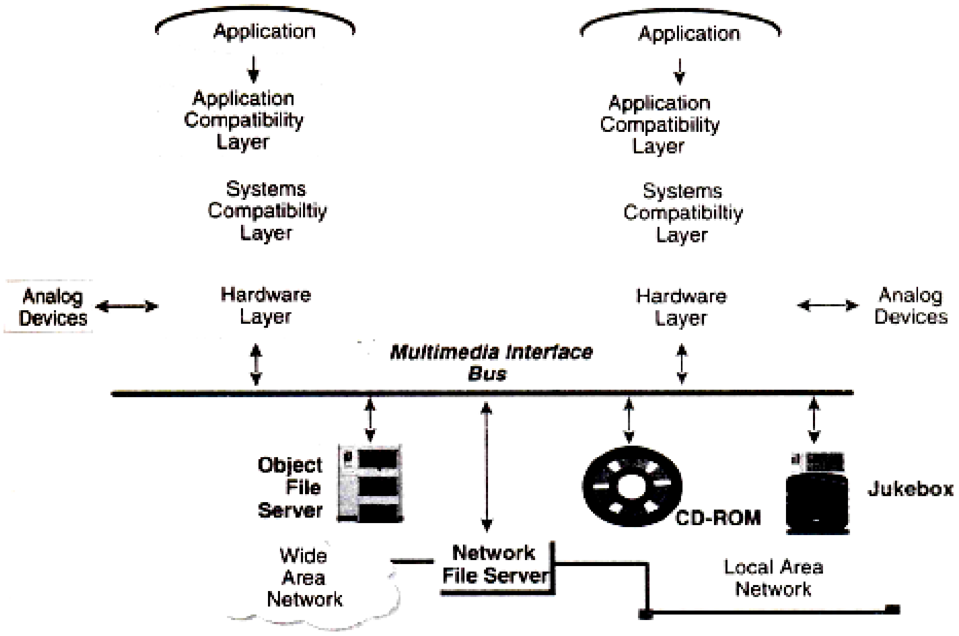
## High Resolution Graphics Display

The various graphics standards such as MCA, GGA and XGA have demonstrated the increasing demands for higher resolutions for GUls.
Combined graphics and imaging applications require functionality at three levels. They are provided by three classes of single-monitor architecture.
**(i) VGA mixing: In** VGA mixing, the image acquisition memory serves as the display source memory, thereby fixing its position and size on screen:
**(ii) VGA mixing with scaling:** Use of scalar ICs allows sizing and positioning of images in pre-defined windows.
Resizing the window causes the things to be retrieved again.
**(iii) Dual-buffered VGA/Mixing/Scaling:** Double buffer schemes maintain the original images in a decompression buffer and the resized image in a display buffer.

## The IMA Architectural Framework

The Interactive Multimedia Association has a task group to define the architectural framework for multimedia to provide interoperability. The task group has C0ncentrated on the desktops and the servers.
Desktop focus is to define the interchange formats. This format allows multimedia objects to be displayed on any work station.

The architectural approach taken by IMA is based on defining interfaces to a multimedia interface bus. This bus would be the interface between systems and multimedia sources. It provides streaming I/O service"s, including filters and translators **Figure 3.4** describes the generalized architectural approach




## Network Architecture for Multimedia Systems:

Multimedia systems need special networks. Because large volumes of images and video messages are being transmitted.

Asynchronous Transfer Mode technology (A TM) simplifies transfers across LANs and W ANs.

## Task based Multi level networking

Higher classes of service require more expensive components in the' workstations as well as in the servers supporting the workstation applications.

Rather than impose this cost on all work stations, an alternate approach is to adjust the class of service to the specific requirement for the user. This approach is to adjust the class of services according to the type of data being handled at a time also.

We call this approach task-based multilevel networking.

## High speed server to server Links

**Duplication:** It is the process of duplicating an object that the user can manipulate. There is no requirement for the duplicated object to remain synchronized with the source (or master) object.

**Replication:** Replication is defined as the process of maintaining two or more copies of the same object in a network that periodically re-synchronize to provide the user faster and more reliable access to the data Replication is a complex process.

## Networking Standards: The two well-known networking standards are Ethernet and token ring.

ATM and FDDI are the two technologies which we are going to discuss in detail.

**ATM:** ATM is a acronym for Asynchronous Transfer Mode. It's topology was originally designed for broadband applications in public networks.

ATM is a method of multiplexing and relaying (cell-switching) 53 byte cells. (48 bytes of user information and 5 bits of header information).

**Cell Switching:** It is a form of fast packet switching based on the use of cells.

**Cells:** Short, fixed length packets are called cells.

ATM provides high capacity, low-latency switching fabric for data. It is independent of protocol and distances. ATM effectively manage a mix of data types, including text data, voice, images and full motion video. ATM was proposed as a means of transmitting multimedia applications over asynchronous networks.

**FDDI:** FDDI is an acronym of Fiber Distributed Data Interface. This FDDI network is an excellent candidate to act as the hub in a network configuration, or as a backbone that interconnects different types of LANs.

FDDI presents a potential for standardization for high speed networks.

The ANSI standard for FDDI allows large-distance networking. It can be used as high-performance backbone networks to complement and extend current LANs.

## 3.4 EVOLVING TECHNOLOGIES FOR MULTIMEDIA SYSTEMS

Multimedia applications use a number of technologies generated for both commercial business application as well as the video game industry.

Let us review some of these technologies in this section.

**Hypermedia documents**

Hypermedia documents are documents which have text, embedded or linked multimedia objects such as image, audio, hologram, or full-motion video.

### Hypertext

Hypertext systems allow authors to link information together, create information paths through a large volume of related text in documents.

It also allows to annotate existing text, and append notes.

It allows fast and easy searching and reading of selected excerpts.

## HYPERMEDIA

It is an extension of hypertext.

In that, we can include texts, any kind of information that can be stored in electronic storage, such as audio, animated video, graphics or full-motion video.

Hypermedia documents used for electronic mail and work flow applications provide a rich functionality for exchanging a variety of information types. The hypermedia document is a definition of a document and a set of pointers to help locate the various elements of the document on the network.

### HYPER SPEECH

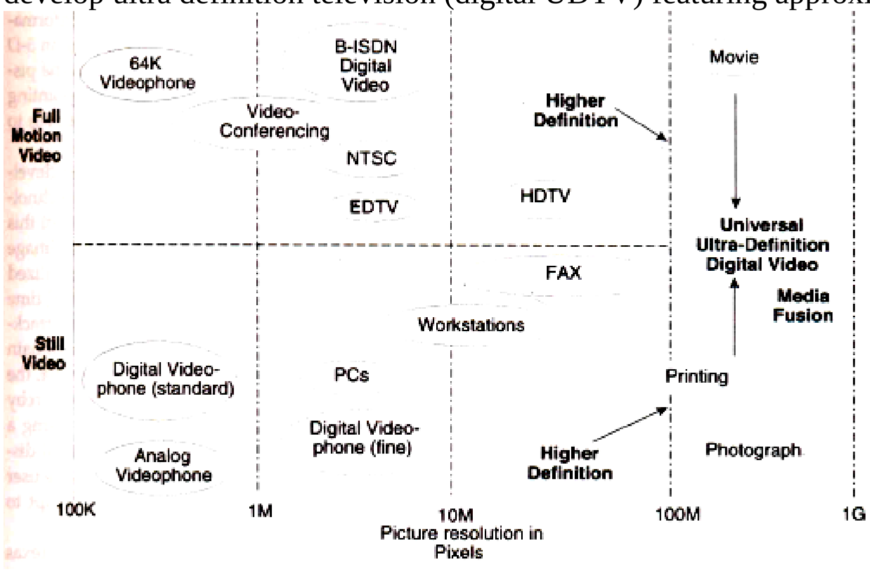
Multimedia stimulated the development of general-purpose speech interfaces. Speech synthesis and speech recognition are fundamental requirement for hyperspeech systems. Speech recognition is nothing but converting the analog speech into a computer action and into ASCII text. Speech-recognition systems cannot segment a stream of sounds without breaks into meaningful units. The user must speak in a stilted fashion. He should make sure to interpose silence between each word.

### HDTV AND UDTV

HDTV is an acronym of High-Definition Television.

The broadcasting standards such as NTSC, PAL, SECAM, NHK have an idea of bringing the world together on a single high-definition Television broadcasting standard.

The japanese broadcasting services developed a 1125-line, along MUSE system. A competing standard in the U.S. changed direction from analog to digital technology:A 1125-line digital HDTV has been developed and is being commercialized. NHK of Japan is trying to leapfrog the digital technology to develop ultra definition television (digital UDTV) featuring approximately 3000 lines



## 3D TECHNOLOGIES AND HOLOGRAPHY

Three-dimensional technologies are concerned with two areas: pointing devices and displays. 3-D pointing devices are essential to manipulate object in a 3-D display system. 3-D displays are achieved using holography techniques.

The techniques developed for holography have been adapted for direct computer use.

## Fuzzy Logic

Fuzzy logic is logic which is used for low-level process controllers.

Use of fuzzy logic in multimedia chips is the key to the emerging graphical interfaces of the future. It is expected to become an integral part of multimedia hardware. Fuzzy logic has mathematical principles. Hence, the application of multimedia can benefit those principles.

**Digital Signal Processing**

Digital Signal Processing are used in applications such as digital servos in hard disk drives, and fax/modems. DSP technology is used in Digital wireless communications, such as personal communication networks (pens), wireless local area networks and digital cordless phones.

## DSP Architectures and Applications

A typical DSP operating system architecture would contain the following subsystems:

**Memory Management:** DSP architectures provide dynamic allocation of arrays from multiple segments, including RAM, SRAM and DRAM.

**Hardware-Interrupt handling:** A DSP operating system must be designed to minimize hardware-interrupt latency to ensure fast response to real time events for applications, such as servo systems.

**Multitasking:** DSPs need real-time kernels that provide pre-emptive multitasking and user-defined and dynamic task prioritization

## INTERTASK SYNCHRONIZATION AND COMMUNICATION

Mechanisms for intertask communication include message queues, semaphores, shared memory, and quick response event flags. Multiple timer services: The ability for the developer to set system clock interrupt managed timers to control and synchronize tasks is needed for most real-time applications.

Device-Independent I/O: DSP operating system should supports

(i) Asynchronous data stream

(ii) Synchronous message passing.

Use of DSP' s has evolved from traditional general purpose digital signal processors to application-specific and customizable DSPs. DSPs were conceived as math engines with a system architecture that was like that of a mini-computer with an array processor.

## 3.5 DEFINING OBJECTS FOR MULTIMEDIA SYSTEMS

The basic data types of object using in multimedia include text, image, audio, holograms and full-motion video.

**TEXT**

It is the simplest of data types and requires the least amount of storage. Text is the base element of a relational database.

It is also the basic building of a document.

The major attributes of text include paragraph styling, character styling, font families and sizes, and relative location in a document

**HYPERTEXT**

It is an application of indexing text to provide a rapid search of specific text strings in one or more documents. It is an integral component of hypermedia documents. A hypermedia document is the basic complex object of which text is a sub object.

Sub-objects include images, sound and full motion video.

A hypermedia document always has text and has one or more other types of sub-objects

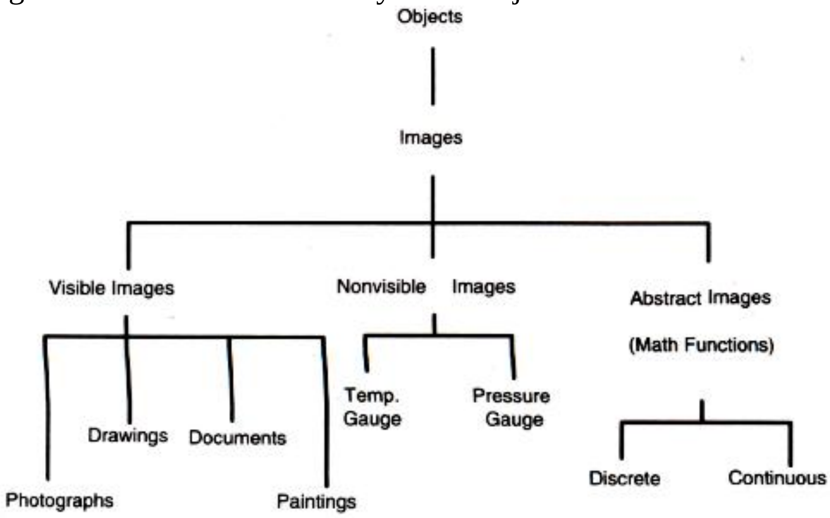
**IMAGES**

Image object is an object that is represented in graphics or encoded form. Image object is a subobject of the hypermedia document object. In this object, there is no direct relationship between successive representations in time.

The image object includes all data types that are not coded text. It do not have a temporal property associated with them.

Thc data types such as document images, facsimile systems, fractals, bitmaps, meta files, and still pictures or still video frames are grouped together.

Figure 3.6 describes a hierarchy of the object classes




**Non-Visible**: This type of images are not stored as images. But they are displayed as images. Example: Pressure gauges, and temperature gauges.

**Abstract:** Abstract images are computer-generated images based on some arithmetic calculations. They are really not images that ever existed as real-world objects. Example of these images is fractals.

**AUDIO AND VOICE**

Stored-Audio and Video objects contain compressed audio information. This can consist of music, speech, telephone conversation and voice commands. An Audio object needs to store information about thc sound clip.

Information here means length of the sound clip, its compression algorithm, playback characteristics, and any annotations associated with the original clip.

## FULL MOTION AND LIVE VIDEO

Full motion video refers to pre-stored video clips. Live video refers to live and it must be processed while it is being captured by the camera. . From a storage perspective, we should have the information about the coding algorithm used for compression. It need decoding also.

From a processing perspective, video should be presented to user with smooth and there should not be any unexpected breaks.

Hence, video object and its associated audio object must be transferred over the network to the decompression unit. It should be then played at the fixed rate specified for it.

For successful playback of compressed video, there are number of technologies. They are database storage, network media and protocols, decompression engines and display engines.

## 3.6 MULTIMEDIA DATA INTERFACE STANDARDS

### File Formats for Multimedia Systems:

(i) **Device-independent Bitrnap (DIB):** This file format contains bit map, color, and color paJlette information.

(ii) **RIFF device Independent Bitrnap (RDIB):** Resource Interchange File Frmat **(RIFF)** is the standard file format defined for Microsoft Windows and OS/2. It allows a more complex set of bit maps than can be handled by DIB.

(iii) **Musical Instrument Digital interface (MIDI):** This is the interface standard for file transfer between a computer and a musical instrument such as a digital piano. It is also, used for full-motion video and voice-mail messaging systems. It has the advantage of ready availability of MIDI device controller boards for personal computers.

**RIFF Musical Instrument Digital Interface**

A MIDI format within a RIFF envelope provides a more complex interface.

**Palette File Format (PAL)**An interface that allows defining a palette of 1 to 256 colours in a representation as RGB values.

**Rich Text Format (RTF)** This file format allows embedding graphics and other file formats within a document. This format is used by products such as Lotus Notus. This format is also the basis for the use of OLE.
**Waveform Audio File Format (WAVE)** A digital file representation of digital audio.
**Windows Metafile Format (WMF)** This is a vector graphic format used by Microsoft Windows as an interchange format.
**Multimedia Movie Format (MMM)** This is a format used for digital video animation.
**Apple's Movie Format** This format was defined as the standard for file exchange by Quick Time enabled systems.
**Digital Video Command Set (DVCS)** This is the set of digital video commands simulating VCR controls.
**Digital Video Media Control Interface** Microsoft's high level control interface for VCR controls, including play, rewind, record and so on.
**Vendor - Independent Messaging (VIM)** Developed by a consortium of Vendors providing a standardized format for cross-product messages.
**Apple's Audio Interchange File Format** Apple's standard file format for compressed audio and voice data.
**SDTS GIS Standard** The Spatial Data Transfer Standard (SDTS) is designed to provide a common storage format for geographic and cartographic data.

VIDEO PROCESSING STANDARDS

INTELS DVI

DVI is an achronym of Digital Video Interface.

DVI standard is to provide a processor independent specification for a video interface. That video interface should accomodate most compression algorithms for fast multimedia displays. An example of custom-designed chip which supports DVI is Intel's i750 B. This chip is designed for enhancing low-end, software based PC video.

**Advantages of the DVI Chip**

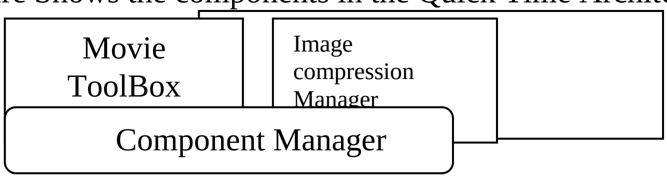
(i) It can operate software video processing real time. (ii) It can share the processing with the host CPU. (iii) It can handle additional vector-quantization-type algorithms in conjunction with host processing. DVI silicon chip relies on a programmable video processor. It gives potential to DVI chips to run a range of compression algorithms.

## APPLE QUICK TIME

Quick Time standard is developed by Apple Computer. It is designed to Support multimedia applications. It is integrated with the operating system. Quick time refers to both the extensions to the Mac Operating system and to the compression/decompression functionality Of the environment. Quick Time is designed to be the graphics standard for timebased graphic data types.

Quick Time's definition has been extended to include (i) System Software, (ii) File Formats, (Hi) Compression! decompression algorithms, (iv) Human Interface Standards.

Figure Shows the components in the Quick Time Architecture.




Quick Time adjust automatically to the hardware being used by the user. MPEG is another competing standard which is comparitively higher-end, hardware-assisted standard. It can produce better resolutions at faster rates.

## MICROSOFT AVI

A VI is an achronym for Audio Video Interleave Standard. It is similar to Apple's Quick Time. It offers low-cost, low-resolution video processing for the average desktop user. It is a layered product. A VI is scalable. It allows users to set parameter such as window size, frame rate, quality and compression algorithm through a number of dialog boxes. AVI-compatible hardware allows enhancing performance through hardware-accelerated compression algorithms such as DVI and MPEG. A VI supports several compression algorithms

## 3.7 MULTIMEDIA DATABASES

Images, sounds and movies can be stored, retrieved and played by many databases. In future, multimedia databases will bcome a main source of interaction between users and multimedia elements.

**Multimedia storage and retrieval** Multimedia storage is characterized by a number of considerations. They are:

(i) massive storage volumes
(ii) large object sizes
(iii) multiple related objects
(iv) temporal requirements for retrieval

**Massive Data Volumes**

A single multimedia document may be a combination of different media Hence indexing of documents, fi lms and tapes is more complex. Locating massive data volumes requires searching through massive storage files.

Locating and indexing systems can be understood only by a few key staff personnel. Hence it requires a major organizational eff0l1 to ensure that they are returned in proper sequence to their original storage location.

**storage technologies**

There are two major mass storage technologies used currently for storage of multimedia documents.

(i) Optical disk storage systems. (ii) High-speed magnetic storage.

**Advantages of Optical disk storage systems:**

(i) Managing a few optical disk platters in a juke box is much simpler than man;lging a large magnetic disk farm. (ii) Optical disk storage is excellent storage system for off line archival of old and infrequently referenced documents for significant periods of time

**Multimedia object storage**

Multimedia object storage in an optical medium serves its original purpose, only if it can be located fast and automatically. A key issue here is random keyed Access t6 various components of hypermedia database record. Optical media providcs very dense storage. Speed of retrieval is another consideration.

Retrieval speed is a direct result of the storage latency, size of the data relative to display resolution, transmission media and speed, and decompression efficiency. Indexing is important for fast retrieval of information. Indexing can be at multiple levels.

## Multimedia document retrieval

The simplest form of identifying a multimedia document is by storage platter identification and its relative position on the platter (file number). These objects can then be grouped using a database in folders (replicating the concept of paper storage in file folders) or within complex objects representing hypermedia documents.

The capability to access objects using identifiers stored in a database requires capability in the database to perform the required multimedia object directory functions. Another important application for sound and full motion video is the ability to clip parts of it and combine them with another set.

Indexing of sound and full-motion video is the subject of intense debate and a number of approaches have been used.

## Database Management Systems for Multimedia Systems

Since most multimedia applications are based primarily on communications technologies, such as electronic mail, the database system must be fully distributed. A number of database storge choices are available.

The choices available are:

* Extending the existing relational database management systems, (RDBMSs) to support the various objects for multimedia as binary objects.
* Extending RDBMSs beyond basis binary objects to the concepts of inheritance and classes. RDBMSs supporting these . features provide extensions for object-programming front ends and/or C++ support.
* Converting to a full fledged object oriented database that supports the standard SQL language.
* Converting the database and the application to an objectoriented database and using an object-oriented language, or an object-enabled SQL for development.

Multimedia applications combine numerical and textual data, graphics from GUI front-ends, CAD/CAM systems and GIS applications, still video, audio and full-motion video with recorded audio and annotated voice components. Relational databases, the dominent database paradigm, have lacked the ability to support multimedia databases. Key limitations of relational database systems for implementing multimedia applications stem from two areas: the relational data model and the relatIonal computational model.
RDBMSs have been designed to manage only tabular alphanumeric forms of data (along with some additional data types stored in binary form such as dates).

**RDBMS EXTENSIONS FOR MULTIMEDIA**

Binary Large Object (BLOB) is a data type which has been adapted by most of the leading relational databases. BLOBs are used for objects such as images or other binary data types.
The relational database is extended to access these BLOBs to present the user 'with a complete' data set.
Extended relational databases provide a gradual migration path to a more object-oriented environment.
Relational database tables include location information for the BLOBs which may be stored outside the database on separate image or video servers. Relational databases have the strength of rigorous set management for maintaining the integrity of the database

## Object-Oriented Databases for Multimedia

.**In** object databases, data remains in RMS or flat files. Object databases can provide the fastest route to multimedia support. Object programming embodies the principles of reusable code and modularity. This will ease future maintenance of these databases.
Object database capabilities such as message passing, extensibility, and the support of hierarchical structures, are important for multimedia systems.
We can develop the application fastest class definitions. ODBMSs are extensible. They allow incremental changes to the database applications.
**Extensibility:** Extensibility .means that the set of operations, structures and constraints that are available to operations are not fixed, and developers can define new operations, which can then be added as needed to their application.
Object-oriented software technology has three important concepts. They are:
**Encapsulation: It** is the ability to deal with software entities as units that interact in pre-defined and controllable manner, and where the control routines are integral with entity.
**Association**: It is the ability to define a software entity in terms of its di fferences from another entity.
**Classification**: It is the ability to represent with a single software entity a number of data items that all have the same behavior and the same state attributes.
Object orientation helps to organize the software in a more, modular and re-usable manner.
**Encapsulation** allows for the development of open systems where one part of the application does not need to know the functioning of other part. It also provides autonomy; **Autonomy** means we can interface to a variety of external programs can be built in one class of objects and the storage of the data in another class of objects.

## Database Organization for Multimedia Applications

Data organization for multimedia systems has some key issues. They are:
(l) Data independence (2) Common distributed database architecture
(3) Distributed database servers· (4) Multimedia object management.

**Data Independence**

Flexible access by a number of databases requires that the data be independent from the application so that future applications can access the data without constraints related to a previous application.
Key features of data independent designs are:

1.Storage design in independent of specific applications.

2.Explicit data definitions are independent of application program.

3.Users need not know data formats or physical storage structures.

4.Integrity assurance in independent of application programs.

5.Recovery in independent of application programs.

## Distributed Data servers

Distributed database servers are a dedicated resource on a network accessible to a number of applications. The database server is built for growth and enhancement, and the network provides the opportunity for the growth of applications and distributed access to the data.

## Multimedia Object Management

The object management system must be capable of indexing, grouping and storing multimedia objects in distributed hierarchical optional storage systems, and accessing these objects on or keyed basis.

The design of the object management system should be capable indexing objects in such a manner that there is no need to maintain multiple storage copies.

