MOHAMED SATHAK A J COLLEGE OF ENGINEERING

ME 8691 -COMPUTER AIDED DESIGN AND MANUFACTURING

K.K.VINOTHKUMAR M.E (Ph.D)
Assistant Professor,
Department of Mechanical Engg.,
MSAJCE

CAD

Computer Aided

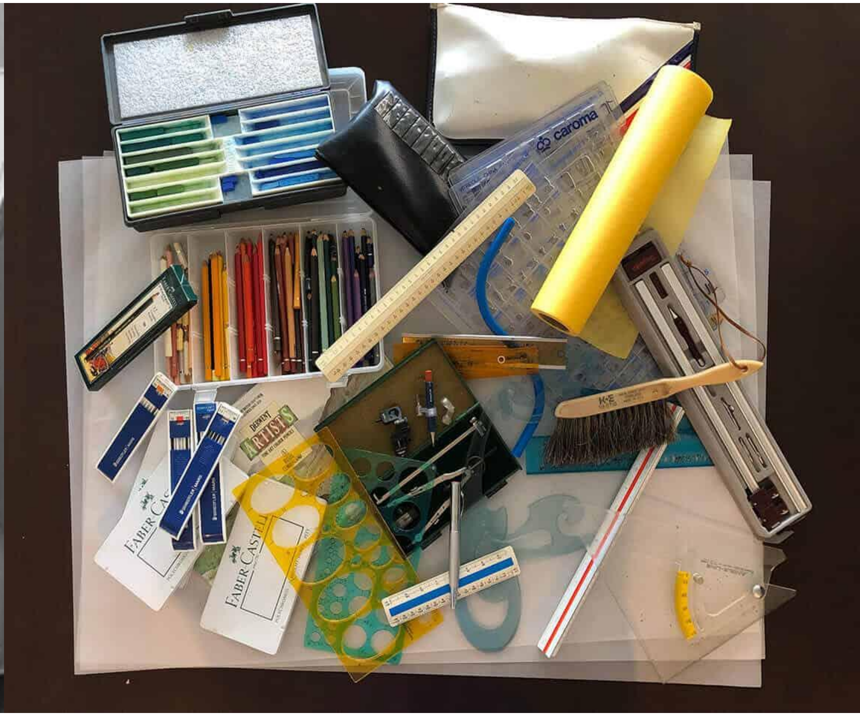
Design or Drafting

Difference between Drafting

DESIGN VERSUS DRAFTING

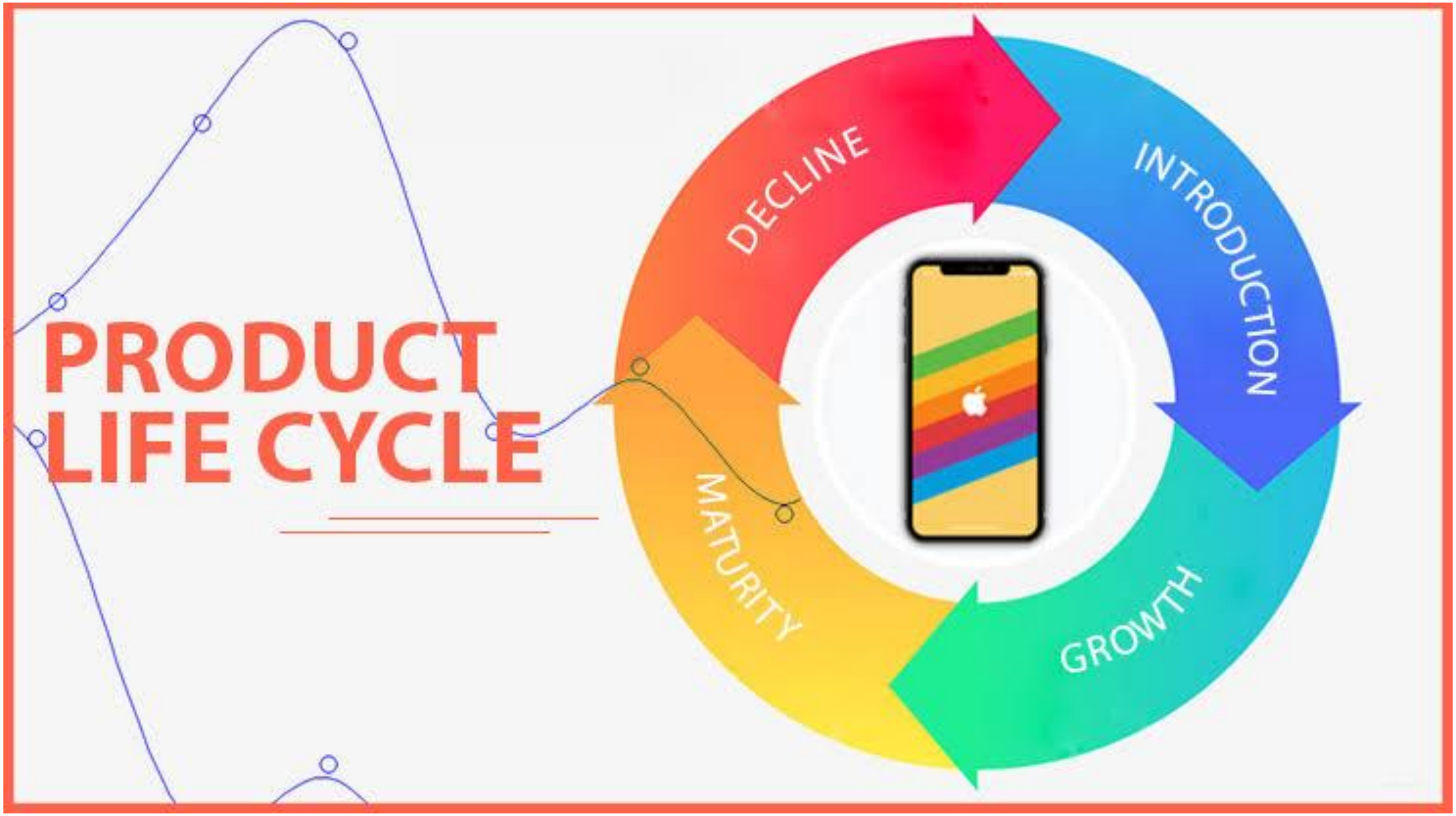
Design	Drafting
In architecture and engineering, design is the process of transforming ideas into life.	Drafting is the technique of creating technical drawings, whether in 2D or 3D.
Design means creating sketches or basic drawings with technical data which is a pictorial representation of the building.	Drafting requires creating technical drawings that will provide the technical specifications of the architectural project.
Design is the process which involves conception and 3D modeling to bring ideas into life using computer-aided drawings.	It's part of the design process that is done by hand or using computer-aided programs and mechanical drawings.
It provides detailed analysis to justify the selection of component sizes based on various factors.	Drafters use AutoCAD program to draft plans and create technical drawings.