Multimedia transactions are very complex transactions. We define a multimedia transaction as the sequence of events that starts when a user makes a request to display, edit, or print a hyper media document. The transaction is complete when the user releases the hypermedia document and stores back the edited versions or discards the copy in memory (including virtual memory) or local storage ..

# UNIT IV -MULTIMEDIA FILE HANDLING

Compression and decompression − Data and file format standards − Multimedia I/Otechnologies − Digital voice and audio − Video image and animation − Full motionvideo − Storage and retrieval technologies.

## 4.1 COMPRESSION AND DECOMPRESSION

Compression is the way of making files to take up less space. In multimedia systems, in order to manage large multimedia data objects efficiently, these data objects need to be compressed to reduce the file size for storage of these objects.

Compression tries to eliminate redundancies in the pattern of data.

For example, if a black pixel is followed by 20 white pixels, there is no need to store all 20 white pixels. A coding mechanism can be used so that only the count of the white pixels is stored. Once such redundancies are removed, the data object requires less time for transmission over a network. This in turn significantly reduces storage and transmission costs.

## TYPES OF COMPRESSION

Compression and decompression techniques are utilized for a number of applications, such as facsimile system, printer systems, document storage and retrieval systems, video teleconferencing systems, and electronic multimedia messaging systems. An important standardization of compression algorithm was achieved by the CCITT when it specified Group 2 compression for facsimile system. .

When information is compressed, the redundancies are removed.

Sometimes removing redundancies is not sufficient to reduce the size of the data object to manageable levels. In such cases, some real information is also removed. The primary criterion is that removal of the real information should not perceptly affect the quality of the result. In the case of video, compression causes some information to be lost; some information at a delete level is considered not essential for a reasonable reproduction of the scene. This type of compression is called lossy compression. Audio compression, on the other hand, is not lossy. It is called lossless compression.

## Lossless Compression.

In lossless compression, data is not altered or lost in the process of compression or decompression. Decompression generates an exact replica ofthe original object. Text compression is a good example of lossless compression. The repetitive nature of text, sound and graphic images allows replacement of repeated strings of characters or bits by codes. Lossless compression techniques are good for text data and for repetitive data in images all like binary images and gray-scale images.

Some of the commonly accepted lossless standards are given below:

- **Packpits encoding (Run-length encoding)**
- **CCITT Group 3 I D**
- **CCITT Group 3 2D**
- **CCITT Group 4**
- **Lempe l-Ziv and Welch algorithm LZW.**

Lossy compression is that some loss would occur while compressing information objects.

Lossy compression is used for compressing audio, gray-scale or color images, and video objects in which absolute data accuracy is not necessary.

The idea behind the lossy compression is that, the human eye fills in the missing information in the case of video.

But, an important consideration is how much information can be lost so that the result should not affect. For example, in a grayscale image, if several bits are missing, the information is still perceived in an acceptable manner as the eye fills in the gaps in the shading gradient.

Lossy compression is applicable in medical screening systems, video tele-conferencing, and multimedia electronic messaging systems.
Lossy compressions techniques can be used alone o~' in combination with other compression methods in a multimedia object consisting of audio, color images, and video as well as other specialized data types.
The following lists some of the lossy compression mechanisms:

- ✓ Joint Photographic Experts Group (JPEG)
- ✓ Moving Picture Experts Group (MPEG)
- ✓ Intel DVI
- ✓ CCITT H.261 (P * 24) Video Coding Algorithm
- ✓ Fractals.

Compression schemes are

## Binary Image compression schemes

Binary Image Compression Scheme is a scheme by which a binary image containing black and white pixel is generated when a document is scanned in a binary mode.
The schemes are used primarily for documents that do not contain any continuous-tone information or where the continuous-tone information can be captured in a black and white mode to serve the desired purpose.
The schemes are applicable in office/business documents, handwritten text, line graphics, engineering drawings, and so on. Let us view the scanning process.A scanner scans a document as sequential scan lines, starting from the top of the page.
A scan line is complete line of pixels, of height equal to one pixel, running across the page. It scans the first line of pixels (Scan Line), then scans second "line, and works its way up to the last scan line of the page. Each scan line is scanned from left to right of the page generating black and white pixels for that scan line.
This uncompressed image consists of a single bit per pixel containing black and white pixels. Binary 1 represents a black pixel, binary 0 a white pixel. Several schemes have been standardized and used to achieve various levels of compressions. Let us review the more commonly used schemes.

### 1 .Packpits Encoding( Run-Length Encoding)

It is a scheme in which a consecutive repeated string of characters is replaced by two bytes. It is the simple, earliest of the data compression scheme developed. It need not to have a standard. It is used to compress black and white (binary) images. Among two bytes which are being replaced, the first byte contains  a number representing the number of times the character is repeated, and the second byte contains the character itself.
In some cases, one byte is used to represent the pixel value, and the other seven bits to represents the run length.

**2. CCITT Group 3 1-D Compression**

This scheme is based on run-length encoding and assumes that a typical scanline has long runs of the same color.
This scheme was designed for black and white images only, not for gray scale or color images. The primary application of this scheme is in facsimile and early document imaging system.

### Huffman Encoding

A modified version of run-length encoding is Huffman encoding.
It is used for many software based document imaging systems. It is used for encoding the pixel run length in CCITT Group 3 1-dGroup 4.
It is variable-length encoding. It generates the shortest code for frequently occurring run lengths and longer code for less frequently occurring run lengths.

**Mathematical Algorithm for huffman encoding:**

Huffman encoding scheme is based on a coding tree.
It is constructed based on the probability of occurrence of white pixels or black pixels in the run length or bit stream.
Table below shows the CCITT Group 3 tables showing codes or white run lengths and black run lengths.

| **White Run** | **Code** | **Black Run** | **Code** |
|---|---|---|---|

| **Length** | **Word** | **Length** | **Word** |
|---|---|---|---|
| 0 | 00110101 | 0 | 0000110111 |
| 1 | 000111 | 1 | 010 |
| 2 | 0111 | 2 | 11 |
| 3 | 1000 | 3 | 10 |
| 4 | 1011 | 4 | 011 |
| 5 | 1100 | 5 | 0011 |
| 6 | 1110 | 6 | 0010 |
| 7 | 1111 | 7 | 00011 |
| 8 | 10011 | 8 | 000101 |
| 9 | 10100 | 9 | 000100 |
| 10 | 00111 | 10 | 0000100 1 |
| 11 | 01000 | 10 | 0000100 |
| 11 | 01000 | 11 | 0000101 |
| 12 | 001000 | 12 | 0000111 |
| 13 | 000011 | 13 | 00000100 |
| 14 | 110100 | 14 | 00000111 |
| 15 | 110101 | 15 | 000011000 |
| 16 | 101010 | 16 | 0000010111 |
| 17 | 101011 | 17 | 0000011000 |
| 18 | 0100111 | 18 | 0000001000 |
| 19 | 0001100 | 19 | 0000 11 00 III |
| 20 | 0001000 | 20 | 00001101000 |
| 21 | 0010111 | 21 | 00001101100 |
| 22 | 0000011 | 22 | 00000110111 |
| 23 | 0000100 | 23 | 00000101000 |
| 24 | 0101000 | 24 | 00000010111 |
| 25 | 0101011 | 25 | 00000011000 |
| 26 | 0010011 | 26 | 000011001010 |
| 27 | 0100100 | 27 | 000011001011 |
| 28 | 0011000 | 28 | 000011 00 11 00 |
| 29 | 00000010 | 29 | 000011001101 |
| 30 | 00000011 | 30 | 000001101000 |
| 31 | 00011010 | 31 | 000001101001 |
| 32 | 00011011 | 32 | 000001101010 |
| 33 | 00010010 | 33 | 000001101011 |
| 34 | 00010011 | 34 | 000011010010 |
| 35 | 00010100 | 35 | 000011 0 10011 |

For example, from **Table 2,** the run-length code of 16 white pixels is 101010, and of 16 black pixels 0000010111. Statistically, the occurrence of 16 white pixels is more frequent than the occurrence of 16 black pixels. Hence, the code generated for 16 white pixels is much shorter. This allows for quicker decoding. For this example, the tree structure could be constructed.

| | | | |
|---|---|---|---|
| 36 | 00010101 | 36 | 000011010100 |
| 37 | 00010110 | 37 | 000011010101 |
| 38 | 000101 II | 38 | 000011010110 |

| | | | |
|---|---|---|---|
| 39 | 00101000 | 39 | 000011 0 1 0 1 1 1 |
| 40 | 00101001 | 40 | 000001I01100 |
| 41 | 00101010 | 41 | 000001101101 |
| 42 | 00101011 | 42 | 000011011010 |
| 43 | 00101100· | 43 | 0000 11 0 1 1011 |
| 44 | 00101101 | 44 | 000001010100 |
| 45 | 00000100 | 45 | 000001010101 |
| 46 | 00000101 | 46 | 000001010110 |
| 47 | 00001010 | 47 | 000001010111 |
| 48 | 00001011 | 48 | 000001100100 |
| 49 | 01010010 | 49 | 000001100101 |
| 50 | 010100II | 50 | 000001010010 |
| 51 | 01010100 | 51 | 000001010011 |
| 52 | 01010101 | 52 | 000000100100 |
| 53 | 00100100 | 53 | 000000110111 |

The codes greater than a string of 1792 pixels are identical for black and white pixels. A new code indicates reversal of color, that is, the pixel Color code is relative to the color of the previous pixel sequence.

**Table 3** shows the codes for pixel sequences larger than 1792 pixels.

| **Run Length** (Black and White) | **Make-up Code** |
|---|---|
| 1792 | 00000001000 |
| 1856 | 00000001100 |
| 1920 | 00000001101 |
| 1984 | 000000010010 |
| 2048 | 000000010011 |
| 2112 | 000000010100 |
| 2176 | 000000010101 |
| 2240 | 000000010110 |
| 2304 | 000000010111 |
| 2368 | 000000011100 |
| 2432 | 000000011101 |
| 2496 | 000000011110 |
| 2560 | 000000011111 |

CCITT Group 3 compression utilizes Huffman coding to generate a set of make-up codes and a set of terminating codes for a given bit stream. Make-up codes are used to represent run length in multiples of 64 pixels. Terminating codes are used to represent run lengths of less than 64 pixels.

As shown in **Table** 2; run-length codes for black pixels are different from the run-length codes for white pixels. For example, the run-length code for 64 white pixels is 11011. The run length code for 64 black pixels is 0000001111. Consequently, the run length of 132 white pixels is encoded by the following two codes:

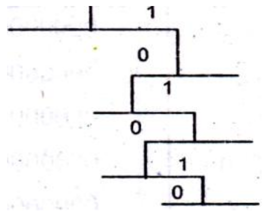
Makeup code for 128 white pixels - 10010

Terminating code for 4 white pixels - 1011

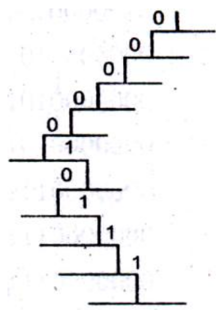
The compressed bit stream for 132 white pixels is 100101011, a total of nine bits. Therefore the compression ratio is 14, the ratio between the total number of bits (132) divided by the number of bits used to code them (9).

CCITT Group 3 uses a very simple data format. This consists of sequential blocks of data for each scanline, as shown in **Table 4.**

Coding tree for 16 white pixels




Coding tree for 16 black pixels




Note that the file is terminated by a number of EOLs (End of. Line) if there is no change in the line [rom the previous line (for example, white space).

TABLE 4: CCITT Group 3  1D File Format

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| EOL | DATA LINE<br><br>1 | FILL | EOL | DATA LINE<br><br>2 | FILL | EOL… | DATA LINE<br><br>n | FILL | EOL | EOL | EOL |

**Advantages of CCITT Group 3 ID**

CCITT Group 3 compression has been used extensively due to the following two advantages:

It is simple to implement in both hardware and software .

It is a worldwide standard for facsimile which is accepted for document imaging application. This allows document imaging applications to incorporate fax documents easily.

- CCITT group 3 compressions utilizes Huffman coding to generate a set of make-up codes and a set of terminating codes for a give bit stream.
- CCITT Group 3 uses a very simply data format. This consists of sequential blocks of data for each scanline.

## CCITT Group 3 2D Compression

It is also known as modified run length encoding. It is used for software based imaging system and facsimile.

It is easier to decompress in software than CCITT Group 4. The CCITT Group 3 2D scheme uses a "k" factor where the

image is divided into several group of k lines. This scheme is based on the statistical nature of images; the image data across the adjacent scanline is redundant.

If black and white transition occurs on a given scanline, chances are the same transition will occur within + or - 3 pixels in the next scanline.

## **Necessity of k factor·**

When CCITT Group 3 2D compression is used, the algorithm embeds Group 3 1 D coding between every k groups of Group 3 2D coding, allowing the Group 3 1 D coding to be the synchronizing line in the event of a transmission error.

Therefore when a transmission error occurs due to a bad communication link, the group 3 I D can be used to synchronize and correct the error.

## **Data formatting for CClIT Group 3 2D**

The 2D scheme uses a combination of additional codes called vertical code, pass code, and horizontal code to encode every line in the group of k lines.

The steps for pseudocode to code the code line are:

**(i)** Parse the coding line and look for the change in the pixel value. (Change is found at al location). **(ii)** Parse the reference line and look for the change in the pixel value. (Change is found at bl location). **(iii)** . Find the difference in location between bland a 1: delta = b1- al

**Advantage of CClIT Group 3 2D**

- The implementation of the k factor allows error-free

transmission . Compression ratio achieved is better than CClTT Group 3 1 D .

- It is accepted for document imaging applications.

**Disadvantage**It doesn't provide dense compression

## CCITT Group 4 2D compression

CClTT Group 4 compression is the two dimensional coding scheme without the k-factor.

In this method, the first reference line is an imaginary all-white line above the top of the image.The first group of pixels (scanline) is encoded utilizing the imaginary white line as the reference line.

The new coded line becomes the references line for the next scan line. The k-factor in this case is the entire page of line. In this method, there are no end-of-line (EOL) markers before the start of the compressed data

## COLOR, GRAY SCALE AND STILL-VIDEO IMAGE COMPRESSION

**Color:**

Color is a part of life we take for granted. Color adds another dimension to objects. It helps in making things standout.

Color adds depth to images, enhance images, and helps set objects apart from -background.

Let us review the physics of color. Visible light is a form of electromagnetic radiation or radiant energy, as are radio frequencies or x-rays. The radiant energy spectrum contains audio frequencies, radio frequencies, infrared, visible light, ultraviolet rays, x-rays and gamma rays.

Radian energy is measured in terms of frequency or wavelength. The relationship between the two is

$\lambda = \frac{c}{f}$ meters

where $\lambda$ - is the wavelength in meters

c is the velocity of light in meters per second.

$f$ is frequency of the radiation in hertz.

Since all electromagnetic waves travel through space at the velocity of light i.e., 3 x $10^8$ meters/second - the wavelength is calculated by

$$\lambda = \frac{3}{f} \times 10^{8} \text{ meters}$$

**Color Characteristics**

We typically define color by its brightness, the hue and depth of color.

**Luminance or Brightness**

This is the measure of the brightness of the light emitted or reflected by an object; it depends on the radiant, energy of the color band.

**Hue** This is the color sensation produced in an observer due to the presence of certain wavelengths of color. Each wavelength represents a different hue.

**Saturation** This is a measure of color intensity, for example, the difference between red and pink.

**Color Models** Several calm' models have been developed to represent color mathematically.

**Chromacity Model** It is a three-dimensional model with two dimensions, x and y, defining the color, and the third dimension defining the luminance. It is an additive model since x and yare added to generate different colors.

**RGBModel** RGB means Red Green Blue. This model implements additive theory in that different intensities of red, green and blue are added to generate various colors.

**HSI Model** The Hue Saturation and Intensity (HSI) model represents an artist's impression of tint, shade and tone. This model has proved suitable for image processing for filtering and smoothing images.

**CMYK Model** The Cyan, Magenta, Yellow and Black color model is used in desktop publishing printing devices. It is a color-subtractive model and is best used in color printing devices only.

**YUV Representation** The NTSC developed the YUV three-dimensional color model.

y -Luminance Component

UV -Chrominance Components.
Luminance component contains the black and white or grayscale information. The chrominance component contains color information where U is red minus cyan and V is megenta minus green.

**YUV Model for JPEG**

The *JPEG* compression scheme uses several stages.
The first stage converts the signal from the spatial RGB domain to the YUV frequency domain by performing discrete cosine transform. This process allows separating luminance or gray-scale components from the chrominance components of the image.

## JOINT PHOTOGRAPHIC EXPERTS GROUP COMPRES5ION (JPEG)

ISO and CCITT working committee joint together and formed Joint Photographic Experts Group. It is focused exclusively on still image compression.
Another joint committee, known as the Motion Picture Experts Group (MPEG), is concerned with full motion video standards.
JPEG is a compression standard for still color images and grayscale images, otherwise known as continuous tone images.
JPEG has been released as an ISO standard in two parts

- Part I specifies the modes of operation, the interchange formats, and the encoder/decoder specifies for these modes along with substantial implementation guide lines .
- Part 2 describes compliance tests which determine whether the implementation of an encoder or decoder conforms to the standard specification of part I to ensure interoperability of systems compliant with JPEG standards

**Requirements addressed by JPEG**

- The design should address image quality .
- The compression standard should be applicable to practically any kind of continuous-tone digital source image .
- It should be scalable from completefy lossless to lossy ranges to adapt it.
- It should provide sequential encoding .
- It should provide for progressive encoding .
- It should also provide for hierarchical encoding .
- The compression standard should provide the option of lossless encoding so that images can be guaranteed to provide full detail at the selected resolution when decompressed.

**Definitions in the JPEG Standard**

The JPEG Standards have three levels of definition as follows:
* Base line system
* Extended system
* Special lossless function.

The base line system must reasonably decompress color images, maintain a high compression ratio, and handle from 4 bits/pixel to 16 bits/pixel.
The extended system covers the various encoding aspects such as variable-length encoding, progressive encoding, and the hierarchical mode of encoding.
The special lossless function is also known as predictive lossless coding. It ensures that at the resolution at which the image is no loss of any detail that was there in the original source image.

**Overview of JPEG Components** JPEG Standard components are:

(i) Baseline Sequential Codec
(ii) OCT Progressive Mode
(Hi) Predictive Lossless Encoding
(iv) Hierarchical Mode.

These four components describe four four different levels of jPEG compression.
The baseline sequential code defines a rich compression scheme the other three modes describe enhancements to this baseline scheme for achieving different results.
Some of the terms used in JPEG methodologies are:

**Discrete Cosine Transform (OCT)**

OCT is closely related to Fourier transforms. Fourier transforms are used to represent a two dimensional sound signal.

DCT uses a similar concept to reduce the gray-scale level or color signal amplitudes to equations that require very few points to locate the amplitude in Y-axis X-axis is for locating frequency.

**DCT Coefficients**

The output amplitudes of the set of 64 orthogonal basis signals are called OCT Co-efficients.

**Quantization** This is a process that attempts to determine what information can be safely discarded without a significant loss in visual fidelity. It uses OCT co-efficient and provides many-to-one mapping. The quantization process is fundamentally lossy due to its many-to-one mapping.

**De Quantization** This process is the reverse of quantization. Note that since quantization used a many-to-one mapping, the information lost in that mapping cannot be fully recovered

**Entropy Encoder / Decoder** Entropy is defined as a measure of randomness, disorder, or chaos, as well as a measure of a system's ability to undergo spontaneous change. The entropy encoder compresses quantized DCT co-efficients more compactly based on their spatial characteristics. The baseline sequential. codec uses Huffman coding. Arithmetic coding is another type of entropy encoding

**Huffman Coding** Huffman coding requires that one or more sets of huff man code tables be specified by the application for encoding as well as decoding. The Huffman tables may be pre-defined and used within an application as defaults, or computed specifically for a given image.

**Baseline Sequential codec**

It consists of three steps: Formation of DCT co-efficients quantization, and entropy encoding. It is a rich compression scheme.

**DCT Progressive Mode**

The key steps of formation of DCT co-efficients and quantization are the same as for the baseline sequential codec. The key difference is that each image component is coded in multiple scans instead ofa single scan.

**Predictive Lossless Encoding**

It is to define a means of approaching lossless continuous-tone compression. A predictor combines sample areas and predicts neighboring areas on the basis of the sample areas. The predicted areas are checked against the fully loss less sample for each area.

The difference is encoded losslessly using huffman on arithmetic entropy encoding .

**Hierarchical Mode**

The hierarchical mode provides a means of carrying multiple resolutions. Each successive encoding of the image is reduced by a factor of two, in either the horizontal or vertical dimension.

**JPEG Methodology**

The JPEG compression scheme is lossy, and utilizes forward discrete cosine transform (or forward DCT mathematical function), a uniform quantizer, and entropy encoding. The DCT function removes data redundancy by transforming data from a spatial domain to a frequency domain; the quantizer quantizes DCT co-efficients with weighting functions to generate quantized DCT co-efficients optimized for the human eye; and the entropy encoder minimizes the entropy of quantized DCT co-efficients.

The JPEG method is a symmetric algorithm. Here, decompression is the exact reverse process of compression.

**Figure** below describes a typical DCT based encoder and decoder.

Symmetric Operation of DCT based Codec

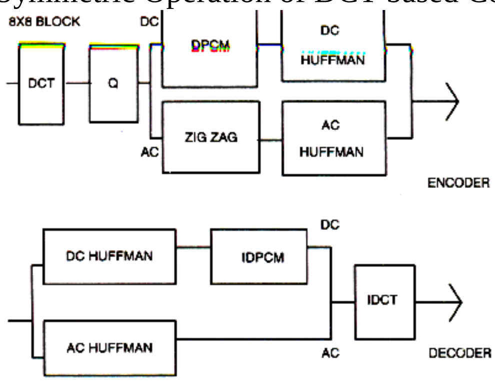



**Figure** below shows the components and sequence of quantization

5 * 8

Image blocks

**Quantization**

Quantization is a process of reducing the precision of an integer, thereby reducing the number of bits required to store the integer, thereby reducing the number of bits required to store the integer.

The baseline JPEG algorithm supports four color quantization tables and two huffman tables for both DC and AC DCT co-efficients. The quantized co-efficient is described by the following equation: Quantized Co-efficient (i, j) = $\frac{DCT(i, j)}{Quantum(i, j)}$

**ZigZag Sequence**

Run-length encoding generates a code to represent the C0unt of zero-value OCT co-efficients. This process of run-length encoding gives an excellent compression of the block consisting mostly of zero values.

Further empirical work proved that the length of zero values in a run can be increased to give a further increase in compression by reordering the runs. JPEG came up with ordering the quantized OCT co-efficients in a ZigZag sequence

**Entropy Encoding**

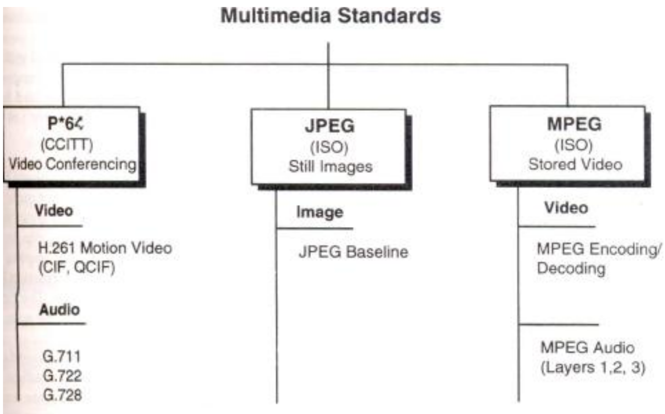
Entropy is a term used in thermodynamics for the study of heat and work. Entropy, as used in data compression, is the measure of the information content of a message in number of bits. It is represented as

**Entropy in number of bits = log2 (probability of Object)**

# VIDEO IMAGE COMPRESSION

The development of digital video technology has made it possible to use digital video compression for a variety of telecommunications applications. Standardization of compression algorithms for video was first initiated by CCITT for teleconferencing and video telephony

Multimedia standards for Video:




**Requirements for full-motion Video Compression**

Applications using MPEG standards can be symmetric or asymmetric. Symmetric applications are applications that require essentially equal use of compression and decompression. Asymmetric applications require frequent decompression.

Symmetric applications require on-line input devices such as video cameras, scanners and microphones for digitized sound. In addition to video and audio compression, this standards activity is concerned with a number of other Issues concerned with playback of video clips and sound clips. The MPEG standard has identified a number of such issues that have been addressed by the standards activity. Let us review these Issues.

**Random Access**

The expectations generated for multimedia systems are the ability to playa sound or video clip from any frame with that clip, irrespective of on what kind-of media the information is stored

**VCR paradigm**

The VCR paradigm consists of the control functions typically found on a VCR such as play, fast forward, rewind, search forward and rewind search.

**Multiplexing Multiple Compressed Audio and Video Bit Streams**

It is a special requirement retrieved from different storage centers on a network. It may be received from different storage centers on a network. It may have to be achieved in a smooth manner to avoid the appearance of a jumpy screen.

**Editability**
**Playback Device Flexibility**
**CCITT H.261 Video Coding Algorithms (P x 64)**

The linear quantizer uses a step algorithm that can be adjusted based on picture quality and coding efficiency. The H.261 is a standard that uses a hybrid of OCT and OPCM (differential pulse Code Modulation) schemes with motion estimation.

It also defines the data format. Each MB contains the OCT coefficients (TCOEFF) ofa block followed by an EOB (a fixed length end-of-block marker). Each MB consists of block data and an MB header. A GOB (Group of Blocks) consists of a GOB header. The picture layer consists of a picture header. The H.261 is designed for dynamic use and provides a fully contained organization and a high level of interactive control.

## Moving Picture Experts Group Compression

The MPEG standards consist of a number of different standards.

The MPEG 2 suite of standards consist of standards for MPEG2 Video, MPEG - 2 Audio and MPEG - 2 systems. It is also defined at different levels, called profiles.

The main profile is designed to cover the largest number of applications. It supports digital video compression in the range of2 to 15 M bits/sec. It also provides a generic solution for television worldwide, including cable, direct broadcast satellite, fibre optic media, and optical storage media (including digital VCRs).

**MPEG Coding Methodology**

The above said requirements can be achieved only by incremental coding of successive frames. It is known as interframe coding. If we access information randomly by frame requires coding confined to a specific frame, then it is known as intraframe coding.

The MPEG standard addresses these two requirements by providing a balance between interframe coding and intraframe coding. The MPEG standard also provides for recursive and non-recursive temporal redundancy reduction.

The MPEG video compression standard provides two basic schemes: discrete-transform-based compression for the reduction of' spatial redundancy and block-based motion compensation for the reduction of temporal (motion) redundancy. During the initial stages of DCT compression, both the full motion MPEG and still image JPEG algorithms are essentially identical. First an image is converted to the YUVcolor space (a luminance/chrominance color space similar to that used for color television). The pixel data is then fed into a discrete cosine transform, which creates a scalar quantization (a two-dimensional array representing various frequency ranges represented in the image) of the pixel data.

Following quantization, a number of compression algorithms are applied, including run-length and Huffman encoding. For full motion video (MPEG I and 2), several more levels of block based motion-compensated techniques are applied to reduce temporal redundancy with both causal and noncausal coding to further reduce spatial redundancy.

The MPEG algorithm for spatial reduction is lossy and is defined as a hybrid which employs motion compensation, forward discrete cosine transform (DCF), a uniform quantizer, and Huffman coding. Block-based motion compensation is *utilized* for *reducing temporal* redundancy (i.e. to reduce the amount of data needed to represent each picture in a video sequence). Motion-compensated reduction is a key feature of MPEG.

**Moving Picture Types**

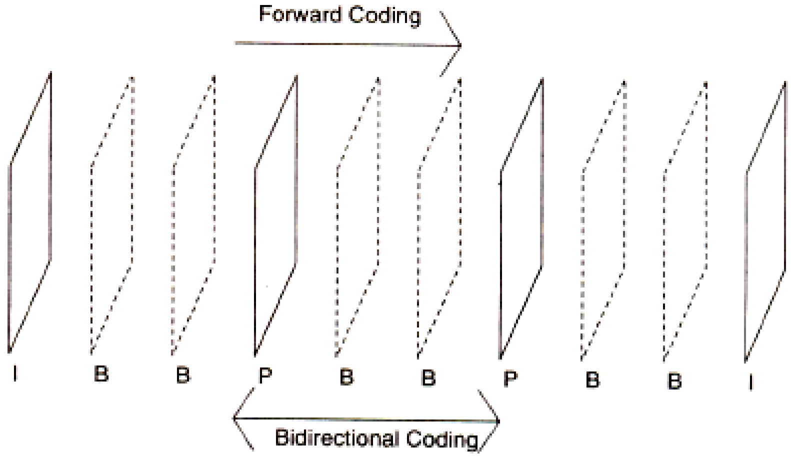
Moving pictures consist of sequences of video pictures or t1'ame'S that are played back a fixed number of frames per second. To achieve the requirement of random access, a set of pictures can be defined to form a group of pictures (GOP) consisting of one or more of the following three types of pictures.

Intra pictures (1)

Undirectionally predicted pictures (U) Bidirectiomflly predicted pictures (B)

A Gap consists of consecutive pictures that begin with an intrapicture. The intrapicture is coded without any reference to any other picture in the group.

PrediGted pictures are coded with a reference to a past picture, either an intrapicture or a unidirectionally predicted picture. Bidirectionally predicted picture is never used as referencesMotion Compensation for Coding MPEG




Let us review the concept of Macroblocks and understand the role they play in compression

**MACRO BLOCKS**

For the video coding algorithm recommended by CCITT, CIF and QCIF are diviqed into a hierarchical block structure consisting of pictures, groups of blocks (GOBs), Macro Blocks(MBs), and blocks. Each picture frame is divided into 16 x 16 blocks. Each Macroblock is composed of four 8 x 8 (Y) luminance blocks and two 8 x 8 ($C_b$ and Cn) chrominance blocks. This set of six blocks, called a macroblock; is the basic hierarchical component used for achieved a high level of compression.

**Motion compensation**

Motion compensation is the basis for most compression algorithms for visual telephony and full-motion video. Motion compensation assumes that the current picture is some translation of a previous picture.

This creates the opportunity for using prediction and interpolation. Prediction requires only the current frame and the reference frame.

Based on motion vectors values generated, the prediction approach attempts to find the relative new position of the object and confirms it by comparing some block exhaustively. In the interpolation approach, the motion vectors are generated in relation to two reference frames, one from the past and the next predicted frame.

The best-matching blocks in both reference frames are searched, and the average is taken as the position of the block in the current frame. The motion vectors for the two reference, frames are averaged.

**Picture Coding Method**

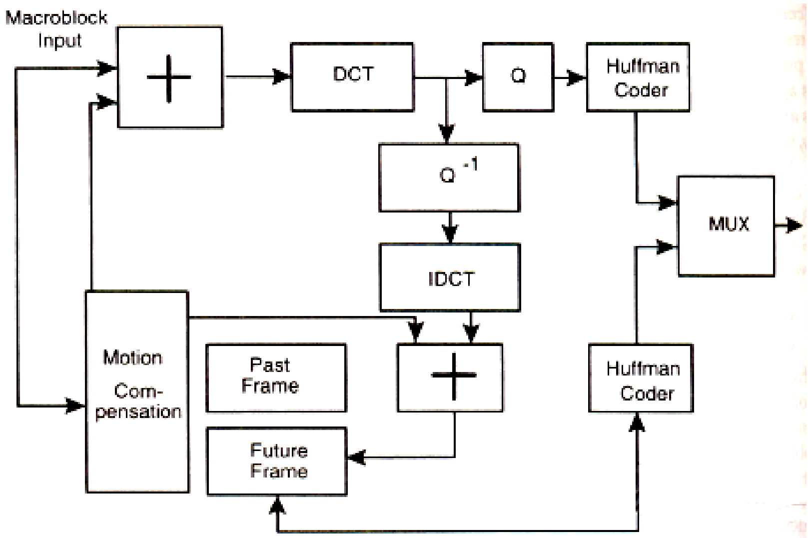
In this coding method, motion compensation is applied bidirectionally. In MPEG terminology, the motion-compensated units are called macro blocks (MBs).

MBs are 16 x 16 blocks that contain a number of 8 x 8 luminance and chrominance blocks. Each 16 x 16 macro block can be of type intrapicture, forward-predicted, backward predicted, or average.

**MPEG Encoder**

**Figure** below shows the architecture of an MPEG encoder. It contains DCT quantizer, Huffman coder and Motion compensation. These represent the key modules in the encoder.

Architecture of MPEG Encoder:




**The Sequence of events for MPEG**

First an image is converted to the YUV color space.
The pixel data is then fed into a DCT, which creates a scalar quantization of the pixel data.
Following quantization, a number of compression algorithms are applied, including run-length and Huffman encoding. For full-motion video, several more levels of motion compensation compression and coding are applied.

**MPEG -2**

It is defined to include current television broadcasting compression and decompression needs, and attempts to include hooks for HDTV broadcasting.
The MPEG-2 Standard Supports:
1.Video Coding: * MPEG-2 profiles and levels.
2.Audio Coding:*MPEG-l audio standard fro backward compatibility.
* Layer-2 audio definitions for MPEG-2 and stereo sound.
* Multichannel sound.
3. Multiplexing: MPEG-2 definitions

**MPEG-2, "The Grand Alliance"**

It consists of following companies AT&T, MIT, Philips, Sarnoff Labs, GI Thomson, and Zenith.
The MPEG-2committee and FCC formed this alliance. These companies together have defined the advanced digital television system that include the US and European HDTV systems. The outline of the advanced digital television system is as follows:
1.Format: 1080/2: 1160 or 720/1.1160
2.Video coding: MPEG-2 main profile and high level
3.Audio coding: Dolby AC3
4.Multiplexor: As defined in MPEG-2
Modulation: 8- VSB for terrestrial and 64-QAM for cable.

**Vector Quantization**

Vector quantization provides a multidimensional representation of information stored in look-up tables, vector quantization is an efficient pattern-matching algorithm in which an image is decomposed into two or more vectors, each representing particular features of the image that are matched to a code book of vectors.
These are coded to indicate the best fit.
In image compression, source samples such as pixels are blocked into vectors so that each vector describes a small segment or sub block of the original image.
The image is then encoded by quantizing each vector separately

**Intel's Indeo Technology**

It is developed by Intel Architecture Labs Indeo Video is a software technology that reduces the size of uncompressed digital video files from five to ten times.
Indeo technology uses multiple types of 'lossy' and 'lossless' compression techniques.

**AUDIO COMPRESSION**

Audio consists of analog signals of varying frequencies. The audio signals are converted to digital form and then processed, stored and transmitted. Schemes such as linear predictive coding and adaptive differential. pulse code modulation (ADPCM) are utilized for compression to achieve 40-80% compression.

**FRACTAL COMPRESSION**

A fractal is a multi-dimensional object with an irregular shape or body that has approximately the same shape or body irrespective of size. For example, if you consider 'stick' as your object, the fractal is defined, mathematically as

$$D = lim \frac{\ln N(L)}{\ln (1/L)}$$

where L - approaches 0,
N(L) ~ number of stick L, and L is the length of the stick.

## 4.2 DATA AND FILE FORMATS STANDARDS

There are large number of formats and standards available for multimedia system.
Let us discuss about the following file formats:
**.:.** Rich-Text Format (RTF)
**.:.** Tagged Image file Format (TIFF)

.:. Resource Image File Format (RIFF)
.:. Musical Instrument Digital Interface (MIDI)
.:. Joint Photographic Experts Group (JPEG)
.:. Audio Video Interleaved (AVI) Indeo file format
.:. TWAIN.

## Rich Text Format

This format extends the range of information from one word processor application or DTP system to another.

The key format information carried across in RTF documents are given below:

Character Set: It determines the characters that supports in a particular implementation.

Font Table: This lists all fonts used. Then, they are mapped to the fonts available in receiving application for displaying text.

Color Table: It lists the colors used in the documents. The color table then mapped for display by receiving application to the nearer set of colors available to that applications.

Document Formatting: Document margins and paragraph indents are specified here.

Section Formatting: Section breaks are specified to define separation of groups of paragraphs.

Paragraph Formatting: It specifies style sheds. It specifies control characters for specifying paragraph justification, tab positions, left, right and first indents relative to document margins, and the spacing between paragraphs.

General Formatting: It includes footnotes, annotations, bookmarks and pictures.

Character Formatting: It includes bold, italic, underline (continuous, dotted or word), strike through, shadow text, outline text, and hidden text.

Special Characters: It includes hyphens, spaces, backslashes, underscore and so on

## TIFF File Format

TIFF is an industry-standard file format designed to represent raster image data generated by scanners, frame grabbers, and paint/ photo retouching applications.

**TIFF Version 6.0 .**

It offers the following formats:

(i) Grayscale, palette color, RGB full-color images and black and white.

(ii) Run-length encoding, uncompressed images and modified Huffman data compression schemes.

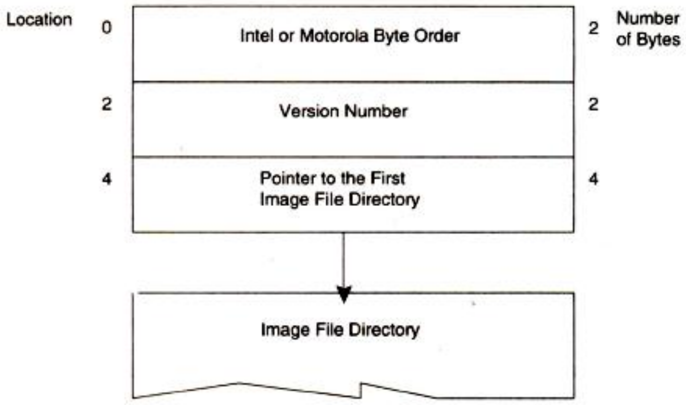
The additional formats are:

(i) Tiled images, compression schemes, images using CMYK, $YCbC_r$ color models.

**TIFF Structure**

TIFF files consists of a header. The header consists of byteordering flag, TIFF file format version number, and a pointer to a table. The pointer points image file directory. This directory contains table of entries of various tags and their information.

TIFF file format Header:




**TIFF Tags**

The first two bytes of each directory entry contain a field called the Tag ID.

Tag IDs arc grouped into several categories. They are Basic, Informational, Facsimile, Document storage and Retrieval.

**TIFF Classes:** (Version 5.0)

It has five classes
1. Class B for binary images
**2.** Class F for Fax
**3.** Class G for gray-scale images
**4.** Class P for palette color images
**5.** Class R for RGB full-color images.

**Resource Interchange File Format (RIFF)**

The **RIFF** file formats consist' of blocks of data called chunks. They are
RIFF Chunk - defines the content of the RIFF file.
List Chunk - allows to embed archival location copy right information and creating date.
Subchunk - allow additional information to a primary chunk
The first chunk in a RIFF file must be a RIFF chunk and it may contain one or more sub chunk
The first four bytes of the RIFF chunk data field are allocated for the form type field containing four characters to identify the format of the data stored in the file: AVI, WAV, RMI, PAL and so

| File type | Form typl | File extension |
|---|---|---|
| Waveform Audio File | WAVE | .WAV |
| Audio Video Interleaved file | AVI | .AVI |
| MIDI File | RMID | .RMI |
| Device Independent Bitmap file | RDIB | .RDI |
| Pallette File | PAL | .PAL |

The sub chunk contains a four-character ASCII string 10 to identify the type of data.
Four bytes of size contains the count of data values, and the data. The data structure of a chunk is same as all other chunks.
**RIFF Chunk**The first 4 characters of the RlFF chunk are reserved for the "RIFF" ASCII string. The next four bytes define the total data size.
The first four characters of the data field are reserved for form tyPe. The rest of the data field contains two subchunk:

(i) fmt ~ defines the recording characteristics of the waveform.
(ii) data ~ contains the data for the waveform.

**LIST Chunk**

*RlFF* chunk may contains one or more list chunks.
List chunks allow embedding additional file information such as archival location, copyright information, creating date, description of the content of the file.

***RlFF* MIDI FILE FORMAT**

*RlFF* MIDI contains a *RlFF* chunk with the form type "RMID"and a subchunk called "data" for MIDI data.
The 4 bytes are for ID of the *RlFF* chunk. 4 bytes are for size 4 bytes are for form type
4 bytes are for ID of the subchunk data and 4 bytes are for the size of MIDI data.

**RIFF DIBS (Device-Independent Bit Maps)** .

DIB is a Microsoft windows standard format. It defines bit maps and color attributes for bit maps independent of devices. DIEs are normally embedded in .BMP files, .WMF meta data files, and .CLP files.

**DIB Structure**

| | | |
|---|---|---|
| BITMAPINFOHEADER | RGBQUAD | PIXELS |

BIT MAP INFOHEADER is the bit map information header.
RGBEQUAD is the color table structure.
PIXELs are the array of bytes for the pixel bit map.
The following shows the DIE file format

| | | |
|---|---|---|
| BITMAPINFOHEADER | BITMAPINFO=BITMAPINF O | PIXELS |

| | HEADER + RGBQUAD | |
|---|---|---|

A RIFF DIB file format contains a RIFF chunk with the Form Type "RDIB" and a subchunk called "data" for DIB data.
4 bytes denote ID of the RIFF chunk
4 bytes refer size ofXYZ.RDI     4 bytes define Forum Type
4 bytes describe ID of the sub chunk data 4 bytes define size of DIB data.

**RIFF PALETTE  File format**

The RIFF Palette file format contains a RIFF chunk with the Form Type "RP AL" and a subchunk called "data" for palette data. The Microsoft Windows logical palette structure is enveloped in the RIFF data subchunk. The palette structure contains the palette version number, number of palette entries, the intensity of red, green and blue colours, and flags for the palette usage. The palette structure is described by the following code segment:

typedef struct tagLOGP ALETTE {
WORD palVersion;    *IIWindows* version number for the structure
*I* !Number of.palettes color entries
PALETIEENTRY palpalEntry []; *llarray* of PALEN TRY data } LOGPALETTE; structure
the form type "AVI" and two mandatory list chunks, "hdr 1" and "n10vi".
The "hdr 1" defines the format of the data "Movi" contains the data for the audio-video streams. The third list chunk called "id xl", is an optional index chunk.

**Boundary condition Handling for AVI files**

Each audio and video stream is grouped together to form a ree chunk. If the size of a rec chunk is not a multiple of2048 bytes, then the rec chunk is padded to make the size of each rec chunk a multiple of 2048 bytes. To align data on a 2048 byte boundary, dummy data is added by a "JUNK" data chunk. The JUNK chunk is a standard RIFF chunk with a 4 character identifier, "JUNK," followed by the dummy data.

**MIDI File Format**

The MIDI file format follows music recording metaphor to provide the means of storing separate tracks of music for each instrument so that they can be read and syn~hronized when they are played.
The MIDI file format also contains chunks (i.e., blocks) of data. There are two types of chunks: (i) header chunks (ii) track chunks.

**Header Chunk**

It is made up of 14 bytes .

- The first four-character string is the identifier string, "MThd" .
- The second four bytes contain the data size for the header chunk. It is set to a fixed value of six bytes .
- The last six bytes contain data for header chunk.

**Track chunk**

The Track chunk is organized as follows:

.:. The first 4-character string is the identifier.
.:. The second 4 bytes contain track length.

**MIDI Communication Protocol**

This protocol uses 2 or more bytes messages.
The number of bytes depends on the types of message. There are two types of messages:
(i) Channel messages and (ii) System messages.

**Channel Messages**

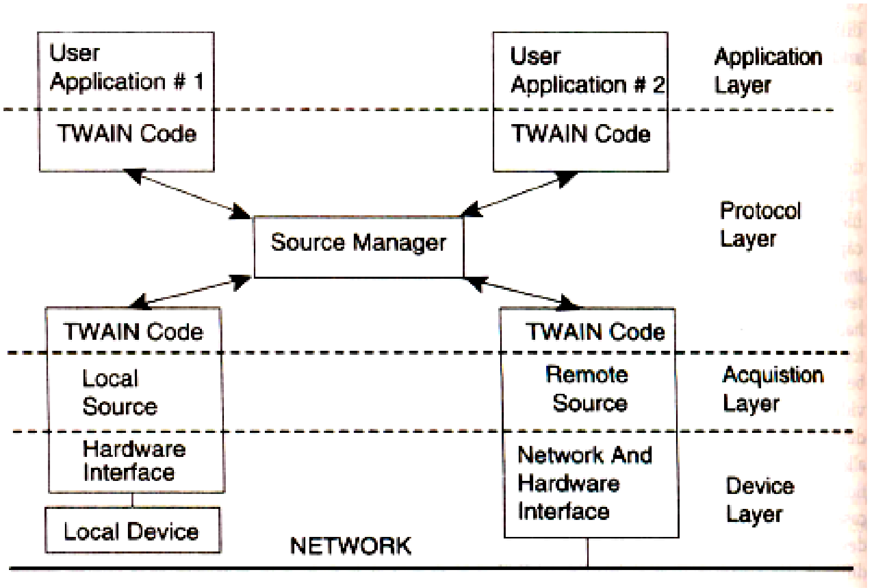
A channel message can have up to three bytes in a message. The first byte is called a status byte, and other two bytes are called data bytes. The channel number, which addresses one of the 16 channels, is encoded by the lower nibble of the status byte. Each MIDI voice has a channel number; and messages are sent to the channel whose channel number matches the channel number encoded in the lower nibble of the status byte. There are two types of channel messages: voice messages and the mode messages.

**Voice messages**

Voice messages are used to control the voice of the instrument (or device); that is, switch the notes on or off and sent key pressure messages indicating that the key is depressed, and send control messages to control effects like vibrato, sustain, and tremolo. Pitch wheel messages are used to change the pitch of all notes .

**Mode messages**
Mode messages are used for assigning voice relationships for up to 16 channels; that *is*, to set the device to MOWO mode or POLY mode. Omny Mode on enables the device to receive voice messages on all channels.
**System Messages**
System messages apply to the complete system rather than specific channels and do not contain any channel numbers. There are three types of system messages: common messages, real-time messages, and exclusive messages. In the following, we will see how these messages are used.
**Common Messages** These messages are common to the complete system. These messages provide for functions such as select a song, setting the song position pointer with number of beats, and sending a tune request to an analog synthesizer.
**System Real Time Messages**
These messages are used for setting the system's real-time parameters. These parameters include the timing clock, starting and stopping the sequencer, ressuming the sequencer from a stopped position, and resetting the system.
**System Exclusive messages**
These messages contain manufacturer-specific data such as identification, serial number, model number, and other information. Here, a standard file format is generated which can be moved across platforms and applications.
**JPEG Motion Image:**
JPEG Motion image will be embedded in A VI RIFF file format.
There are two standards available:
(i) MPEG ~ In this, patent and copyright issues are there.
(ii) MPEG 2 ~ It provide better resolution and picture quality.

## TWAIN
To address the problem of custom interfaces, the TWAIN working group was formed to define an open industry standard interface for input devices. They designed a standard interface called a generic TW AIN . interface. It allows applications to interface scanners, digital still cameras, video cameras.
**TWAIN ARCHITECHTURE:**




- The Twain architecture defines a set of application programming interfaces (APls) and a protocol to acquire data from input devices.
- It is a layered architecture.
- It has application layer, the protocol layer, the acquisition layer and device layer.
- Application Layer: This layer sets up a logical connection with a device. The application layer interfaces with protocol layer.
- Protocol Layer: This layer is responsible for communications between the application and acquisition layers.
- The main part of the protocol layer is the source Manager.

- Source manager manages all sessions between an application and the sources, and monitors data acquisition transactions. The protocol layer is a complex layer.

It provides the important aspects of device and application interfacing functions.

**The Acquisition Layer:** It contains the virtual device driver.

It interacts directly with the device driver. This layer is also known as source.

It performs the following functions:

1. Control of the device.
2. Acquisition of data from the device.
3. Transfer of data in agreed format.
4. Provision of user interface to control the device.

**The Device Layer:** The device layer receives software commands and controls the device hardware.

**NEW WAVE RIFF File Format:** This format contains two subchunks:

**(i)** Fmt **(ii)** Data.

It may contain optional subchunks:

**(i)** Fact
**(ii)** Cue points
**(iii)** Play list
**(iv)** Associated datalist.

**Fact Chunk:** It stores file-dependent information about the contents of the WAVE file. **Cue Points Chunk:** It identifies a series of positions in the waveform data stream. **Playlist Chunk:** It specifies a play order for series of cue points. **Associated Data Chunk:** It provides the ability to attach information, such as labels, to sections of the waveform data stream. **Inst Chunk:** The file format stores sampled sound synthesizer's samples.

## **4.3** MULTIMEDIA INPUT/OUTPUT TECHNOLOGIES

**Multimedia Input and Output Devices**

Wide ranges of Input and output devices are available for multimedia.

**Image Scanners:** Image scanners are the scanners by which documents or a manufactured part are scanned. The scanner acts as the camera eye and take a photograph of the document, creating an unaltered electronic pixel representation of the original.

**Sound and Voice:** When voice or music is captured by a microphone, it generates an electrical signal. This electrical signal has analog sinusoidal waveforms. To digitize, this signal is converted into digital voice using an analog-to-digital converter.

**Full-Motion Video:** It is the most important and most complex component of Multimedia System. Video Cameras are the primary source of input for full-motion video.

. **Pen Driver**: It is a pen device driver that interacts with the digitizer to receive all digitized information about the pen location and builds pen packets for the recognition context manager. Recognition context manager: It is the main part of the pen system. It is responsible for co-ordinating windows pen applications with the pen. It works with Recognizer, dictionary, and display driver to recognize and display pen drawn objects.

**Recognizor**: It recognizes hand written characters and converts them to ASCII.

**Dictionary**: A dictionary is a dynamic link library (DLL); The windows form pen computing system uses this dictionary to validate the recognition results.

**Display Driver**: It interacts with the graphics device interface' and display hardware.

When a user starts writing or drawing, the display driver paints the ink trace on the screen.

**Video and Image Display Systems Display System Technologies**

There are variety of display system technologies employed for decoding compressed data for displaying.

Mixing and scaling technology: For VGA screen, these technologies are used.

**VGA mixing**: Images from multiple sources are mixed in the image acquisition memory.

**VGA mixing with scaling**: Scalar ICs are used to sizing and positioning of images in predefined windows.

**Dual buffered VGA mixing/Scaling**: If we provide dual buffering, the original image is prevented from loss. In this technology, a separate buffer is used to maintain the original image.

## Visual Display Technology Standards

**MDA: Monochrome Display Adapter.**

It was introduced by IBM .

•:. It displays 80 x 25 rows and columns .
•:. It could not display bitmap graphics .
•:. It was introduced in 1981.
**CGA: Color Graphics Adapter .**
•:. It was introduced in 1981.
.:. It was designed to display both text and bitmap graphicsi
•it supported RGB color display,
•.:. It could display text at a resolution of 640 x 200 pixels .
•:. It displays both 40 x 25 and 80 x 25 row!' and columns of text characters.
**MGA: Monochrome Gr.aphics Adapter .**
•:. It was introduced in 1982 .
•:. It could display both text and graphics .
•:. It could display at a resolution 720 x 350 for text and
720 x 338 for Graphics . MDA is compatible mode for this standard.
**EGA: Enhanced Graphics Adapter** .
•:. It was introduced in 1984 .
•:. It emulated both *MDt.* and CGA standards .
•:. It allowed the display of both text and graphics in 16 colors at
a resolution of 640 x· 350 pixels.
**PGA: Professional Graphics Adapter.**
.:. It was introduced in 1985 .
•:. It could display bit map graphics at 640 x 480 resolution and 256 colors .
• :. Compatible mode of this standard is CGA.
**VGA: Video Graphics Array .** :. It was introduced by IBM in 1988 .
•:. It offers CGA and EGA compatibility .
•:. It display both text and graphics .
•:. It generates analog RGB signals to display 256 colors .
•:. It remains the basic standard for most video display systems.
**SVGA: Super Video Graphics Adapter.** It is developed by VESA (Video Electronics Standard Association) . It's goal is to display with higher resolution than the VGA
with higher refresh rates with minimize flicker.
**XGA: Extended Graphics Array**
It is developed by IBM . It offers VGA compatible mode . Resolution of 1024 x 768 pixels in 256 colors is offered by it. XGA utilizes an interlace scheme for refresh rates.

## Flat Panel Display system

Flat panel displays use a fluorescent tube for backlighting to give the display a sufficient level of brightness. The four basic technologies used for flat panel display are:

1. Passive-matrix monochrome
2. Active-matrix monochrome
3. Passive-matrix color
4. Active-matrix color.

## LCD (Liquid Crystal Display)

Construction: Two glass plates each containing a light polarizer at right angles to the other plate, sandwich the nematic (thread like) liquid crystal material.
Liquid crystal is the compounds having a crystalline arrangement of molecules. But it flow like a liquId.
Nematic liquid crystal compounds are tend to keep the long axes of rod-shaped molecules aligned.
Rows of horizontal transparent conductors are built into one glass plate, and columns of vertical conductors are put into the other plate. The intersection of two conductors defines a pixel position.

## Passive Matrix LCD

**Working:** Normally, the molecules are aligned in the 'ON' state.
Polarized light passing through the materials is twisted so that it will pass through the opposite polarizer. The light is then reflected back to the viewer. To turn off the pixel, we have to apply a voltage to the two intersecting conductors to align molecules so that the light is not twisted.

## ACTIVE Matrix LCD

In this device, a transistor is placed at each pixel position, using thin-film transisor technology.
The transistors are used to control the voltage at pixel locations and to prevent charge from gradually leaking out of the liquid crystal cells.

## PRINT OUTPUT TECHNOLOGIES

There are various printing technologies available namely Dot matrix, inkjet, laser print server and ink jet color. But, laser printing technology is the most common for multimedia systems.
To explain this technology, let us take Hewlett Packard Laser jet-III laser printer as an example. The basic components of the laser printer are
**.:.** Paper feed mechanism **.:.** Paper guide **.:.** Laser assembly **.:.** Fuser **.:.** Toner cartridge.
**Working:** The paper feed mechanism moves the paper from a paper tray through the paper path in the printer. The paper passes over a set of corona wires that induce a change in the paper .

- The charged paper passes over a drum coated with fine-grain carbon (toner), and the toner attaches itself to the paper as a thin film of carbon .The paper is then struck by a scanning laser beam that follows the pattern of the text on graphics to be printed . The carbon particles attach themselves to the pixels traced by the laser beam . The fuser assembly then binds the carbon particles to the paper.

**Role of Software in the printing mechanism:**

The software package sends information to the printer to select and control printing features .

Printer drivers (files) are controlling the actual operation of the printer and allow the application software to access the features ofthe printer.

## IMAGE SCANNERS

In a document imaging system, documents are scanned using a scanner. \The document being scanned is placed on the scanner bed or fed into the sheet feeder of the scanner .The scanner acts as the camera eye and takes a photograph of the document, creating an image of the original. The pixel representation (image) is recreated by the display software to render the image of the original document on screen or to print a copy of it.

**Types of Scanners**

A and B size Scanners, large form factor scanners, flat bed scanners, Rotory drum scanners and hand held scanners are the examples of scanners.
**Charge-Coupled Devices** All scanners use charge-coupled devices as their photosensors. CCDs consists of cells arranged in a fixed array on a small square or rectangular solid state surface. Light source moves across a document. The intensity of the light reflected by the mirror charges those cells. The amount of charge is depending upon intensity of the reflected light, which depends on the pixel shade in the document.

## Image Enhancement Techniques

**HalfTones** In a half-tone process, patterns of dots used to build .scanned or printed image create the illusion of continuous shades of gray or continuous shades of color. Hence only limited number of shades are created. This process is implemented in news paper printers.
But in black and white photograph or color photograph, almost infinite levels of tones are used.

**Dithering**

Dithering is a process in which group of pixels in different patterns are used to approximate halftone patterns by the scanners. It is used in scanning original black and white photographs.
Image enhancement techniques includes controls of brightness, deskew (Automatically corrects page alignment), contrast, sharpening, emphasis and cleaning up blacknoise dots by software.

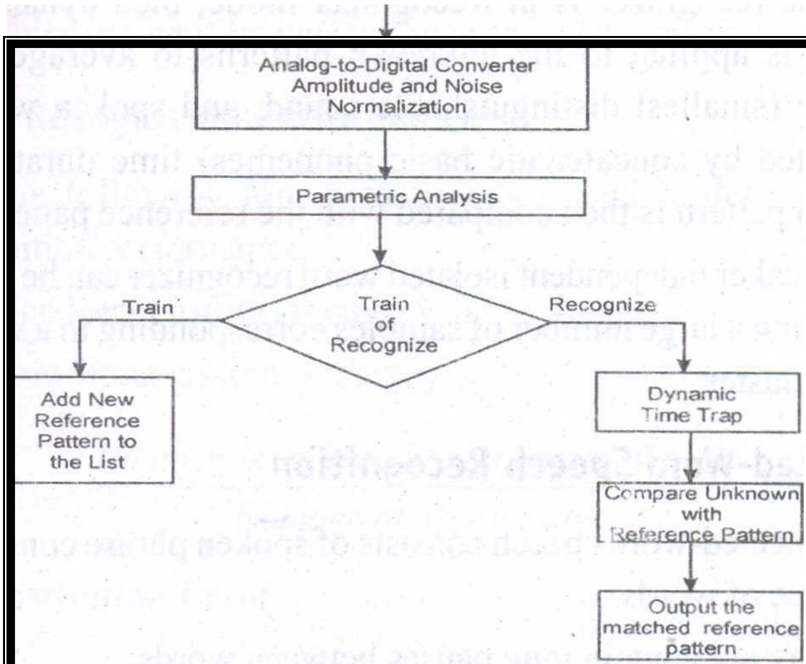
**Image Manipulation**

It includes scaling, cropping and rotation.
**Scaling:** Scaling can be up or down, the scaling software is available to reduce or enlarge. This software uses algorithms.
**Cropping:** To remove some parts of the image and to put the rest of the image as the subset of the old image.
**Rotation:** Image could be rotated at any degree for displaying it in different angles.

**4.4 DIGITAL VOICE AND AUDIO**

**Digital Audio**

Sound is made up of continuous analog sine waves that tend to repeat depending on the music or voice. The analog waveforms are converted into digital fornlat by analog-to-digital converter (ADC) using sampling process.




**Sampling process**
Sampling is a process where the analog signal is sampled over time at regular intervals to obtain the amplitude of the analog signal at the sampling time.
**Sampling rate**
The regular interval at which the sampling occurs is called the sampling rate.
**Digital Voice**
Speech is analog in nature and is cOl1veli to digital form by an analog-to-digital converter (ADC). An ADC takes an input signal from a microphone and converts the amplitude of the sampled analog signal to an 8, 16 or 32 bit digital value.
The four important factors governing the ADC process are

- **sampling rate**
- **resolution**
- **linearity and**
- **conversion speed.**

**Sampling Rate:** The rate at which the ADC takes a sample of an analog signal.
**Resolution:** The number of bits utilized for conversion determines the resolution of ADC.
**Linearity:** Linearity implies that the sampling is linear at all frequencies and that the amplitude tmly represents the signal.
**Conversion Speed:** It is a speed of ADC to convert the analog signal into Digital signals. It must be fast enough.

## VOICE Recognition System

Voice Recognition Systems can be classified into three types.
1.Isolated-word Speech Recognition.
2.Connected-word Speech Recognition.
3.Continuous Speech Recognition.

1. **Isolated-word Speech Recognition.**

It provides recognition of a single word at a time. The user must separate every word by a pause. The pause marks the end of one word and the beginning of the next word.
**Stage 1: Normalization**
The recognizer's first task is to carry out amplitude and noise normalization to minimize the variation in speech due to ambient noise, the speaker's voice, the speaker's distance from and position relative to the microphone, and the speaker's breath noise.
**Stage2: Parametric Analysis**
It is a preprocessing stage that extracts relevent time-varying sequences of speech parameters. This stage serves two purposes: (i) It extracts time-varying speech parameters. (ii) It reduces the amount of data of extracting the relevant speech parameters.
**Training mode**In training mode of the recognizer, the new frames are added to the reference list.
**Recognizer mode**If the recognizer is in Recognizer mode, then dynamic time warping is applied to the unknown patterns to average out the phoneme (smallest distinguishable sound, and spoken words are constructed by concatenatic basic phonemes) time duration. The unknown pattern is then compared with the reference patterns.
A speaker independent isolated word recognizer can be achieved by groupi.ng a large number of samples corresponding to a word into a single cluster.
**2Connected-Word Speech Recognition**Connected-word speech consists of spoken phrase consisting of a sequence of words. It may not contain long pauses between words.

**The method using Word Spotting technique**
It Recognizes words in a connected-word phrase. In this technique, Recognition is carried out by compensating for rate of speech variations by the process called dynamic time warping (this process is used to expand or compress the time duration of the word), and sliding the adjusted connected-word phrase representation in time past a stored word template for a likely match.
**Continuous Speech Recognition**
This sytem can be divided into three sections:

(i) A section consisting of digitization, amplitude normalization, time nonnalization and parametric representation.

(ii) Second section consisting of segmentation and labeling of the speech segment into a symbolic string based on a knowledgebased or rule-based systems.

(iii) The final section is to match speech segments to recognize word sequences.

**Voice Recognition performance**
It is categorized into two measures: Voice recognition performance and system performance.
The following four measures are used to determine voice recognition performance.

1. **Voice Recognition Accuracy**

Voice Recognition Accuracy = $\frac{\textit{Number of correctly recognized words}}{\textit{Number of test words}}$ x l00

2. **Substitution Error**

Substitution error = $\frac{\textit{Number of substituted words}}{\textit{Number of test words}}$ x l00

3. **No Response Error**

$\frac{\textit{Number of no responses}}{\textit{Number of test words}}$ x 100

4. **Insertion Error**

Insertion error = $\frac{\textit{Number of insertion error}}{\textit{Number of test words}}$ x 100

**Voice Recognition Applications**
Voice mail integration: The voice-mail message can be integrated with e-mail messages to create an integrated message.
**DataBase Input and Query Applications**
A number of applications are developed around the voice recognition and voice synthesis function.
The following lists a few applications which use Voice recognition.

- Application such as order entry and tracking

It is a server function; It is centralized; Remote users can dial into the system to enter an order or to track the order by making a Voice query.

- Voice-activated rolodex or address book

When a user speaks the name of the person, the rolodex application searches the name and address and voice-synthesizes the name, address, telephone numbers and fax numbers of a selected person. In medical emergency, ambulance technicians can dial in and register patients by speaking into the hospital's centralized system.

- Police can make a voice query through central data base to take follow-up action ifhe catch any suspect.
- Language-teaching systems are an obvious use for this technology. The system can ask the student to spell or speak a word. When the student speaks or spells the word, the systems performs voice recognition and measures the student's ability to spell. Based on the student's ability, the system can adjust the level of the course. This creates a self-adjustable learning system to follow the individual's pace.
- Foreign language learning is another good application where"' an individual student can input words and sentences in the system. The system can then correct for pronunciation or grammar.

## Musical Instrument Digital Interface (MIDI)
MIDI interface is developed by Daver Smith of sequential circuits, inc in 1982. It is an universal synthesizer interface
**MIDI Specification 1.0**

MIDI is a system specification consisting of both hardware and software ~omponents which define inter-coimectivity and a communication protocol for electronic sysnthesizers, sequences, rythm machines, personal computers, and other electronic musical instruments. The inter-connectivity defines the standard cabling scheme, connector type and input/output circuitry which enable these different MIDI instruments to be interconnected. The communication protocol defines standard multibyte messages that allow controlling the instrument"s voice and messages including to send response, to send status and to send exclusive.

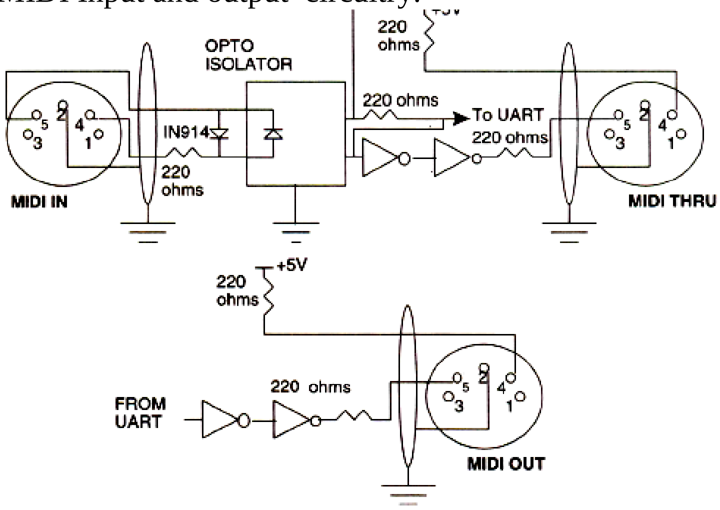
**MIDI Hardware Specification**

The MIDI. hardware specification require five pin panel mount requires five pin panel mount receptacle DIN connectors for MIDI IN, MIDI OUT and MIDI THRU signals. The MIDI IN connector is for input signals The MIDI OUT is for output signals MIDI THRU connector is for daisy-chaining multiple MIDI instruments.

**MIDI Interconnections**

The MIDI IN port of an instrument receives MIDI ncssages to play the instrument's internal synthesizer. The MIDI OUT port sends MIDI messages to play these messages to an external synthesizer. The MIDI THRU port outputs MIDI messages received by the MIDI IN port for daisy-chaining external synthesizers.

MIDI Input and output circuitry:




**Communication Protocol**

The MIDI communication protocol uses multibyte messages; There are two types of messages:

(i) **Channel messages**

**(ii) System messages**.

The channel message have three bytes. The first byte is called a status byte, and the other two bytes are called data bytes.

The two types of channel messages:

(**i) Voice messages**

**(ii) Mode messages**.

System messages: The three types of system messages.

Common message: These messages are common to the complete system. These messages provide for functions.

System real.time messages: These messages are used for setting the system's real-time parameters. These parameters include the timing clock, starting and stopping the sequencer, resuming the sequencer from a stopped position and restarting the system.

System exclusive message: These messages contain manufacturer specific data such as identification, serial number, model number and other information.

**SOUND BOARD ARCHITECTURE**A sound card consist of the following components:

MIDI Input/Output Circuitry, MIDI Synthesizer Chip, input mixture circuitry to mix CD audio input with

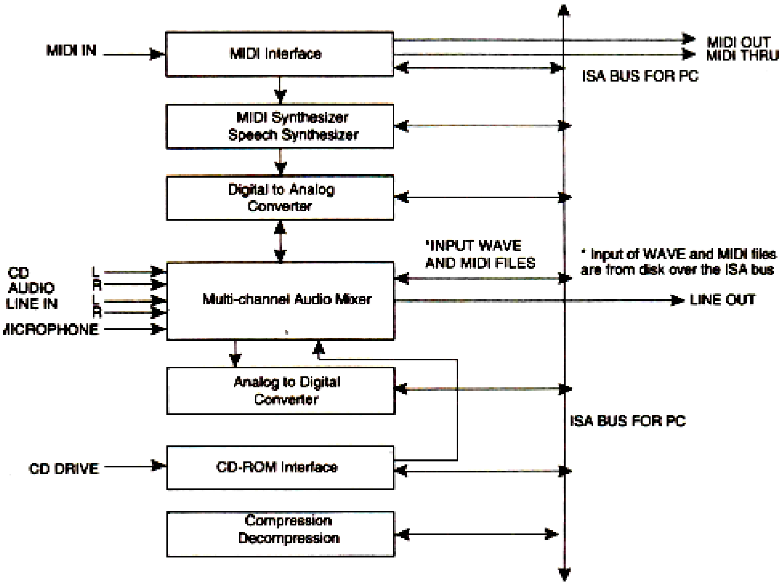
LINE IN input and microphone input, analog-to-digital converter with a pulse code modulation circuit to convert analog signals to digital to create WAVfiles, a decompression and compression chip to compress and decompress audio files, a speech synthesizer to synthesize speech output, a speech recognition circuitry to recognize speech input and output circuitry to output stereo audio OUT or LINEOUT.

**AUDIO MIXER**

The audio mixer c:omponent of the sound card typically has external inputs for stereo CD audio, stereo LINE IN, and stereo microphone MICIN.

These are analog inputs, and they go through analog-to-digitaf conversion in conjunction with PCM or ADPCM to generate digitized samples.

**SOUND BOARD ARCHITECTURE:**




**Analog-to-Digital Converters**: The ADC gets its input from the audio mixer and converts the amplitude of a sampled analog signal to either an 8-bit or 16-bit digital value.

**Digital-to-Analog Converter** (DAC): A DAC converts digital input in the 'foml of W AVE files, MIDI output and CD audio to analog output signals.

Sound Compression and Decompression: Most sound boards include a codec for sound compression and decompression.

ADPCM for windows provides algorithms for sound compression.

**CD-ROM Interface**: The CD-ROM interface allows connecting u CD ROM drive.to the sound board.

## VIDEO IMAGES AND ANIMATION

## VIDEO FRAME GRABBER ARCHITECTURE

A video frame grabber is used to capture, manipulate and enhance video images.

A video frame grabber card consists of video channel multiplexer, Video ADC, Input look-up table with arithmetic logic unit, image frame buffer, compression-decompression circuitry, output color look-up table, video DAC and synchronizing circuitry.

**Video Channel Multiplexer:**

A video channel multiplexer has multiple inputs for different video inputs. The video channel multiplexer allows the video channel to be selected under program control and switches to the control circuitry appropriate for the selected channel in aTV with multi – system inputs.

Analog to Digital Converter: The ADC takes inputs from video multiplexer and converts the amplitude of a sampled analog signal to either an 8-bit digital value for monochrome or a 24 bit digital value for colour.

**Input lookup table**: The input lookup table along with the arithmetic logic unit (ALU) allows performing image processing functions on a pixel basis and an image frame basis. The pixel image-processing functions ate histogram stretching or histogram shrinking for image brightness and contrast, and histogram sliding to brighten or darken the image. The frame-basis image-processing functions perform logical and arithmetic operations.

**Image Frame Buffer Memory**: The image frame buffer is organized as a l024 x 1024 x 24 storage buffer to store image for image processing and display.

**Video Compression-Decompression**: The video compressiondecompression processor is used to compress and decompress still image data and video data.

**Frame Buffer Output Lookup Table**: The frame buffer data represents the pixel data and is used to index into the output look uptable. The output lookup table generates either an 8 bit pixel value for monochrome or a 24 bit pixel value for color.

**SVGA Interface:** This is an optional interface for the frame grabber. The frame grabber can be designed to include an SVGA frame buffer with its own output lookup table and digital-to-analog converter.

**Analog Output Mixer**: The output from the SVGA DAC and the output from image frame buffer DAC is mixed to generate overlay output signals. The primary components involved include the display image frame buffer and the display SVGA buffer. The display SVGA frame buffer is overlaid on the image frame buffer or live video, This allows SVGA to display live video.

**Video and Still Image Processing**

Video image processing is defined as the process of manipulating a bit map image so that the image can be enhanced, restored, distorted, or analyzed.

Let us discuss about some of the terms using in video and still image processing.

**Pixel point to point processing:** In pixel point-to-point processing, operations are carried out on individual pixels one at a time.

**Histogram Sliding:** It is used to change the overall visible effect of brightening or darkening of the image. Histogram sliding is implemented by modifying the input look-up table values and using the input lookup table in conjunction with arithmetic logic unit.

**Histogram Stretching and Shrinking:** It is to increase or decrease the contrast.

In histogram shrinking, the brighter pixels are made less bright and the darker pixels are made less dark.

**Pixel Threshold:** Setting pixel threshold levels set a limit on the bright or dark areas of a picture. Pixel threshold setting is also achieved through the input lookup table.

**Inter- frame image processing**

Inter- frame image processing is the same as point-to-point image processing, except that the image processor operates on two images at the same time. The equation of the image operations is as follows:

Pixel output (x, y) = (Image l(x, y)

Operator (Image 2(x, y)

Image Averaging: Image averaging minimizes or cancels the effects of random noise.

Image Subtraction: Image subtraction is used to determine the change from one frame to the next .for image comparisons for key frame detection or motion detection.

**Logical Image Operation:** Logical image processing operations are useful for comparing image frames and masking a block in an image frame.

**Spatial Filter Processing** The rate of change of shades of gray or colors is called spatial frequency. The process of generating images with either low-spatial frequency-components or high frequency components is called spatial filter processing.

**Low Pass Filter:** A low pass filter causes blurring of the image and appears to cause a reduction in noise.

**High Pass Filter:** The high-pass filter causes edges to be emphasized. The high-pass filter attenuates low-spatial frequency components, thereby enhancing edges and sharpening the image.

**Laplacian Filter:** This filter sharply attenuates low-spatial-frequency components without affecting and high-spatial frequency components, thereby enhancing edges sharply.

**Frame Processing** Frame processing operations are most commonly for geometric operations, image transformation, and image data compression and decompression Frame processing operations are very compute intensive many multiply and add operations, similar to spatial filter convolution operations.

**Image scaling:** Image scaling allows enlarging or shrinking the whole or part of an image.

**Image rotation**: Image rotation allows the image to be rotated about a center point. The operation can be used to rotate the image orthogonally to reorient the image if it was scanned incorrectly. The operation can also be used for animation. The rotation formula is:

pixel output-(x, *y)* = pixel input *(x,* cos Q + *y* sin Q, - *x* sin Q + Y cos Q)

where, Q is the orientation angle

*x,* yare the spatial co-ordinates of the original pixel.

**Image translation**: Image translation allows the image to be moved up and down or side to side. Again, this function can be used for animation.

**The translation formula is:**

Pixel output *(x, y)* =Pixel Input *(x + Tx, y + Ty)* where

*Tx* and *Ty* are the horizontal and vertical coordinates. *x*, yare the spatial coordinates of the original pixel.
**Image transformation:** An image contains varying degrees of brightness or colors defined by the spatial frequency. The image can be transformed from spatial domain to the frequency domain by using frequency transform.

## Image Animation Techniques

**Animation:** Animation is an illusion of movement created by sequentially playing still image frames at the rate of 15-20 frames per second.
**Toggling between image frames:** We can create simple animation by changing images at display time. The simplest way is to toggle between two different images. This approach is good to indicate a "Yes" or "No" type situation.
**Rotating through several image frames:** The animation contains several frames displayed in a loop. Since the animation consists of individual frames, the playback can be paused and resumed at any time.

## 4.5 FULL MOTION VIDEO

Most modem cameras use a CCD for capturing the image. HDTV video cameras will be all-digital, and the capture method will be significantly different based on the new NTSC HDTV Standard.

**Full-Motion Video Controller Requirements**

**Video Capture Board Architecture:** A full-motion video capture board is a circuit card in the computer that consists of the following components:
(i) Video input to accept video input signals.
(ii) S- Video input to accept RS 170 input.
(iii) Video compression-decompression processor to handle different video compression-decompression algorithms for video data.
(iv) Audio compression-decompression processor to compress and decompress audio data.
(v) Analog to digital converter.
(vi) Digital to analog converter.
(vii) Audio input for stereo audio LINE IN, CD IN. (viii) Microphone.
A video capture board can handle a variety of different audio and video input signals and convert them from analog to digital or digital to analog.
**Video Channel Multiplexer:** It is similar to the video grabber's video channel multiplexer.
**Video Compression and Decompression:** A video compression and decompression processor is used to compress and decompress video data.
The video compression and decompression processor contains multiple stages for compression and decompression. The stages include forward discrete cosine transformation and inverse discrete cosine transformation, quantization and inverse quantization, ZigZag and Zero run-length encoding and decoding, and motion estimation and compensation.
**Audio Compression:** MPEG-2 uses adaptive pulse code modulation (ADPCM) to sample the audio signal. The method takes a difference between the actual sample value and predicted sample value. The difference is then encoded by a 4-bit value or 8-bit value depending upon the sample rate
**Analog to Digital Converter**: The ADC takes inputs from the video switch and converts the amplitude of a sampled analog signal to either an 8-bit or 16-bit digital value.

**4.6 STORAGE AND RETRIVAL TECHNOLOGY**

Multimedia systems require storage for large capacity objects such as video, audio and images.
Another requirement is delivery of audio and video objects. Storage technologies include battery powered RAM, Nonvolatile flash, rotating magnetic disk drives, and rotating optical disk drives: Let us discuss these technologies in detail.

**MAGNETIC MEDIA TECHNOLOGY**

Magnetic hard disk drive storage is a mass storage medium.
It has advantages of it continual reduction in the price per mega byte of high-capacity storage.
It has high capacity and available in low cost.
In this section let us concentrate on magnetic disk I/O subsystems most applicable to multimedia uses.

**HARD DISK TECHNOLOGY**

Magnetic hard disk storage remains a much faster mass storage to play an important rol~ in multimedia systems.
It remains a much faster mass storage medium than any other mass storage medium.

**ST506 and MFM Hard drives**: ST506 is an interface that defines the signals and the operation of signals between a hard disk controller and the hard disk. It is developed by seagate. It is used to control platter speed and the movement of heads for a drive. Parallel data is converted to a series of encoded pulses by using a scheme called MFM (modified frequency modulation). The MFM encoding scheme offers greater packing of bits and accuracy than the FM encoding scheme. Other encoding scheme is Run-Length-Limited. Its drive capacity varies from 20 M Bytes to 200 M Bytes.

**ESDI Hard Drive**: ESDI (Enhanced Small Device Interface) was developed by a consortium of several manufacturers. It converts the data into serial bit streams.

It uses the Run-Length-Limited Scheme for encoding. The drive has data separator circuitry Drive capacity varies from 80 M Bytes to 2 GB. ESDI interface has two ribbon cables: (i) 36 pin cable for control signals. (ii) 20 pin cable for data signals.

**IDE**: Integrated Device Electronics (IDE) contains a,n integrated controller with drive.

The interface is 16 bit parallel data interface. The IDE interface supports two IDE drives. One is master drive and other is slave drive. Here, Jumper setting is required. The transfer rate is 8 MHz at bus speed.

**New Enhanced IDE Interface**

This new interface has a transfer rate of 9-13 M Bytes/See with maximum capacity around 8 GB. It supports upto four drives CD ROM and tape drives.

**SCSI** (Small Computer System Interface)

It is an ANSI X3T9.2 standard which supports SCSI and SCSI2 Standards. The Standard defines both software and hardware.

**SCSI-I**:It defines an 8-bit parallel data path between host adapter and device.

Here, host adapter is known as initiator and the device is known as target. There are one initiator and seven targets.

Nine control signals define the activity phases of the SCSI bus during a transaction between an initiator and a target. The phases are:

(i) arbitration phase (ii) selection phase (iii) command phase (iv) data phase (v) status phase
(vi) message phase (vii) bus free phase.

**Arbitrary Phase**: In this phase an initiator starts arbitration and tries to acquire the bus.

**Selection Phase**: In this phase, an initiator has acquired the bus and selects the target to which it needs to communicate.

**Command Phase**: The target now enters into this phase. It requests a command from the initiator. Initiator places a command on the bus. It is accepted by the target.

**Data Phase**: The target now enters in this phase. It requests data transfer with the initiator. The data is placed on the bus by the target and is then accepted by the initiator.

**Status Phase**: Now, the target enters in status phase. It indicates the end of data transfer to the initiator.

**Message Phase**: This is the last phase. It is to interrupt the initiator signaling completion of the read message. The bus free phase is a phase without any activity on the bus so that the bus can settle down before the next transaction. SCSI-l transfers data in 8-bit parallel form, and the transfer rate vades rom I M Bytes/See to 5 M Bytes/Sec. SCSI-I'drive capacity varies from 20 M bytes to 2 GB. SCSI-1 has over 64 commands specified to carry out transactions.

Commands include read, write, seek, enquiry, copy, verify, copy and verify, compare and so on.

**SCSI-2**

It has the same aspects of SCSI -1, But with faster data transfer . rates, and wider data width.

It includes few more new commands, and vender-unique command sets for optical drives, tape drives, scanners and so on. To make the bus wider, a system designer uses a second 68-pin connector in addition to the standard 50 pin connector.

**Magnetic Storage Densities and Latencies**

The Latency is divided into two categories: seek latency and rotational latency. Data management provides the command queuing mechanism to minimize latencies and also set-up the scatter-gather process to gather scattered data in CPU main memory.

**Seek Latencies**: There are three seek latencies available. They are· overlapped seek latency, Mid-transfer seek and Elevator seek.

**Rotational Latencies**: To reduce latency, we use two methods. They are:

(i) Zero latency read/write: Zero latency reads allow transferring data immediately after the head settles. It does not wait for disk revolution to sector property.

(ii) Interleaving factor: It keeps up with the data stream without skipping seccors. It determines the organization of sectors.

**Transfer Rate and I/O per Second**: I/O transfer nite varies from 1.2 M bytes/Sec. to 40 M bytes/Sec. Transfer rate is defined as the rate at which data is transferred from the drive buffer to the host adapter memory.

**Data Management**: It includes Command queueing and Scattergather. Command queueing allows execution of multiple sequential commands with system CPU intervention.Scatter is a process of setting the data for best fit in available block of memory or disk. Gather is a process which reassembles data into contiguous blocks on memory or disk ..

Figure below shows the relationship between seek latency, Rotational latency and Data transfer

It is a method of attaching multiple drives to a single host adapter. The data is written to the first drive first, then after filling it, the controller, allow the data to write in second drive, and so on. Meantime Between Failure (MTBF) = MTBF of single/drivel Total no. of dr

## RAID (Redundant Array of Inexpensive Disks)

It is an alternative to mass storage for multimedia systems that combines throughput speed and reliability improvements.

RAID is an array of multiple disks. In RAID the data is spread across the drives. It achieves fault tolerance, large storage capacity and performance improvement.

If we use RAID as our hot backups, it will be economy. A number of RAID schemes havebeen developed:

1. Hot backup of disk systems
2. Large volume storage at lowercost
3. Higher performance at lower cost
4. Ease of data recovery
5. High MTBF.

There are six levels of RAID available.

**(i) RAID Level 0 Disk Striping**

It spreads data across drives. Data is striped to spread segments of data across multiple drives. Data striping provides high transfer rate. Mainly, it is used for database applications.

RAID level 0 provides performance improvement. It is achieved by overlapping disk reads and writes. Overlapping here means, while segment I is being written to drive 1, segment 2 writes can be initiated for drive 2.

**RAID Level 1 Disk Mirroring**

The Disk mirroring causes two copies of every file to be written on two separate drives. (Data redundancy is achieved).

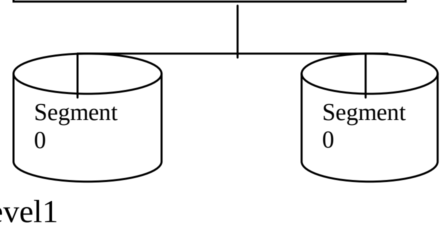
These drives are connected to a single disk controller. It is useful in mainframe and networking systems. Apart from that, if one drive fails, the other drive which has its copy can be used.

Performance:

- Writing is slow.
- Reading can be speeded up by overlapping seeks.
- Read transfer rate and number ofI/O per second is better than a single drive.

I/O transfer rate (Bandwidth) = No. of drives x drive I/O transfer rate

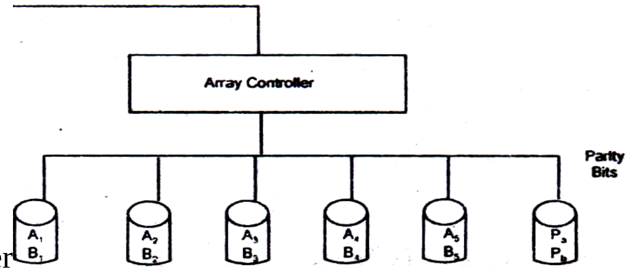
No of I/O's Per second = $\frac{I/Otransferrate}{Averagesizeoftransfer}$




Disk controller arrangement for RAID Level1

RAID Level 2, - Bit interleaving of Data: It contains arrays of multiple drives connected-to a disk array controller.

Data (written one bit at a time) is bit interleaved across multiple drives. Multiple check disks are used to detect and correct errors.

Host adapter

organization of bit interleaving for RAID level2

It provides the ability to handle very large files, and a high level of integrity and reliability. It is good for multimedia system. RAID Level 2 utilizes a hamming error correcting code to correct single-bit errors and doublebit errors.

**Drawbacks:**

(i) It requires multiple drives for error correction (ii) It is an expensive approach to data redundancy. (iii) It is slow.

Uses: It is used in multimedia system. Because we can store bulk of video and audio data.

**RAID Level-3 Parallel Disk Array:** RAID 3 subsystem contains an array of multiple data drives and one parity drive, connected to a disk array controller.

The difference between RAID 2 and RAID 3 is that RAID 3 employs only parity checking instead of the full hamming code error detection and correction. It has the advantages of high transfer rate, cost effective than RAID 2, and data integrity.

**RAID Level-4 Sector Interleaving**: Sector interleaving means writing successive sectors of data on different drives.

As in RAID 3, RAID 4 employs multiple data drives and typically a single dedicated parity drive.

Unlike RAID 3, where bits of data are Written to successive disk drives, an Ri\ID 4, the first sector of a block of data is written to the first drive, the second sector of data is written to the secohd drive, and so on. The data is interleaved at the data level.

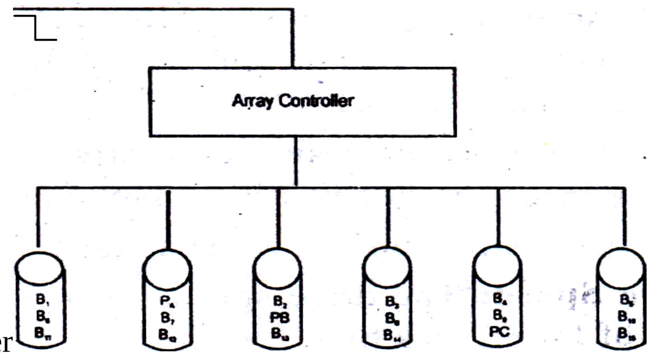
**RAID *Leve1-4*** offers *cost-effective* improvement in performance with data.

**RAID Level-5 Block Interleaving**: In RAID LevelS, as in all the other RAID systems, multiple drives are connected to a disk array controller.

The disk array controller contains multiple SCSI channels.

A RAID 5 system can be designed with a single SCSI host adapter with multiple drives connected to the single SCSI channel.

Unlike RAID Level-4, where the data is sector-interleaved, in RAID Level-5 the data is block-interleaved.

Host adapter


RAID LEVEL 5 DISK ARRAYS

**Optical Media**

CD ROM, WORM (Write once, Read many) and rewritable optical systems are optical drives.

·CD-ROMs have become the primary media of choice for music due to the quality of ,sound.

WORMs and erasable opticel drives both use lasers to pack
information densely on a removable disk.

Optical Media can be classified by technology as follows:

- CD-ROM - Compact Disc Read Only Memory
- WORM - Write Once Read Many
- Rewritable - Erasable

•Multifunction - WORM and Erasable.

•

**CD-ROM**

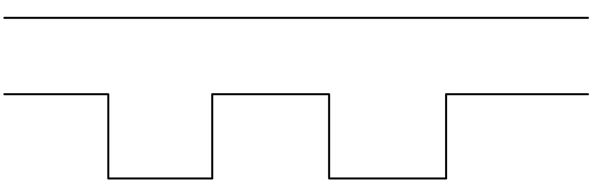
**Physical Construction of CD ROMs**:

It consists of a polycarbonate disk. It has 15 mm spindle hole in the center. The polycarbonate substrate contains lands and pits.

The space between two adjacent pits is called a land. Pits, represent binary zero, and the transition from land to pits and from pits to land is represented by binary one.

The polycarbonate substrate is covered by reflective aluminium or aluminium alloy or gold to increase the reflectivity of the recorded surface. The reflective surface is protected by a coat oflacquer to prevent oxidation. A CD-ROM consists of a single track which starts at the center from inside and spirals outwards. The data is encoded on this track in the form of lands and pits. A single track is divided into equal length sectors and blocks**.**

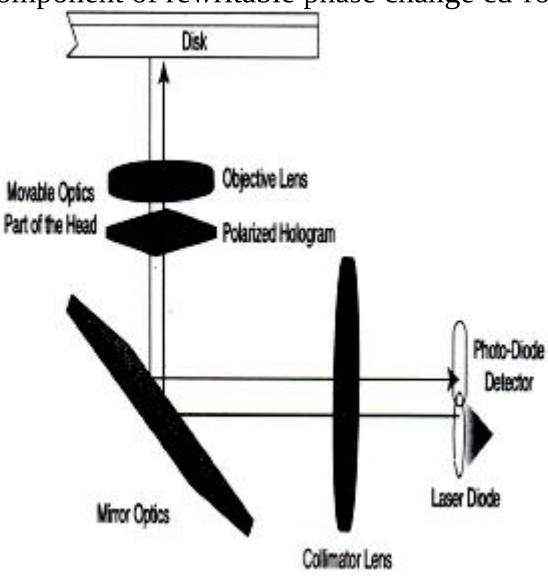
## CD-ROM Physical Layers



Each sector or block consists of2352 bytes, also called a frame. For Audio CD, the data is indexed on addressed by hours, rninutes, seconds and frames. There are 75 frames in a second.

**Magnetic Disk Organization**: Magnetic disks are organized by CYlinder, track and sector. Magnetic hard disks contain concentric circular tracks. They are divided into sector.

Component of rewritable phase change cd-rom




## Organization of magnetic media

CD-ROM Standards : A number of recording standards have emerged for CD-ROMs.

They are:

**CD-DA (DD-Digital Audio) Red Book:** CD-ROM is developed by philips and sony to store audio information. CD-DA is the basic medium for the music industry.

The standard specifies multiple tracks, with one song per track. One track contains one frame worth of data: 2352 bytes. There are 75 frames in a second. Bandwidth = 176 *KB/s.*

**CD-ROM Mode 1 Yellow Book**: The Mode 1 Yellow Book Stnadard was developed for error correction. The Yellow Book Standard dedicates 288 bytes for error detection codes (EDCs) and error correction codes (ECCs).

**CD-ROM Mode 2 Yellow Book**

The Mode 2 Yellow Book standard was developed for compressed audio and video applications where, due to lossy compression, data integrity is not quite as important. This standard maintains the frame stmcture but it does not contain the *ECC/EDC* bytes. Removing the *ECC/EDC* bytes allows a frame to contain an additional 288 bytes of data, resulting in an increase of 14% more data. The frame stmcture is shown in the Table below:

| Synchronization | Header | Data |
|---|---|---|
| 12 Bytes | 4 Bytes | 2336 Bytes |
| 0-11 | 13-15 | 16-2351 |

**CD-ROMXA**

XA stands for Extended Architecture. The standard was created for extending the present CD-ROM format.

CD-ROM XA contains multiple tracks. Each track's content is desclibed by mo~e. CD-ROM XA also allows interleaving audio and video objects with data for synchroni~ed playback. It does not support video compression. It supports audio compression. It uses Adaptive differential pulse Code Modulation algorithms.

**CD-MO Orange Book Part 1**

This standard defines an optional pre-mastered area conforming to the Red, Yellow or Green book standards for read-only, and a recordable area. It utilizes a read/write head similar to that found in magnetooptical drives. We can combine the pre-master multimedia objects as the base and develop their own versions.

**CD-R Orange Book Part 2**

This standard allows writing data once to a writeable disk. Here, the CD contains a polycarbonate substrate with pits and lands.

The polycarbonate layer is covered with an organic dye recording layer.

As in CD-ROM construction, the track starts from the center and spirals outwards. CD-R uses a high powered laser beam. The laser beam alters the state of the organic dye such that when the data is read, the altered state of dye disperses light instead of reflecting it. The reflected beam is measured for reading the state of each bit on the disk.

**Mini-Disk**

Mini-Disk for Data is known as MD-Data. It was developed by Sony Corporation. It is the data version of the new rewritable storage format. It can be used in three formats to support all users.

- A premastered optical disk.
- A recordable magneto-optical disk.
- A hybrid of mastered and recorded.

Its size is 2.5 inch. It provides large capacity. It is low cost. It is used in multimedia applications.

## WORM Optical Drives

It records data using a high power laser to create a permanent burnt-in record of data. The laser beam makes permanent impressions on the surface of the disk.

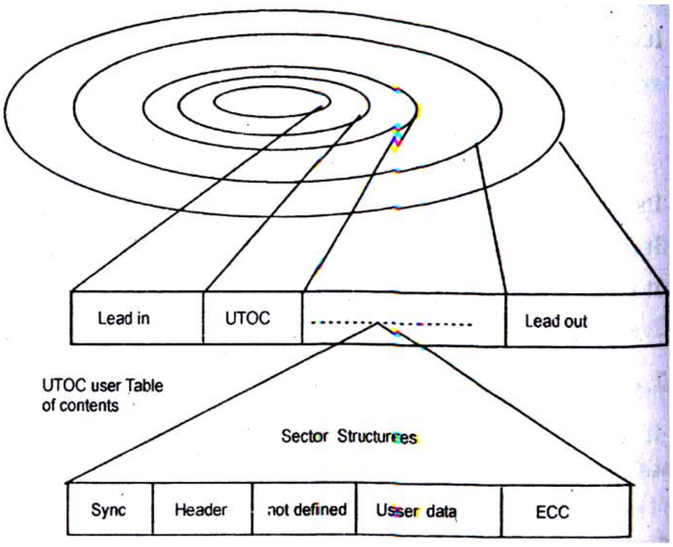
It creates pits. Information is written once. It cannot be written over and cannot be erased. i.e., Here data cannot be edited.

**Recording of information**: During recording, the input Signal is fed to a laser diode. The laser beam from the laser diode is modulated by the inpUt signal. It switches the laser beam on and off. if the beam is on, it strikes the three recording layers.

The beam is absorbed by the bismuth-tellurium layer. Heat is generated within the layer. This heat diffuses the atoms in the three recording layers. It forms four-element alloy layer. Now, the layer becomes recorded layers.

**Reading Information from disk**:

During disk read, a weaker, laser beam is focused on to the disk. It is reflectted back. The beam splitter mirror and lens arrangement sends the reflected beam to the photo detector. The photo sensor detects the beam and converts it into an electrical signal.




**WORM DRIVE Applications**

- On-line catalogs
- Large-volume distribution
- Transaction logging
- Multimedia archival.

**Rewritable Optical Disk Technologies**

This technology allows erasing old data and rewriting new data over old data. There are two types of rewritable technology: (i) Magneto-optical ii)Phase change.

## Magneto-Optical Technology

It uses a combination of magnetic and laser technology to achieve read/write capability. The disk recording layer uses a weak magnetic field to record data under high temperature. High temperature is achieved by laser beam.

When the beam is on, it heats the spot on the magneto optical disk to its curie temperature. The rise in temperature makes the spot extra sensitive to the magnetic field of bias field.

Magneto-optical drives require two passes to write data; in the first pass, the magneto optical head goes through an erase cycle, and in the second pass, it writes the data.

During the erase cycle, the laser beam is turned on and the bias field is modulated to change the polarity of spots to be erased. During the write cycle, the bias field is turned on and the laser beam is modulated to change the polarity of some spots to 1 according to the bit value.

## Phase change Rewritable optical Disk

In phase change technology the recording layer changes the physical characteristics from crystalline to amorphous and back under the influence of heat from a laser beam.

To read the data, a low power laser beam is transmitted to the disk. The reflected beam is different for a crystalline state than for an amorphous state. The difference in reflectivity determines the polarity of the spot.

**Benefits**: it requires only one pass to write.

**Dye Polymer Rewritable Disk**

There is no need of magnetic technology here.

This technology consists of giant molecules formed from smaller molecules of the same kind with light-sensitive dye. This technology is also used in WORM drives.

**HIERARCHICAL STORAGE MANAGEMENT**

multi-function drive is a single drive unit. It is capable of reading and writing a variety of disk media. Three types of technologies are used for multi-function drives. They are:

(i) Magneto-optical disk for both rewritable and WORM capability.
(ii) Magneto-optical disk for rewritable and dye polymer disk for WORM capability.
(iii) Phase change technology for both rewritable and WORM capability.

The storage hierarchies described in thE pyramid consist of random access memory (RAM), on-line fast magnetic hard disks, optical disks and juke boxes, diskettes, and tapes.

**Permanent Vs. Transient Storage issues**

The process of moving an object from one level in the storage hierarchy to another level in that hierarchy is called migration. Migration to objects to off-line media and removal of these objects from on-line media is called archiving. Migration Can be set up to be manual or automatic.

Manual migration requires the user or the system administrator to move objects from one level of storage to another level. Systems with automatic migration perform this task automatically. In document-imaging systems, compressed image files are created in magnetic cache areas on fast storage devices when documents are scanned.

**Optical Disk Library (Juke box)**

An optical juke box stacks disk platters to be played. In the optical disk library, the platters are optical and contain objects such as data, audio, video, and images.

An optical disk library has one or more optical drives. An optical disk library uses a very-high-speed and accurate server-controlled electromechanical robotics elevator mechanism for moving the optical platters between their slots on a disk stacks and the drives. The robotics mechanism removes disk platter from a drive and returns it to its slots on the stack after the disk has finished playing (usually when the drive is required for another disk). The robotics device operates and manages multiple drives under program control. .

A juke box may contain drives of different types, including WORM, rewritable, or multifunction. Juke boxes contain one or more drives. A juke box is used for storing large volumes of multimedia information in one cost effective store.

Juke box-based optical disk libraries can be networked so that multiple users can access the information. Optical disk libraries serve as near-line storage for infrequently used data.

**Hierarchical Storage Applications**: Banks, insurance companies, hosptials, state and federal governments, manufacturing companies and a variety of other business and service organizations need to permanently store large volumes of their records, from simple documents to video information, for audit trail use.

**CACHE MANAGEMENT FOR STORAGE SYSTEMS**

Disk caches are an integral part of a hierarchical storage management architecture. Hierarchical storage consists of a number of media classes, ranging from high speed and expensive on-line fast cache storage to low-cost off-line storage.

**Role of on-line caches:** The primary role of on-line caches as used in document-imaging systems is to provide high speed on-line storage for documents currently in use that may be accessed in the future. This role can be extended to multimedia systems.

Hierarchical Organization of Caches: Caches are used at various storage levels.

The following lists representative storage systems using cache storage.

~ Hardware disk caches and system memory caches for stand alone systems.

~ Disk storage caches for optical disk libraries.

~ Disk storage caches for networked systems.

**Low-level Disk Caching**

**Disk Caching Controllers**: Two approaches to implement disk caching controllers are: in hardware and software. A hardware caching controller is designed with its own on-board CPU and private memory

The private memory is used for storing disk data temporarily. It is known as disk cache.

When an *1/0* request is received by the caching controller, the CPU on the caching controller initiates sectors of data, including sectors that contain the requested data.

Disk writes managed through a disk cache can be delayed writes or write-throughs. For delayed through, data is written by the host CPU to the disk cache, and the caching controller writes the data from the disk cache to the disk when read activity is low.

**Cache Organization for Hierarchical Storage Systems**: The hierarchical, storage management system consists of at least three or four types of storage as follows:

- System memory Cache
- On-line high speed magnetic disk storage
- Near-line optical disk libraries
- Off-line optical tape storage
-

Many Cache designs use a high-water mark and a low-water mark to trigger cache management operations. When the cache storage fills up to the high-water mark, the cache manager starts creating more space in cache storage. Space is created by discarding objects.

- The cache manager maintains a data base of objects in the cache. Cache areas containing updated objects are frequently called dirty cache.
- Objects in dirty cache are written back at predetermined time intervals or before discarding an object.

.

**UNIT V HYPERMEDIA 9**

Multimedia authoring and user interface − Hypermedia messaging − Mobilemessaging − Hypermedia message component − Creating hypermedia message −Integrated multimedia message standards − Integrated document management −Distributed multimedia systems.

## 5.1 Multimedia authoring and User Interface

## Multimedia Authoring Systems

Multimedia authoring systems are designed with two primary target users:

They are

(i) Professionals who prepare documents, audio or sound tracks, and full motion video clips for wide distribution.

(il) Average business users preparing documents, audio recordings, or full motion video clips for stored messages' or presentations.

The authoring system covers user interface. The authoring system spans issues such as data access, storage structures for individual components embedded in a document, the user's ability to browse through stored objects, and so on.

Most authoring systems are managed by a control application.

## Design Issues for Multimedia Authoring

Enterprise wide standards should be set up to ensure that the user requirements are fulfilled with good quality and made the objects transferable from one system to another.

So standards must be set for a number of design issues

1. Display resolution
2. Data formula for capturing data
3. Compression algorithms
4. Network interfaces
5. Storage formats.

**Display resolution**

A number of design issues must be considered for handling different display outputs. They are:

(a) Level of standardization on display resolutions.

(b) Display protocol standardization.

(c) Corporate norms for service degradations

(d) Corporate norms for network traffic degradations as they relate to resolution issuesSetting norms will be easy if the number of different work station types, window managers, and monitor resolutions are limited in number.But if they are more in number, setting norms will be difficult.Another consideration is selecting protocols to use. Because a number of protocols have emerged, including AVI, Indeo, Quick Time and so on.So, there should be some level of convergence that allows these three display protocols to exchange data and allow viewing files in other formats.

## File Format and Data Compression Issues

There are variety of data formats available for image, audio, and full motion video objects.

Since the varieties are so large, controlling them becomes difficult. So we should not standardize on a single format. Instead, we should select a set for which reliable conversion application tools are available.

Another key design Issue is to standardize on one or two compression formula for each type of data object. For example for facsimile machines, CCITT Group 3 and 4 should be included in the selected standard. Similarly, for full motion video, the selected standard should include MPEG and its derivatives such as MPEG 2.

While doing storage, it is useful to have some information (attribute information) about the object itself available outside the object to allow a user to decide if they need to access the object data. one of such attribute information are:

(i) Compression type (ii) Size of the object

(iii) Object orientation (iv)Data and time of creation

(v) Source file name (vi)Version number (if any)

(vii) Required software application to display or playback the object.

**Service degradation policies**: Setting up Corporate norms for network traffic degradation is difficult as they relate to resolution Issues:

To address these design issues, several policies are possible. They are:

1. Decline further requests with a message to try later.
2. Provide the playback server but at a lower resolution.
3. Provide the playback service at full resolution but, in the case of sound and full motion video, drop intermediate frames.

**Design Approach to Authoring**

Designing an authoring system spans a number of design issues. They include:

Hypermedia application design specifics, User Interface aspects, Embedding/Linking streams of objects to a main document or presentation, Storage of and access to multimedia objects. Playing back combined streams in a synchronized manner.

A good user interface design is more important to the success of hypermedia applications.

## Types of Multimedia Authoring Systems

There are varying degrees of complexity among the authoring systems. For example, dedicated authoring systems that handle only one kind of an object for a single user is simple, where as programmable systems are most complex.

**Dedicated Authority Systems**

Dedicated authoring systems are designed for a single user and generally for single streams.

Designing this type of authoring system is simple, but if it should be capable of combining even two object streams, it becomes complex. The authoring is performed on objects captured by the local video camera and image scanner or an objects stored in some form of multimedia object library. In the case of dedicated authoring system, users need not to be experts in multimedia or a professional artist. But the dedicated systems should be designed in such a way that. It has to provide user interfaces that are extremely intuitive and follow real-world metaphors.

A structured design approach will be useful in isolating the visual and procedural design components.

**TimeLine –based authoring**

In a timeline based authoring system, objects are placed along a timeline. The timeline can be drawn on the screen in a window in a graphic manner, or it created using a script in a mann.er similar to a project plan. But, the user must specify a resource object and position it in the timeline.

On playback, the object starts playing at that point in the time Scale.

Fig:TimeLinebased authoring

In most timeline based approaches, once the multimedia object has been captured in a timeline,.it is fixed in location and cannot be manipulated easily, So, a single timeline causes loss of information about the relative time lines for each individual object.

**Structured Multimedia Authoring**

A structured multimedia authoring approach was presented by Hardman. It is an evolutionary approach based on structured object-level construction of complex presentations. This approach consists of two stages:

(i) The construction of the structure of a presentation.

(ii) Assignment of detailed timing constraints.

A successful structured authoring system must provide the following capabilities for navigating through the structure of presentation.

1.Ability to view the complete structure.

2.Maintain a hierarchy of objects.

3.Capability to zoom down to any specific component.

4.View specific components in part or from start to finish.

5.Provide a running status of percentage full of the designated length of the presentation.

6.Clearly show the timing relations between the various components.

7.Ability to address all multimedia types including text, image, audio, video and frame based digital images.

The author must ensure that there is a good fit within each object hierarchy level. The navigation design of authoring system should allow the author to view the overall structure while examining a specific object segment more closely.

**Programmable Authoring Systems** :Early structured authoring tools were not able to allow the authors to express automatic function for handling certain routine tasks. But, programmable authoring system bas improved in providing powerful functions based on image processing and analysis and embedding program interpreters to use image-processing functious.

The capability of this authoring system is enhanced by Building user programmability in the authoring tool to perform the analysis and to manipulate the stream based on the analysis results and also manipulate the stream based on the analysis results. The programmability allows the following tasks through the program interpreter rather than manually. Return the time stamp of the next frame. Delete a specified movie segment. Copy or cut a specified movie segment to the clip board . Replace the current segment with clip board contents.

## Multisource Multi-user Authoring Systems

We can have an object hierarchy in a geographic plane; that is, some objects may be linked to other objects by position, while others may be independent and fixed in position".

We need object data, and information on composing it. Composing means locating it in reference to other objects in time as Well as space.

Once the object is rendered (display of multimedia object on the screen) the author can manipulate it and change its rendering information must be available at the same time for display.If there are no limits on network bandwidth and server performance, it would be possible to assemble required components on cue at the right time to be rendered.

In addition to the multi-user compositing function A multi user authoring system must provide resource allocation and scheduling of multimedia objects.

## Telephone Authoring systems

There is an application where the phone is linking into multimedia electronic mail application

1.Tele phone can be used as a reading device by providing fill text to-speech synthesis capability so that a user on the road can have electronic mail messages read out on the telephone.

2. The phone can be used for voice command input for setting up and managing voice mail messages. Digitized voice clips are captured via the phone and embedded in electronic mail messages.

3. As the capability to recognize continuous speech is deploy phones can be used to create electronic mail messages where the voice is converted to ASCII text on the fly by high-performance voice recognition engines.

Phones provide a means of using voice where the alternative of text on a screen is not available. A phone can be used to provide interactive access to electronic mail, calendar information databases, public information databass and news reports, electronic news papers and a variety of other applications. !ntegrating of all these applications in a common authoring tool requires great skill in planning.

The telephone authoring systems support different kinds of applications. Some of them are:

1.Workstation controls for phone mail.

2.Voice command controls for phone mail.

3.Embedding of phone mail in electric mail.

**Hypermedia Application Design Consideration**

The user interface must be highly intuitive to allow the user to learn the tools quickly and be able to use them effectively. In addition, the user interface should be designed to cater to the needs of both experienced and inexperienced user.

In addition to control of their desktop environments, user also need control of their system environment. This control should include some of the following:

- The ability to specify a primary server for each object class within a domain specified by the system administrative. A domain can be viewed as a list of servers to which they have unrestricted access.
- The ability to specify whether all multimedia -objects or only references should be replicated.
- The ability to specify that the multimedia object should be retrieved immediately for display versus waiting for a signal to "play" the object. This is more significant if the object must be retrieved from a remote server.
- Display resolution defaults for each type of graphics or video object.

**Essential for good hypermedia design:**

1.Determining the type of hypermedia application.

2.Structuring the information.
3.Determining the navigation throughout the application.
4.Methodologies for accessing the information.
5.Designing the user interface.

**Integration of Applications**

The computer may be called upon to run a diverse set of applications, including some combination of the following:

1.Electronic mail.
2.Word processing or technical publishing.
3.Graphics and formal presentation preparation software. .
4.. Spreadsheet or some other decision support software.
5.Access to a relational on object-oriented database.
6.Customized applications directly related to job function:
   * Billing  * Portfolio management    * Others.

Integration of these applications consists of two major themes: the appearance of the applications and the ability of the applications to exchange of data.

**Common UI and Application Integration**

Microsoft Windows has standardized the user interface for a large number of applications by providing standardization at the following levels: Overall visual look and feel of the application windows

This standardization level makes it easier for the user to interact with applications designed for the Microsoft Windows operational environment. Standardization is being provided for Object Linking and Embedding (OLE), Dynamic Data Exchange (DOE), and the Remote Procedure Call (RPC).

**Data Exchange**

The Microsoft Windows Clipboard allows exchanging data in any format. It can be used to exchange multimedia objects also.  We can cut and copy a multimedia objects in one document and pasting in another. These documents can be opened under different applications.The windows clipboard allows the following formats to be stored:

.:. Text Bitrnap
.:. Image Sound
.:. Video (AVI format).

**Distributed Data Access**

If all applications required for a compound object can access the subobjects that they manipulate, then only application integration succeeds.

Fully distributed data access implies that any application at any client workstation in the enterprise-wide WAN must be able to access any data object as if it were local. The underlying data management software should provide transport mechanisms to achieve transparence for the application.

**Hypermedia Application Design**

Hypermedia applicati'ons are applications consisting of compound objects that include the multimedia objects. An authoring applicationn may use existing multimedIa objects or call upon a media editor to CD create new object.

## Structuring the Information

A good information structure should consist the following modeling primitives:

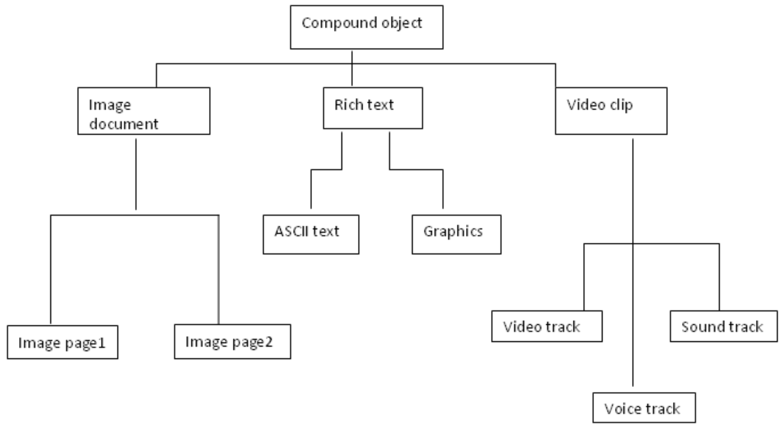
**.:.** Object types and object hierarchies.
**.:.** Object representations.
**.:.** Object connections.
**.:.** Derived connections and representations.

The goal of information Structuring is to identify the information objects and to develop an information model to define the relationships among these objects.

**Types and Object Hierarchies**

Object types are related with various attributes and representations of the objects. The nature of the information structure determines the functions that can be performed on that information set. The object hierarchy defines a contained-in relationship between objects. The manner in which this hierarchy is approached depends on whether the document is being created or played back.

**Example of an Object Hierarchy**




Users need the ability to search for an object knowing very little about the object. Hypermedia application design should allow for such searches.

The user interface with the application depends on the design of the application, particularly the navigation options provided for the user.

**Object representations**

Multimedia objects have a variety of different object representations. A hypermedia object is a compound object, consists of s~ information elements, including data, text, image, and video

Since each of these multimedia objects may have its own sub objects, the design must consider the representation of objects.

An object representation may require controls that allow the user to alter the rendering of the object dynamically. The controls required for each object representation must be specified with the object.

**Object connection**

In the relational model, the connections are achieved through joins, and in the object oriented models, through pointers hidden inside objects. Some means of describing explicit connections is required for hypermedia design to define the relationships among objects more clearly and to help in establishing the navigation.

**Derived Connections and Representations**

Modeling of a hypermedia system should attempt to take derived objects into consideration for establishing connection guidelines.

**User Interface Design** Multi media applications contain user interface design. There are four kinds of user interface development tools. They are

1. Media editors
2. An authoring application
3. Hypermedia object creation
4. Multimedia object locator and browser

A media editor is an application responsible of the creation and editing of a specific multimedia object such as an image, voice, or Video object. Any application that allows the user to edit a multimedia object contains a media editor. Whether the object is text, ~voice, or full-motion video, the basic functions provided by the editor are the same: create, delete, cut, copy, paste, move, and merge.

**Navigation through the application**

Navigation refers to the sequence in which the application progresses and objects are created, searched and used.

Naviation can be of three modes:

(i) Direct: It is completely predefined. In this case, the user needs to know what to expect with successive navigation actions.

**Free-form mode**: In this mode~ the user determines the next sequence of actions.

**Browse mode**: In this mode, the user does not know the precise question and wnats to get general information about a particular topic. It is a very common mode in application based on large volumes of non-symbolic data. This mode allows a user to explore the databases to support the hypothesis.

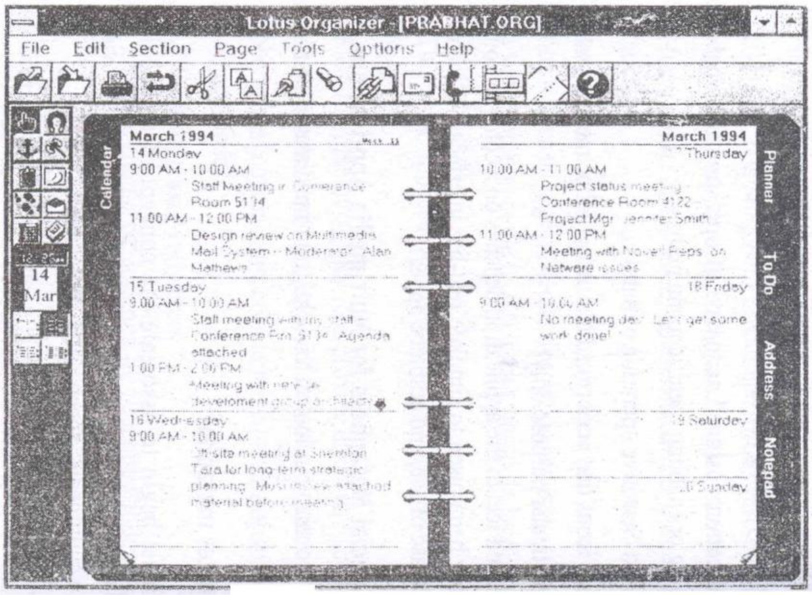
## Designing user Interfaces

User Interface should be designed by structured following design guidelines as follows:

1.Planning the overall structure of the application

2.Planning the content of the application

3.Planning the interactive behavior
4.Planning the look and feel of the application
A good user interface must be efficient and intuitive by most users.
The interactive behaviour of the application determines how the User interacts with the application. A number of issues are determined at this level.
They are Data entry dialog boxes
Application designed sequence of operation depicted by graying or enabling specific menu items Context-Sensitive operation of buttons. Active icons that perform ad hoc tasks (adhoc means created for particular purpose only)




A look and feel of the application depends on a combination of the metaphor being used to simulate real-life interfaces, Windows guidelines, ease of use, and aesthetic appeal.

**Special Metaphors for Multimedia Applications**

In this section let us look at a few key multimedia user interface metaphors.

**The organizer metaphor**

One must begin to associate the concept of embedding multimedia object in the appointment diary or notepad to get obvious view of the multimedia aspe.cts of the organizer.
Other use of multimedia object in an organizer is to associate maps or voice mail directions with addresses in address books.
The lotus organizer was the first to use a screen representation of the office diary type organizer
**'Telephone Metaphor**: The role of the telephone was changed b the advent of voice mail system. Voice mail servers convert the analog voice and store it in digital form. With the standards for voice ~ail file formats and digital storage of sound for computer. Now, computer system is used to manage the phone system. The two essential components of a phone system are speakers and microphones. They are included in most personal computers.
Figure 5.5 shows how a telephone can be created on a screen to make it a good user interface
The telephone keypad on the screen allows using the interface just as a telephone keypad is used. Push buttons in dialog boxes and function selections in memos duplicate the function provided by the keypad. Push buttons, radio buttons, list boxes, and data entry fields and menu selections allow a range of functionality than can be achieved by the telephone.
**Aural User Interface**: A Aural user interface allows computer systems to accept speech as direct input and provide an oral response to the user actions. Speech enabling is an important feature in this UI. To design AUI system first, we have to create an aural desk top which substitutes voice and ear for the keyboard and display and be able to mix and match them Aural cues should be able to represent icons, voice, menus and the windows of graphical user interface.
AUl design involves human perception, cagnitive science and psycho-acoutic theory. AUI systems learn systems to perform routine functions without user's feedback. An AUI must be temporal and use time based metaphors.
AUI has to address the following issues

1. Recent user memory
2. Attention span
3. Rhythms
4. Quick return to missed oral cues

The VCR metaphor: The User interface metaphor for VCR is to draw a TV on screen and provide live buttons on it for selecting channels, increasing sound volume and changing channel.

User interface for functions suchas video capture, channel play, and stored video playback is to emulate the camera, television and VCR on screen Fi5.6 shows all functions of typical video camera when it is in a video capture mode.

**Audio/Video Indexing Functions**

Index marking allowed users to mark the location on tape in the case of both audio and video to which they may wish to fast forward are rewind.

Other form of index marking is time based. In his form the tape counter shows playtime in hours, minutes, and secondsfrom the time the counter was reset.

Three paradigms for indexing audio and video tapes are

Counter identify tape locations, and the user maintains index listingSpecial events are used as index markersUsers can specify locations for index markings and the system maintains the index.Indexing is useful only if the video is stored. Unless live video is stored, indexing information is lost since the video cannot be repeated.In most systems where video is stored, the sound and video streams are decompressed and managed separately, so synchronization for playback is important. The indexing information n\must be stored on apermanent basis.

**Information Access:**

Access structure defines the way objects can be accessed and how navigation takes place through the information objects.

The common forms of navigations for information access are:

**Direct**: Direct information accessis completely predefined. User must have knowledge about the object that need to be accessed. That information includes object representations in a compound object.

**Indexed:** Index access abstracts the real object from the access to the object. If the object ID of the object is an index entry that resolves to a filename on a specific server and disk partition, then the information access mechanism is an indexed mechanism. \

**Random Selection:** In this fonn, the user can pick one of several possible items. The items need not arranged in any logical sequence; and they need not to be displayed sequentially. The user need not have much knowledge about the infonnation. They must browse through the infornlation.

**Path selection or Guided tour:** In guided tour, the application guides the user through a predefined path acrosS a number of objects and operations. The user may pause to examine the objects at any stage, but the overall access is controlled by the application. Guided tours can also be used for operations such as controlling the timing for discrete media, such as slide show. It can be used for control a sound track or a video clip.

**Browsing:** It is useful when the user does not have much knowledge about the object to access it directly.

**Object Display Playback Issues**: User expects some common features apart from basic functions for authoring systems. And to provide users with same special control on the display/ playback of these objects, designer have to address some of these issues for image, audio and video objects.

**Image Display Issues Scaling**: Image scaling is performed on the fly after decompressio The image is scaled to fit in an application defined window at t:' full pixel rate for the window.The image may be scaled by using factors. For eg: for the window 3600 x 4400 pixels can be scaled by a factor of 6 x 10 ie.60 x 440 (60 times).

**Zooming**: Zooming allows the user to see more detail for a specific area of the image. Users can zoom by defining a zoom factor (eg: 2: 1,5: 1 or 10: 1). These are setup as preselected zoom values.

**Rubber banding:** This is another form of zooming. In this case, the user uses a mouse to define two comers of the rectangle. The selected area can be copied to the clipboard, cut, moved or zoomed.

**Panning**: If the image window is unable to display the full image at the ·selected resolution for display. The image can be panned left to right or right to left as wellas top to bottom or bottom to top. Panning is useful for finding detail that is not visible in the full image.

**Audio Quality**: Audio files are stored in one of a number of formats, including WAVE and A VI. Playing back audio requires that the audio file server be capable of playing back data at the rate of 480 kbytes/min uncompressed or 48 kbytes/min for compressed 8 bit sound or 96 kbytes/min for 16 bit sound.

The calculation is based on an 8 MHz sampling rate and ADCPM compression with an estimated compression ratio. 32 bit audio will need to be supported to get concert hall quality in stored audio. Audio files can be very long. A 20 minute audio clip is over 1 MB long. When played back from the server, it must be transferred completely in one burst or in a controlled manner.

**Special features for video playback**: Before seeing the features of video playback let us learn what is isochronous playback. The playback at a constant rate to ensure proper cadence (the rise and fall in pitch of a person's voice) is known as isochronous playback. But isochronous playback is more complex With video than It is for sound. .

If video consists of multiple clips of video and multiple soundtracks being retrieved from different servers and combined for playback by accurately synchronizing them, the problem becomes more complex.To achieve isochronous playback, most video storage systems

use frame interleaving concepts. Video Frame Interleaving: Frame interleaving defines the structure o;the video file in terms of the layout of sound and video components.

**Programmed Degradation**: When the client workstation is unable to keep up with the incoming data, programmed degradation occurs. Most video servers are designed to transfer data from storage to the client at constant rates. The video server reads the file from storage, separate the sound and video components, and feeds them as a seperate streams over the network to the client workstations. Unless specified by the user, the video server defaults to favoring sound and degrades video playback by dropping frames. So, sound can be heard on a constant basis. But the video loses its smooth motion and starts looking shaky. Because intermediate frames are not seen.

The user can force the ratio of sound to video degradation by changing the interleaving factor for playback; ie the video server holds back sound until the required video frames are transferred. This problem becomes more complex when multiple streams of video and audio are being played back from multiple source servers. .

**Scene change Frame Detection:** The scene we see changes every few seconds or minutes and it replaced by a new image.Even within the same scene, there may be a constant motion of some objects in a scene.

**Reason for scene change detection:** Automating scence change detection is very useful for browsing through very large video clips to find the exact frame sequence of interest. Spontaneous scene change detection provides an automatic indexing mechanism that can be very useful in browsing. A user can scan a complete video clip very rapidly if the key frame for each new scene is displayed in an iconic (poster frame) form in a slide sorter type display. The user can then click on a specific icon to see a particular scene. This saves the user a significant amount of time and effort and reduces resource load by decompressing and displaying only the specific scene of interest rather than the entire video.

Scene change detection is of real advantage if it can be performed without decompressing the video object. Let us take a closer-look at potential techniques that can be employed for this purpose.

Techniques:

**(i) Histogram Generation**: Within a scene, the histogram changes as the subject of the scene mover. For example, if a person is running and the camera pans the scene, a large part of the scene is duplicated with a little shift. But if the scene changes from a field to a room, the histogram changes quite substantially. That is, when a scene cuts over to a new scene, the histogram changes rapidly. Normal histograms require decompressing the video for the successive scenes to allow the optical flow of pixels to be plotted on a histogram. The foot that the video has tobe decompressed does help in that the user can jump from one scene to the nect. However, to show a slide sorter view requires the entire video to be decompressed. So this solution does not really of the job.

Since MPEG and JPEG encoded video uses DCT coefficients, DCT quantization analysis on uncompressed video or Audio provides the best alternatives for scene change detection without decompressing video

The efficiency can be managed by determining the frame interval for checks and by deciding on the regions within the frame that are being checked. A new cut in a scene or a scene change can be detected by concentrating on a very small portion of the frame

The scene change detection technology as is the case with video compression devices as well as devices that can process compressed video, the implementations of scene change detection can be significantly enhanced.

## Video scaling, Panning and Zooming:

**Scaling:**

Scaling is a feature since users are used in changing window sizes. When the size of the video window is changed, scaling take place.

**Panning:** Panning allows the user to move to other parts of the window. Panning is useful incombination with zooming. Only if the video is being displayed at full resolution and the video window is not capable of displaying the entire window then panning is useful. Therfore panning is useful only for video captured using very high resolution cameras.

**Zooming:**

Zooming implies that the stored number of pixels is greater than the number that can be displayed in the video window . In that case, a video scaled to show the complete image in the video window can be paused and an area selected to be shown in a higher resolution within the same video window. The video can be played again from that point either in the zoomed mode or in scaled to fit window mode.

**Three Dimensional Object Display and VR(Virtual Reality)**

Number of 3D effects are used in home entertainment a advanced systems used for specialized applications to achieve find Ine results.

Let us review the approaches in use to determine the impact 0 multimedia display system design due to these advanced systems.

**Planar Imaging Technique**: The planar imaging technique, used in computer-aided tomography (CAT Scan) systems, displays a twodimensional [20] cut of X-ray images through multidimensional data specialized display techniques try to project a 3D image constructed from the 2D data. An important design issue is the volume of data being displayed (based on the image resolution and sampling rate) and the rate at which 3D renderings need to be constructed to ensure a proper time sequence for the changes in the data.

Computed tomography has a high range of pixel density and can be used for a variety of applications. Magnetic resonance imaging, on the other hand, is not as fast, nor does it provide as high a pixel density as CT. Ultrasound is the third technique used for 3D imaging in the medical and other fields. .

## 5.2 HYPER MEDIA MESSAGING

Messaging is one of the major multimedia applications. Messaging started out as a simple text-based electronic mail application. Multimedia components have made messaging nuch more complex.

We see how these components are added to messages.

### Mobile Messaging

Mobile messaging represents a major new dimension in the users interaction with the messaging system. With the emergence of remote access from users using personal digital assistants and notebook computers, made possible by wireless communications developments supporting wide ranging access using wireless modems and cellular telephone links, mobile messaging has significantly influence messaging paradigms.

Hypermedia messaging is not restricted to the desktops; it is increasingly being used on the road through mobile communications in metaphors very different from the traditional desktop metaphors.

## 5.3 Hypermedia Message Components

A hypermedia message may be a simple message in the form of text with an embedded graphics, sound track, or video clip, or it may be the result of analysis of material based books, CD ROMs, and other on-line applications. An authoring sequence for a message based on such analysis may consist of the following components.

1. The user may have watched some video presentation on the material and may want to attach a part of that clip in the message. While watching it, the user marks possible quotes and saves an annotated copy.
2. Some pages of the book are scanned as images. The images provide an illustration or a clearer analysis of the topic
3. The user writes the text of the message using a word processor. The text summarizes the highlights of the analysis and presents conclusions.

These three components must be combined in a message using an authoring tool provided by the messaging system. The messaging system must prompt the user to enter the name of the addressee forthe message.

The message system looks up the name in an online directory and convert it to an electronic addresses well as routing information before sending the message. The user is now ready to compose the message. The first step is to copy the word processed text report prepared in step 3 above in the body area of the message or use the text editor provided by the messaging system. The user then marks the spots where the images are referenced and uses the link and embed facilitites of the authoring tool to link in references to the images. The user also marks one or more spots for video clips and again uses the link and embed facilities to add the video clips to the message

When the message is fully composed, the user signs it (electronic signature) and mails to the message to the addressee (recipient). The addressing system must ensure that the images and video clips referenced in the message are also transferred to a server "local' to the recipient.

**Text Messages**

In earlier days, messaging systems used a limited subset of plain ASCII text. Later, messaging systems were designed to allow users to communicate using short messages. Then, new messaging standards have added on new capabilities to simple messages. They provide various classes of service and delivery reports.

Pratap

To : Karan

Copy to: Madhan Date : 01 Jan'07

subject: WISHING A HAPPY NEW YEAR

Hai Karan,

I wish you a very bright and prosperous new year. - Pratap Delivery notification: Normal Priority: High

**Typical Electronic mail message**

Other capabilities of messaging systems includ~ a name and address directory of all users accessible to the messaging system.

**Rich-Text Messages**

Microsoft defined a standard for exporting and importing text data that included character set, font table, section and paragraph formatting, document formatting, and color information-called Rich Text Format (RTF), this standard is used for storage as well as Import and export of text files across a variety of word-processing and messaging systems.

When sections of this document are cut and pasted into another application, the font and formatting information is .retained. This allows the target application to display the text m the nearest equivalent fonts and formats.

Rich-text messages based on the RTF formats provide the capability to create messages in one word processor and edit in another at the recipient end. Most messaging systems provIde richtext capability for the field of a message.

**Voice Messages**

Voice mail systems answer telephones using recorded messages and direct the caller through a sequence of touch tone key operations until the caller is connected to the desired party or is able to leave a recorded message.

**Audio' (Music)**

The Musical Instrument Digital interface (MIDI) was developed initially by the music industry to allow computer control of and music recordings from musical instruments such as digital pianos and electric keyboards. MIDI interfaces are now being used for a variety of peripherals, including digital pianos, digital organs, video games with high-fidelity sound output, and business presentations.

**Full-Motion Video Management**

Use of full-motion video for information repositories and memos are more informative. More information can be 'conveyed and explained in a short full-motion video clip than can be conveyed In a long text document. Because a picture is equivalent to thousand words.

**Full Motion video Authoring System**

An authoring system is an important component of a multimedia messaging system. A good authoring system must provide a number of tools for the creation and editing of multimedia objects. The subset of tools that are necessary are listed below:

1. A video capture program - to allow fast and simple capture of digital video from analog sources such as a video camera or a video tape. .
2. Compression and decompression Interfaces for compressing the captured video as it is being captured.
3. A video editor with the ability to decompress, combine, edit, and compress digital video clips.
4. Video indexing and annotating software for marking sections of a videoclip and recording annotations.

Identifying and indexing video clips for storage.

**Full-Motion Video Playback Systems**

The playback system allows the recipient to detach the embedded vIdeo reference object, Interpret its contents and retrieve the actual video clip from a specialized video server and launch the Playback application. A number of factors are involved in playing back the video correctly.

They are:

1.How the compression format used for the storage of the video clip relates to the available hardware and software facilities for decompression.

2.Resolution of the screen and the system facilites available for managing display windows. The display resolution may be higher or lower than the resolution of the source of the video clip.

3.The CPU processing power and the expected level of degradation as well as managing the degraded output on the fly.

4.Ability to determine hardware and software facilities of the recipient's system, and adjusting playback, parameters to provide the best resolution and perfonnance on playback.

The three main technologies for playing full motion video are microsoft's video for windows: Apple's Quicktime, and Intel's Indeo.

**Video for Windows (VFW):** It is the most common environment for multimedia messaging.

VFW provides capture, edit, and playback tools for full-motion video. The tools provided by VFW are: The VidCap tool, designed for fast digital video capture.

The VidEdit tool designed for decompression, edition, and compressing full-motion digital video. The VFW playback tool.

The VFW architecture uses OLE. With the development of DDE and OLE, Microsoft introduced in windows the capability to link or multimedia objects in a standardized manner. Hence variety :;windows based applications can interact with them. We can add full-motion video to any windows-based application with the help ofVFW. The VFW playback tool is designed to use a number of codecs (software encoder/decoders) for decompressing and playing video files. The default is for A VI files.

**Apple's QuickTime**

An Apple QuickTime product is also an integrated system for playing back video files. The QuickTime product supports four compression methodologies.

**Intel's Indeo**

Indeo is a digital video recording format. It is a software technology that reduces the size of un compressed video files through successive compression methodologies, including YUV sub sampling, vector quantization, Huffman's run-length encoding, and variable content encoding. Indeo technology is designed to be scalable for playing back video; It determines the hardware available and optimizes playback for the hardware by controlling the frame rate. The compressed file must be decompressed for playback. The Indeo technology decompresses the video file dynamically in real time for playback. Number of operating systems provide Indeo technology as standard feature and with other software products (eg. VFW).

**Hypermedia Linking and Embedding**

Linking and embedding are two methods for associating multimedia objects with documents.

**Linking Objects**

When an object is linked, the source data object, called the link source, continues to stay whenever it was at the time the link was created. This may be at the object server where it was created, or where it has been copied.

Only reference is required in the hypermedia document. The' reference is also known as link. This link reference includes information about the multimedia object storage, its presentation parameters, and the server application that is needed to dIsplay/play or edit it. When this document is copied, the link reference is transferred. But the actual multimedia document remains in its original location. A linked object is not a part of the hypermedia document and it does not takeup storage space within the hypermedia document. If the creator, or authorised user edits the original stored multimedia object, subsequent calls to the linked object bring the copy.

**Embedded Objects**

If a copy of the object is physically stored in the hypermedia document, then'the multimedia object :3 said to be embedded. Any changes to the original copy of t4at object are not reflected in the embedded copy. When the hypermedia document is copied, the multimedia object is transferred with it to the new locations.

Graphics and images can be inserted in a rich-text document on embedded using such techniques as OLE Voice and audio components can be included in a text message; or they cim be part of a full voice-recorded message that has embedded text and other components.

## 5.4 Creating Hypermedia Messages

Hypermedia message is a complex collection of a variety of objects.

It is an integrated message consisting of text, rich text, binary files, images, bitmaps. voice and sound, and full motion video. Creating of a hypermedia message requires some preparation. A hypermedia report is more complex. It requires the following steps:

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

The planning phase for preparing the hypermedia message consists of determining the various sources of input. These can include any of the following:

1. A text report prepared in a word-processing system.
2. A spreadsheet in a spreadsheet program.
3. Some diagrams from a graphics program. 4 .. Images of documents.
4. Sound dips.
5. Video clips.

We should determine which components are required for the message, in what sequence should they be, and where in the text report they should be referenced. The length of each component should be determined. Careful planning is necessary to ensure that the capabilities of the messaging system are used appropriately.

Each component must be created using the authoring tool provided by the application used for creating it. All applications Involved in creating various components must have common formats to allow combining these various components. The various components must be authored, reviewed, and edited as needed, checked for smooth flow when the user launches an embedded object and stored in the final format in which it will become a part of the hyperrnedia message. The final step in this process is mailing the hypermedia message.

## 5.5 Integrated Multimedia Message Standards

Let us review some of the Integrated Multimedia MessageStandards in detail.

### Vendor Independent Messaging (VIM)

VIM interface is designed to facilitate messaging between VIM. enabled electronic mail systems as well as other applications. The VIM interface makes mail and messages services available through a well defined interface.

A messaging service enables its clients to communicate with each other in a store-and-forward manner. VIM-aware applications may also use one-or-more address books.

Address books are used to store information about users, groups, applications, and so on. **VIM Messages:**

VIM defines messaging as a stored-and-forward method of application-to-application all program-to-program data exchange. The objects transported by a messaging system are caIled messages. The message, along with the address is sent to the messaging system. The messaging system providing VIM services accept the responsibility for routing and delivering the message to the message container of the recipient.

**Message Definition:**

Each message has a message type. The message type defines the syntax of the message and the type of information that can be contained in the message.

A VIM message consists of message header. It may contain one or more message items. The message header consists of header attributes: recipient address, originator address, time/date prior

A message item is a block of arbitrary-sized (means any size) data of a defined type. The contents of the data block are defined by the data-item type.

The actual items in a message and its syntax and semantics are defined by the message type. The message may also contain file attachments. VIM allows the nesting of messages; means one message may be enclosed in another message.

A VIM message can be digitally signed so that we can ensure that the message 'received is without any modification during the transit.
**Mail Message:** It is a message of a well-defmed type that must include a message header and may include note parts, attachments, and other application-defined components. End users can see their mail messages through their mail programs.
**Message Delivery**: If message is delivered successfully, a delivery report is generated and send to the sender of the message if the sender requested the d~livei-y report. If a message is not delivered, a non-delivered report is sent to the sender.
A message that delivered will be in a message container will be marked as 'unread', until the recipient open and read it.
**Message Container**: Multiple users or applications can access one message container. Each message in a message container has a reference number associated with it for as long as the message remains stored in the message container.
**VIM Services:** The VIM interface provides a number of services for creating and mailing a message. Some of them are:
.:. Electronic message composition and submission.
.:. Electronic message sending and receiving.
.:. Message extraction from mail system.
.:. Address book services.

## MAPI Support (Multimedia Application Programmable Interface)

MAPI provides a layer of functionality between applications and underlying messaging systems. The primary goals of MAPI are: Separate client applications from the underlying messaging services. Make basic mail enabling a standard feature for all applic·ations. Support message-reliant workgroup applications.
**MAPI Architecture:** MAPI Architecture provides two perspectives (i) A client API
(ii) A service provider interface. The Client API provides the link between the client applications andMAPI. The service provider interface links MAPI to the messaging system.
The two interfaces combine to provide an open architecture such that any messaging application can use any messaging service that has a MAPI driver. MAPI drivers are provided by microsoft or third party developers.

## Telephony API (TAPI)

TAPI standard has been defined by Microsoft and Intel. The telephone can be used for reading e-mail as well as for entering e-mail messages remotely.

## X 400 Message Handling Service

The CCITT X 400 series recommendations define the OSI message handling system, (MHS).
The MHS describes a functional model that provides end users the ability t6 send and receive electronic messages. In the as I, an end user is an originator. He composes and sends messages.
Receiver is the one who receives messages. A User Agent (UA) is an entity that provides the end user function for composing and sending messages and for delivering messages. Most user agent implementations provide storage of mail, sorting directories, and forwarding.
A Message Transfer Agent (MTA) forwards messages from the originator UA to another MT A. A number of MTAs are combine to form Message transfer System (MTS).
The MTAs in an MTS provide message routing services at intermediate nodes in a WAN.
Figure below shows the overall X 400 architecture and the relationships between the components.

### X·500 Directory System Standards

The X·500 is the joint International Standard Organization
CCITT standard for a distributed directory system that lets users store information such as addresses and databases on a local server and easily query, exchange, and update that information in an interoperable networked environment.
The X 500 directory structure is described in the CCITT standard known as Data Communications Network Directory, Recommendations X·500-X·521, 1988.

### 5 X·500 Directory System Architecture

Directory System Agents carryout updates and management operations. X ·500 defines a structured information model, an objectoriented model and database schema.

The X ·500 architecture is based on a number of models, as follow'>:
**The information model**: It specifies the contents of directory entries, how they are identified, and the way in which they are organized to form the directory information base.

**The Directory model**: It describes the directory and its users, the functional model for directory operation, and the organization of the directory. .
**The security model**: It specifies the way in which the contents of the directory are protected from unauthorised access and authentication methods for updates.
The X 500 directory system is designed to be capable of spanning national and corporate boundaries.
**X 500 Directory System Components**: All information in an X 500 database is organized as entries in the Directory-Information Base(DlB). The directory system provides agents to manipulate entries in the DIB.
X 500 directories consist of the following basic components:

1. **Diretory Information Base (DIB**); The DIB contains information about users, applications, resources and the configuration of the directory that enables servers to locate one another.

2. **Directory User Agents (DUA**): A DUA issues inquiry and update requests, and accesses directory information through the directory access protocol.

3.**Directory Service Agents (DSAs**): DSAs cooperate with one another to resolve user requests over a distributed network. They interact through a specialized protocol called a directory system protocol.

## 5.6 Integrated Document Management

It is for managing integrated documents.
Integrated document Management for Messaging Specialized messaging system such as Lotus Notes provide Integrated document management for messaging. The user can attach embed or link a variety of multimedia objects.
When document is forwarded to other users, all associated multimedia objects are also forwarded and avaiiable to the new receivers of the forward message.
**Multimedia Object Server and Mail Server Interactions**:
The mail server is used to store i'lll e-mail messages. It consists of a file server with mail files for each user recipient. This file server act as a mail box.
All received mail is dropped in the user's mail file. The user can review or delete these mails. When mail messages include references to multimedia objects, mail file contains only link information.

## 5.7 DISTRIBUTED MULTIMEDIA SYSTEMS

If the multimedia systems are supported by multiuser system, then we call those multimedia systems as distributed multimedia systems.
A multi user system designed to support multimedia applications for a large number of users consists of a number of system components. A typical multimedia application environment consists of the following components:

1. Application software.
2. Container object store.
3. Image and still video store.
4. Audio and video component store.
5. Object directory service agent.
6. component service agent.
7. User interface and service agent.
8. Networks (LAN and WAN).

## Application Software

The application software perfom1s a number of tasks related to a specific business process. A business process consists ofa series of actions that may be performed by one or more users.
The basic tasks combined to form an application include the following:
**(1) Object Selection** - The user selects a database record or a hypermedia document from a file system, database management system, or document server.
(**2) Object Retrieval**- The application ret:ieves the base object.
(**3) Object Component Display** - Some document components are displayed automatically when the user moves the pointer to the field or button associated with the multimedia object.
**(4) User Initiated Display** - Some document components require user action before playback/display.
**(5) Object Display Management and Editing**: Component selection may invoke a component control subapplication which allows a user to control playback or edit the component object.

## Document store

A document store is necessary for application that requires storage oflarge volume of documents. The following describes some characteristics of document stores.
**1. Primary Document Storage**: A file systems or database that contains primary document objects (container objects). Other attached or embedded documents and multimedia objects may be stored in the document server along with the container object.
**2. Linked Object Storage**: Embedded components, such as text and formatting information, and linked information, and linked components, such as pointers to image, audio, and video. Components contained in a document, may be stored on separate servers.
**3. Linked Object Management**: Link information contains the name of the component, service class or type, general attributes such as size, duration of play for isochronous objects and hardware, and software requirements for rendering.

## Image and still video store

An image and still video is a database system optimized for storage of images. Most systems employ optical disk libraries. Optical disk libraries consist of multiple optical disk platters that are played back by automatically loading the appropriate platter in the drive under device driver control.
The characteristics of image and still video stores are as follows:
(i) Compressed information (ii) Multi-image documents
(iii) Related annotations (iv) Large volumes
(v) Migration between high-volume such as an optical disk library and high-speed media such as magnetic cache storages(vi) Shared access: The server software managing the server has to be able to manage the different requirements.

## Audio and video Full motion video store

Audio and Video objects are isochronous. The following lists some characteristIcs of audio and full-motion video object stores:
(i) Large-capacity file system: A compressed video object can be as large as six to ten M bytes for one minute of video playback.Temporary or permanent Storage: Video objects may be stored temporarily on client workstations, servers PFoviding disk caches, and multiple audio or video object servers. Migration to high volume/lower-cost media. Playback isochronocity: Playing back a video object requires consistent speed without breaks. Multiple shared access objects being played back in a stream mode must be accessible by other users.

## Object Directory Service Agent

The directory service agent is a distributed service that providea directory of all multimedia objects on the server tracked by that element of the directoryy service agent.
The following describes various services provided by a directory service Agent.
(1) Directory Service: It lists all multimedia objects by class and server location.
(2) Object Assignment: The directory service agent assigns unique identification to each multimedia object.
(3) Object Status Management: The directory service must track the current usage status of each object.
(4) Directory Service Domains: The directory service should be modular to allow setting up domains constructed around groups of servers that form the core operating environment for a group of users.

(5) Directory Service Server Elements: Each multimedia object server must have directory service element that reside on either server or some other resources.

(6) Network Access: The directory service agent must be accessible from any workstation on the network.

## Component Service Agent

A service is provided to the multimedia used workstation by each multimedia component. This service consists of retrieving objects, managing playback of objects, storing objects, and so on. The characteristics of services provided by each multimedia component are object creating service, playback service, component object service agent, service agents on servers and multifaceted services means (multifaceted services component objects may exist in several forms, such as compressed Or uncompressed).

## User Interface Service Agent

It resides on each user workstation. It provides direct services to the application software for the management of the multimedia object display windows, creation and storage of multimedia objects, and scaling and frame shedding for rendering of multimedia objects.

The services provided by user interface service agents are windows management, object creation and capture, object display and playback, services on workstations and using display software. The user interface service agent is the client side of the service agents. The user interface agent manages all redirection since objects are located by a look-up mechanism in the directory service agent

## Distributed client server operation

The agents so far we have discussed combine to form a distributed client-server system for multimedia applications. Multimedia applications require functionality beyond the traditional client server architecture.

Most client-server systems were designed to connect a client across a network to a server that provided database functions. In this case, the client-server link was firmly established over the network. There was only one copy of the object on the specified server. With the development of distributed work group computing, the picture has changed for the clients and servers. Actually in this case, there is a provision of custom views in large databases. The advantage of several custom views is the decoupling between the physical data and user.

The physical organization of the data can be changed without affecting the conceptual schema by changing the distributed data dictionary and the distributed data repository.

**Clients in Distributed Work Group Computing**

Clients in distributed workgroup computing are the end users with workstations running multimedia applications. The client systems interact with the data servers in any of the following *w3fs.*

1. Request specific textual data.
2. Request specific multimedia objects embedded or linked in retrieved container objects.
3. Require activation of a rendering server application to display/ playback multimedia objects.
4. Create and store multimedia-objects on servers.

Request directory information. on locations of objects on servers

**Servers in Distributed Workgroup Computing**

Servers are storing data objects. They provide storage for a variety f object classes, they transfer objects on demand on clients. They rovide hierarchical storage for moving unused objects to optical_ isk lirbaries or optical tape libraries. They provide system dministration functions for backing up stored data. They provide le function of direct high-speed LAN and WAN server-to-server ~ansport for copying multimedia objects.

**Middleware in Distributed Workgroup Computing**

The middleware is like interface between back-end database and font-end clients.The primary role of middleware is to link back end database to front end clients in a highly flexible and loosely connected network nodel. Middleware provides the glue for dynamically redirecting client requests to appropriate servers that are on-line.

Multimedia Object Servers The resources where information objects are storedareknown as servers. Other users (clients) can share the information stored in these resources through the network.

## Types of Multimedia Servers

Each object type of multimedia systems would have its own dedicated server optimized for the type of data maintained in the object. A network would consist of some combination of the following types of servers.

**(1) Data-processing servers RDBMSs and ODBMSs. (2) Document database servers.**
**(3) Document imaging and still-video servers. (4) Audio and voice mail servers.**
**(5) Full motion video server.**

Data base processing servers are traditional database servers that contain alphanumeric data. In a relational database, data fields are stored in columns in a table. In an object-oriented database these fields become attributes ofthe object. The database serves the purpose of organizing the data and providing rapid indexed access to it. The DBMS can interpret the contents of any column or attribute for performing a search.

## Mass Storage for Multimedia Servers

**RAID(Redundant Arrays of Inexpensive Disks)**

In terms of redundancy,RAID provides a more cost effective solution than disk mirroring.

RAID is a means of increasing disk redundancy, RAID systems use multiple and potentially slower disks to achieve the same task as a single expensive large capacity and high transfer rate disk.

In RAID high transfer rates are achieved by performing operations in parallel on multiple disks. There are different levels of RAID available, namely disk striping(level 0), disk mirroring(level 1, Bit interleaving of date(level 2), Byte interleaving (level 3), sector interleaving(level 4), and block interleaving(level 5)RAID technology is faster than rewritable optical disk and high data volumes can be achieved with RAID. RAID technology provides high performance for disk reads for almost all types of applications.

## Write Once Read Many Optical Drives: (WORM)

WORM Optical drives provide very high volumes of storage for very low cost.

Some important characteristics of WORM optical disks are:

- Optical drives tend to be slower than magnetic drives by a factor of three to four. .
- WORM drives can write once only; typically 5-10% of disk capacity m left free to provide for changes to existing information.
- They are useful for recording informations that would not change very much.
- They are virtually indestructible in normal office use and have long shelf lives.
- They an be used in optical disk libraries (Juke boxes). A Juke box may provide anywhere from 50-100 disk platters with two or more drives.
- These characteristics make optical disks ideal candidates for on-line document images (which change very little once scanned and do not have an isochronous requirement) and archived data.

**Rewritable Optical Disks**:

Rewritable optical drives are produced by using the technologies like magneto-optical. It has the advantage ofrewritability over the WORM where rewritable is not possible. It can be used as primary or secondary media for storage of large objects, which are then archieved. (Placed where documents are preserved) on WORM disks.

If it is used as primary media, it should be accompanied by highspeed magnetic disk cache. This is to achieve acceptable video performance.

**Optical Disk Libraries**:

Optical disk libraries are nothing but juke boxes. Work disks and rewritables can be used in optical disk libraries to achieve very high volumes of near-lines storage. Optical disk libraries range from desk top juke boxes with one 5' 1/4" drive and I O-slot optical disk stack for upto lOG Bytes of stroage of large libraries using as many as four 12" drives with an 80-s10t optical disk stack for upto terabytes of storage. The disadvantage of optical disk library is the time taken for a platter to be loaded into a drive and span to operating speed.

**Network Topologies for Multimedia Object Servers**

A number of network topologies are available Network topology is the geometric arrangement of nodes and cable links in a network. We still study three different approaches to setting up multimedia servers.

(i) **Centralized Multimedia Server**: A centralized multimedia object server performs as a central store for multimedia objects. All user requests for multimedia objects are forwarded by the applications to the centralized server and are played back from this server. The centralized server may serve a particular site of the corporation or the entire enterprise. Every multimedia object has a unique identity across the enterprise and can be accessed from any workstation. The multimedia object identifier is referenced in every data that embeds or links to it.

**Dedicated Multimedia Servers**: This is the approach where a video server is on a separatededicated segmentIn this approach, when a workstation dumps a large video, the other servers on the networks are not affected. Provides high performance for all local operations. The isochronocity of the objects is handled quite well in a dedicated mode.
Disadvantage of this approach is that the level of duplication of objects.

## Distributed multimedia servers:

In this approach multimedia object servers are distributed in such a manner that they are placed in starategic locations on different LANs.They are replicated on a programmed basis to provide balanced serviceto all users.

## Multiserver Network Topologies

To distribute the full functionality of multimedia network wide there are vareity of network topologies available. ' The primary topologies are Traditional LANs (Ethernet or Token Ring Extended LANs (Using network switching hubs bridges and routers). ' High speed LANs (ATM and FDDI II). WANs (Including LANs, dial-up links-including ISDN T1 and $T_3$ lines-etc.). ' I

**Traditional LANS (Ethernet or Token Ring) Ethernet:**

**Ethernet**: It is a Local Area Network hardware, communication, and cabling standard originally developed by Xerox corporation that link up to 1024 nodes in a bus network. It is ahigh speed standard using a baseband (single-channel) communication technique. It provides for a raw data transfer rate of 10 Mbps, with actual throughput in the range of 2-3 Mbps. It support a number of sessions ina mixof live video, audio electronic mail and so on.

**Token Ring**: It is a Local Area Network architecture that combines token passing with a hybrid star/ring topology. It was developed by IBM. Token Ring Network uses a multistation Access unit at its hub ..

**ATM (Asynchronous Transfer Mode)**

It is a network architecture that divides messages into fixed size units (called cells) of small size and that establishes a switched connection between the originating and receiving stations.
A TM appears to be a potential technology for multimedia systems for connecting object servers and user workstations. ATM is actually a good candidate for two reasons: as a hub and spoke teclmology, it adapts very well to the wiring closest paradign; and it allows workstations to operate at speeds defined by the workstation. Figure 5.12 below illustrates LAN topology using an A TM Switching System.

**FDDI II (Fiber Distributed Data Interface II)**

It is a standard for creating highspeed computer networks that employ fiber-optic cable. FOOl II operates exactly like token ring, . with one difference: FOOl employs two wires through all the hosts in a network.
FOOl II is a single media LAN and its full bandwidth supports all users.
FOOl II appears to be a very useful high-speed technology for connecting servers on an additional separate network and providing the dedicated high bandwidth necessary for rapid transfer and replication of information objects. Figure 5.13 shows a multiievel network based

**WANS (Wide Area Network)**

This includes LANs, dial up ISDN, T1 (1.544 Mbits/sec) and T3 (45.3 Mbits/sec) lines and regular telephone dial-up lines. The two big issues here are:

- :. W ANs may have a mix of networking and communication protocols.
- :. WAN has a variety of speeds at which various parts of it where it communicates.

Protocol Layering: Layering helps to isolate the network from the . application. Layering of protocols started with the release ofthe ISO model.

## Distributed Multimedia Databases:

A multimedia database consists of a member of different types of multimedia objects. These may include relational database records, object-oriented databases with objects for alphanumeric attributes, and s:orage servers for multimedia objects such &s images, still video, audio, and full-motion video. It is feasible to include an image or a video object as a binary large object (BLOB) in a relational database. It is also feasible to include such an object as an attribute in an object.

## Database Organization for Multimedia Applications

Optical disk storage technology has reduced the cost of multimedia document storage by a significant factor. Distributed architectures have opened the way for a variety of applications distributed around a network accessing the safe database in an independent manner. The following discussion addresses some key issues of the data organization for multimedia systems.

**Data Independence:** Flexible access to a variety of distributed databases for one or more applications requires that the data be independent from the application so that future applications can access the data without constraints related to a previous application. Important features of data independent design are:

1. Storage design is independent of specific applications.
2. Explicit data definitions are independent of application programs.
3. Users need not know data formats or physical storage structures .
4. Integrity assurance is independent of application programs.
5. Recovery is independent of application programs .

**Common Distributed Database Architecture:** Employment of Common Distributed database architectured is presented by the insulation of data from an application and distributed application access.

Key features of this architecture are:

1.The ability for multiple independent data structures to co-exist
in the system (multiple server classes).
2.Uniform distributed access by clients.
3.Single point for recovery of each database server.
4.Convenient data re-organization to suit requirements.
5.Tunability and creation of object classes.
6.Expandibility.

Mnltiple Data Servers: A database server is a dedicated resource on a network accessible to a number of apphcations, When a large number of users need to access the same resources, problem arises
This problem is solved by setting up multiple data servers that have copies of the same resources,

**Transaction management for Multimedia Systems**

It is defined as the sequence of events that starts when a user makes a request to create, render, edit, or print a hypermedia document. The transaction is complete when the user releases the hypermedia document and stores back any edited versions or discards the copy in memory or local storage.

Use of object classes provides an excellent way for managing and tracking hypermedia documents, Given that all components of a hypermedia document can be referenced within an object as attributes, we can find a solution for the three-dimensional transaction management problem also in the concept of objects.

Andleigh and Gretzinger expand on the basic concepts developed for the object request broker (ORB) by the Object Management Group (OMG) and combine it with their transaction management approach.

**Managing Hypermedia Records as Objects**

Hypermedia records or documents are complex objects that contain multimedia information objects within them, A hypermedia document can be stored in a document data base, as a BLOB in a relational database, or in an object-oriented data base. A Hyper media document may contain multimedia objects embedded in it as special fields.

**Object linking and embedding:** OLE provides an object oriented framework for compound documents. When a user double cliks or click on an icon for an embedded object, the application that created the object starts, and allows the user to view andor the object .

**Managing Distributed Objects:**We see the nature of communication between servers and the managing of distributed objects.

**Interserver communications:** Object replication , object distribution, object recompilation and object management and network resources are some of the design requirements that play a role in defining interserver The following lists the types of communications that one server may make to another server:

1. Obtain a token from an object name server for creating a new multimedia object; the object is not accessible by others users until complete and released.

2. Search the object class directory for the current locations of that object and the least expensive route for accessing it.
3. Perform a shared read lock on the object to ensure that it is n archived or purged while it is being retrieved.
4. Replicate a copy of the object; update the object name server directory.
5. Copy an object for non-persistent use.
6. Test and set an exclusive lock on an object for editing purposes'
7. create new versions.
8. Pause the retrieval of an object to support a user action or to pace the retrieval to the speed supported by the network.
9. A Sound server architecture is necessary for providing these services in a fully distributed environment.

**Object Server Architecture**

**Figure** describes an object server architecture that can support multimedia applications for a large number of users.

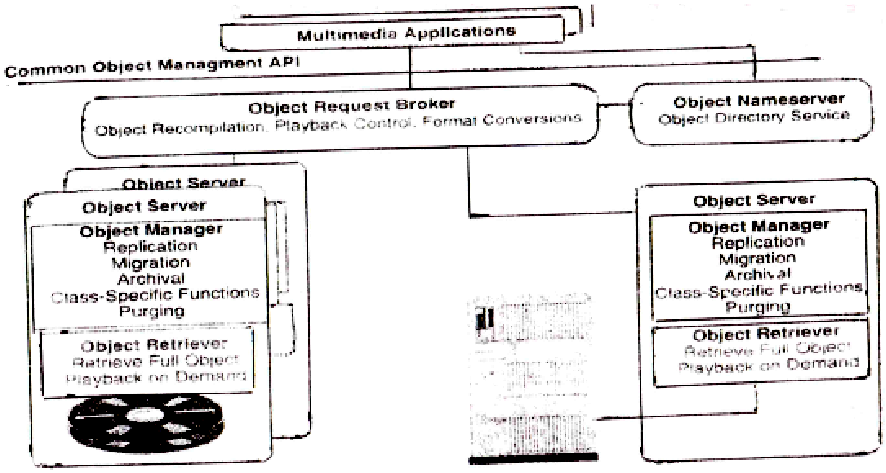



**Fig. 5.14: Object Server Architecture**

The architecture describes the logical distribution of functions. The following lists the key elements of this architecture:

Multimedia Applications -Common Object Management API.Object Request Broker. Object Name Server -Object Directory Manager -object Server -Object Manager.Network Manager . Object Data Store. Any multimedia application designed to operate on the common object management API can function in this architecture

The common object management API is a programming interface definition that provides a library of functions the applications can call.

The common Broker Architecture API provides a uniform interface to all applications and a standardized method for managing all information objects in a corporate network.

A common Object Request Broker Architecture (CORBA) h been defined by a Object Management Group. An object request broker performs the following functions:

**(i) Object recompilation.**
**(ii) Playback control.**
**(iii) Format conversions.**

The object name server provides an object directory service. The object directory manager may exist in a distributed form within an object server. The object directory manager updates the object directory when changes take place.The object server is a logical subsystem in the network responsIble for storing and retrieving objects on demand. The object manager consists of a number of object classes that performs a number of specialized services. They are: Object retrieval. (ii) Replication(Ui) Migration. (iv) Transaction and Lock Management. (v) User Preferen'ce. (vi) Versioning. (vii) System Administration. (ix) Archival. (x) Purging. (xi) Class-Specific functions.

**Identification method:** Objects can be distinguished from one another in many potential ways. Identification of objects in a persistent state is different from non-persistent objects. At the highest level, persistent objects are distinguished by the class of objects. Andleigh and Gretzinger defined a rule for unique object identification as follows:

**ROLE:** An object must have an identifier that is unique in a time dimension as well as with location such that it cannot be modified by any programmed action. An alternative approach is to divide the network into domains and have a name server in each domain be responsible for assigning new object IDs for all objects created in that domain. An object identification algorithm can be made unique by combining several of the following components.

.:. Network domain name. --Address and server ID of the name server node.--A time stamp of creating time.-- An object class identifier.

**Object Directory services**

A multimedia object directory manager is the name server for all multimedia objects in a LAN. It has an entry for every multimedia object on all servers on the LAN, or in a domain if a LAN or WAN is subdivided into domains. The object directory manager manages changes to the object directory resulting from object manager actions.

**Multimedia Object Retrieval**

The multimedia object manager performs the functions of managing all requests from the multimedia applications for retrieving existing multimedia objects *01'* storing new or edited multimedia objects created by the user. In systems actively designed using an object request broker, this request is channeled through the object request broker. Data structure maintained by the multimedia object manager:

**Database Replication Techniques** In the simplest fonn of data management, the databases are set up as duplicates of the databases. Database duplication ensures that the multiple copies are identical.

There is an approach to allow each copy of the database to be modified as needed and to synchronize them by comparing them and copying the changes to all other database copies on a very frequent basis, this process is called replication.

**Types of Database Replication**: There are eight types of modes available. They are: Round Robin replication.2.Manual replication. (Hi) Scheduled replication. (iv) Immediate replication. V)Replication-on-demand. Vi) Predictive replication. Vii) Replication references. Viii)No replication. **Object Migration Schemes**

**Optimizing Object Storage** A number of techniques are available for optimizing data storage for multimedia objects. Let us consider the three design approaches

1. **Optimizing Servers by Object Type:**
   The mechanism for optimizing storage is to dedicate a server to a particular type of object. The object server may ne designed to provide specialized services for specific object classes related to rendering
2. **Automatic Load Balancing:** It can be achieved by programming the replication algorithm to monitor use counts for each copy of a replicated object.
3. **Versioned Object Storage:**
   The storage problem will be more complex if multiple vcrsions need to be stored. Hence, we should follow the technique which is based on saving changes rather than storing whole new objects. New versions of the object can be complex objects,.

**C 3160**

B.E./B.Tech. DEGREE EXAMINATION, MAY/JUNE 2007.

Sixth Semester

CS 1354 — COMPUTER GRAPHICS AND MULTIMEDIA SYSTEM

(Regulation 2001)

Time : Three hours Maximum : 100 marks

Answer ALL questions.

PART A — (10 × 2 = 20 marks)

1. Explain the disadvantages of DDA line drawing algorithm.

2. Define vanishing point with an example.

3. Define morphing.

4. Define view port with an example.

5. Define Multimedia.

6. List the components of Multimedia.

7. List any four hardware requirement of Multimedia.

8. What is MIDI?

9. List 3 multimedia authoring tool.

12. (a) Explain 3D basic transformation with an example. (16)

Or

(b) (i) Design a storyboard layout and accompanying key frames for an animation of a single polyhedron. (9)

(ii) How to specify objects motion in an animation system. (7)

13. (a) (i) List the Multimedia application. Explain them briefly. (12)

(ii) Briefly discuss the history and future of Multimedia. (4)

Or

(b) (i) Explain the characteristics of MDBMS. (10)

(ii) Write short notes on multimedia system architecture. (6)

14. (a) List the types of fixed and removable storage devices available for multimedia, and discuss the strengths and weakness of each one. (16)

Or

(b) Explain the data compression technique used in multimedia. (16)

15. (a) (i) Distinguish between multimedia system and hypermedia system. (4)

(ii) List the main attribute, benefits and drawbacks of 3 types of authoring systems. (12)

Or

(b) Write short notes on the following :

(i) Mobile messaging. (8)

(ii) Document management. (8)

**T 3187**

B.E./B.Tech. DEGREE EXAMINATION, APRIL/MAY 2008.

Sixth Semester

(Regulation 2004)

Computer Science and Engineering

CS 1354 — GRAPHICS AND MULTIMEDIA

(Common to B.E. (Part-Time) Fifth Semester Regulation – 2005)

Time : Three hours Maximum : 100 marks

Answer ALL questions.

PART A — (10 x 2 = 20 marks)

1. What is 'Scan Conversion'?

2. How do you correct the shape of line ends for very thick and inclined lines?

3. How are fractals classified?

4. What does Y, I, Q represent in YIQ color model?

5. Name a few analog and digital video broadcast standards.

6. What kind of database management systems are preferred for multimedia data?

7. Calculate the file size in MB for a 1 minute stereo recording of CD quality music at 44.1 KHz sampling rate and 16 bit resolution.

8. What do you understand by refresh rates of *a* monitor and how do you measure it?

9. What are common navigation modes?

10. Compare the performances of High speed LAN's for multimedia data transfer.

PART B — (5 x 16 = 80 marks)

11. (a) (i) Derive Bresenham's algorithm for lines with slope magnitudes>1. (8)

(ii) Use the above algorithm to find all the points on a triangle in the first quadrant with vertices at (0, 2), (6, 2), (3, 6). (8)

Or

(b) (i) Find the reflection of the point (2, 4) with respect to the line $y = x + 1$. (8)

(ii) Apply Cohen Sutherland line clipping algorithm to clip a line with end points (1, 7) and (7, 5) against a window with boundaries $xw_{min} = 2$, $xw_{m}$ $= 6$, $yw_{min} = 2$, $xw_{max} = 6$. (8)

12. (a) (i) Discuss the characteristics of Bezier curves and Bezier surfaces in detail. (12)

(ii) Write a short note on B-Spline curves. (4)

Or

(b) (i) Depict and discuss the HSV color model in detail. (8)

(ii) How do you implement morphing animation technique? Discuss with an example. (8)

13. (a) (i) Explain the multimedia systems architecture in detail. (8)

(ii) What are the basic objects of multimedia? (8)

Or

(b) Write a detailed note on the predominant areas of multimedia applications. (16)

14. (a) (i) Discuss the CCITT group of compression standards in detail. (8)

(ii) Explain the TIFF file format. (8)

Or

(b) (i) Describe the operation of a pen Input system. (8)

(ii) Draw the video capture board architecture and write a short note on the functions of each component. (8)

15. (a) Discuss the types of multimedia authoring systems. (16)

Or

(b) (i) What are the components of a typical hypermedia message? (8)

(ii) What is hypermedia linking and embedding? Explain. (8)

---

T 3187

B.E./B.Tech. DEGREE EXAMINATION, APRIL/MAY 2008.

Sixth Semester

(Regulation 2004)

Computer Science and Engineering

CS 1354 — GRAPHICS AND MULTIMEDIA

(Common to B.E. (Part-Time) Fifth Semester Regulation – 2005)

Time : Three hours                                                  Maximum : 100 marks

Answer ALL questions.

PART A — (10 x 2 = 20 marks)

1. What is 'Scan Conversion'?

2. How do you correct the shape of line ends for very thick and inclined lines?

3. How are fractals classified?

4. What does Y, I, Q represent in YIQ color model?

5. Name a few analog and digital video broadcast standards.

6. What kind of database management systems are preferred for multimedia data?

7. Calculate the file size in MB for a 1 minute stereo recording of CD quality music at 44.1 KHz sampling rate and 16 bit resolution.

8. What do you understand by refresh rates of *a* monitor and how do you measure it?

9. What are common navigation modes?

10. Compare the performances of High speed LAN's for multimedia data transfer.

PART B — (5 x 16 = 80 marks)

11. (a) (i) Derive Bresenham's algorithm for lines with slope magnitudes>1. (8)

(ii) Use the above algorithm to find all the points on a triangle in the first quadrant with vertices at (0, 2), (6, 2), (3, 6). (8)

Or

(b) (i) Find the reflection of the point (2, 4) with respect to the line $y = x + 1$. (8)

(ii) Apply Cohen Sutherland line clipping algorithm to clip a line with end points (1, 7) and (7, 5) against a window with boundaries $xw_{min} = 2$, $xw_{} = 6$, $yw_{min} = 2$, $xw_{max} = 6$. (8)

12. (a) (i) Discuss the characteristics of Bezier curves and Bezier surfaces in detail. (12)

(ii) Write a short note on B-Spline curves. (4)

Or

(b) (i) Depict and discuss the HSV color model in detail. (8)

(ii) How do you implement morphing animation technique? Discuss with an example. (8)

13. (a) (i) Explain the multimedia systems architecture in detail. (8)

(ii) What are the basic objects of multimedia? (8)

Or

(b) Write a detailed note on the predominant areas of multimedia applications. (16)

14. (a) (i) Discuss the CCITT group of compression standards in detail. (8)

(ii) Explain the TIFF file format. (8)

Or

(b) (i) Describe the operation of a pen Input system. (8)

(ii) Draw the video capture board architecture and write a short note on the functions of each component. (8)

15. (a) Discuss the types of multimedia authoring systems. (16)

Or

(b) (i) What are the components of a typical hypermedia message? (8)

(ii) What is hypermedia linking and embedding? Explain. (8)