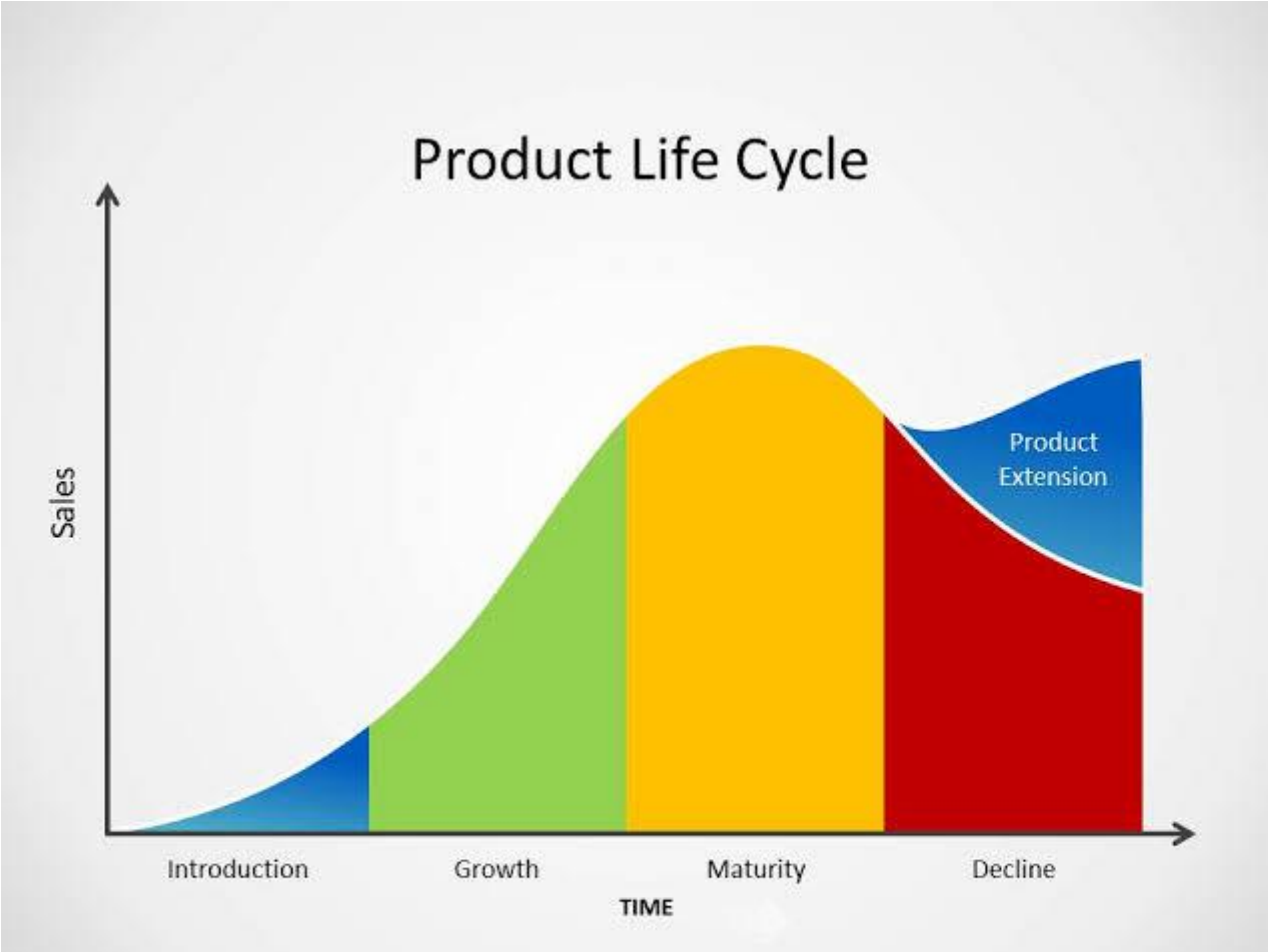
Manual Vs Computer Aided



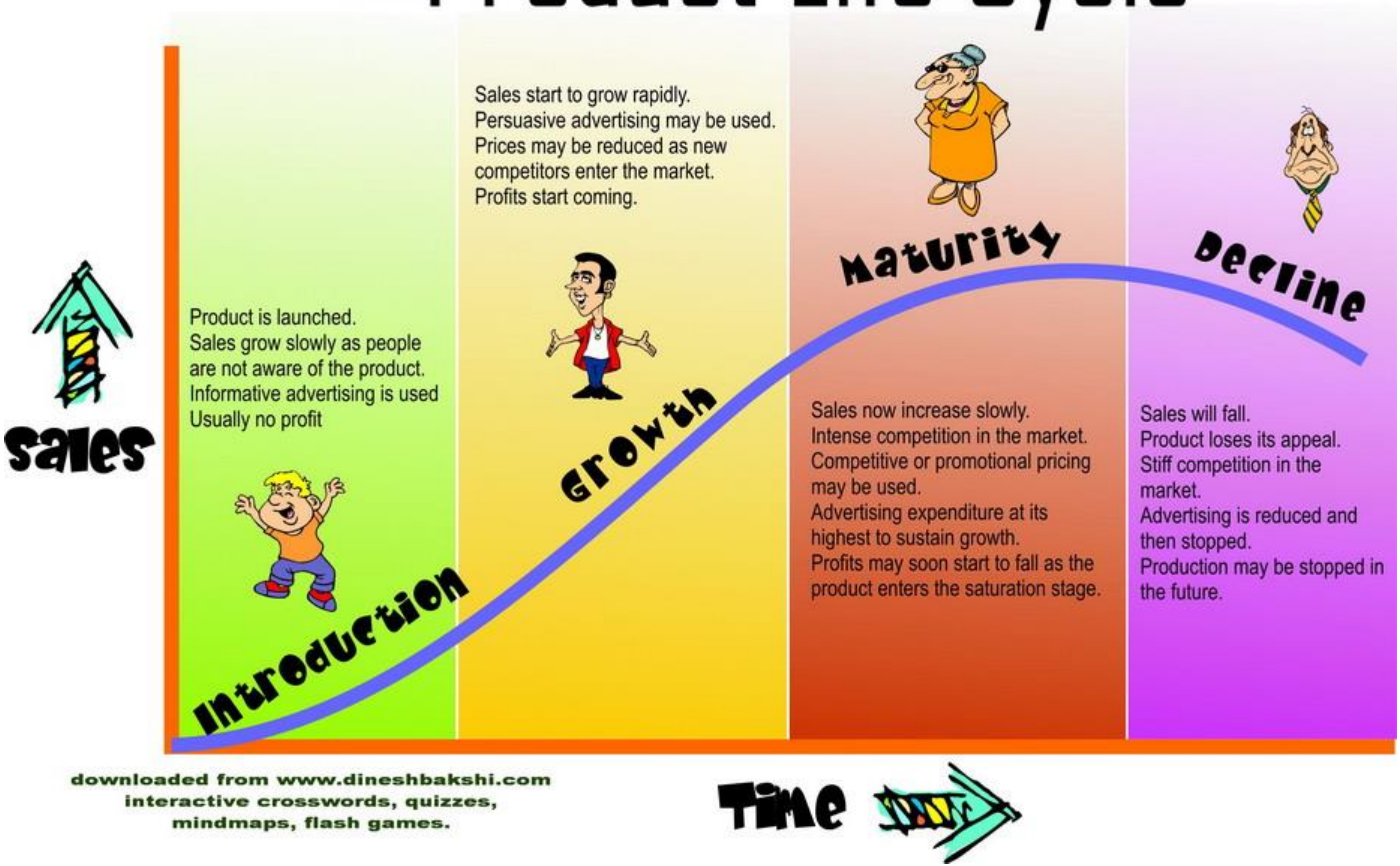
Unit -1 INTRODUCTION

Product cycle- Design process- sequential and concurrent engineering- Computer aided design – CAD system architecture- Computer graphics – coordinate systems- 2D and 3D transformations homogeneous coordinates – Line drawing -Clipping- viewing transformation-Brief introduction to CAD and CAM – Manufacturing Planning, Manufacturing control- Introduction to CAD/CAM – CAD/CAM concepts ––Types of production – Manufacturing models and Metrics – Mathematical models of Production Performance

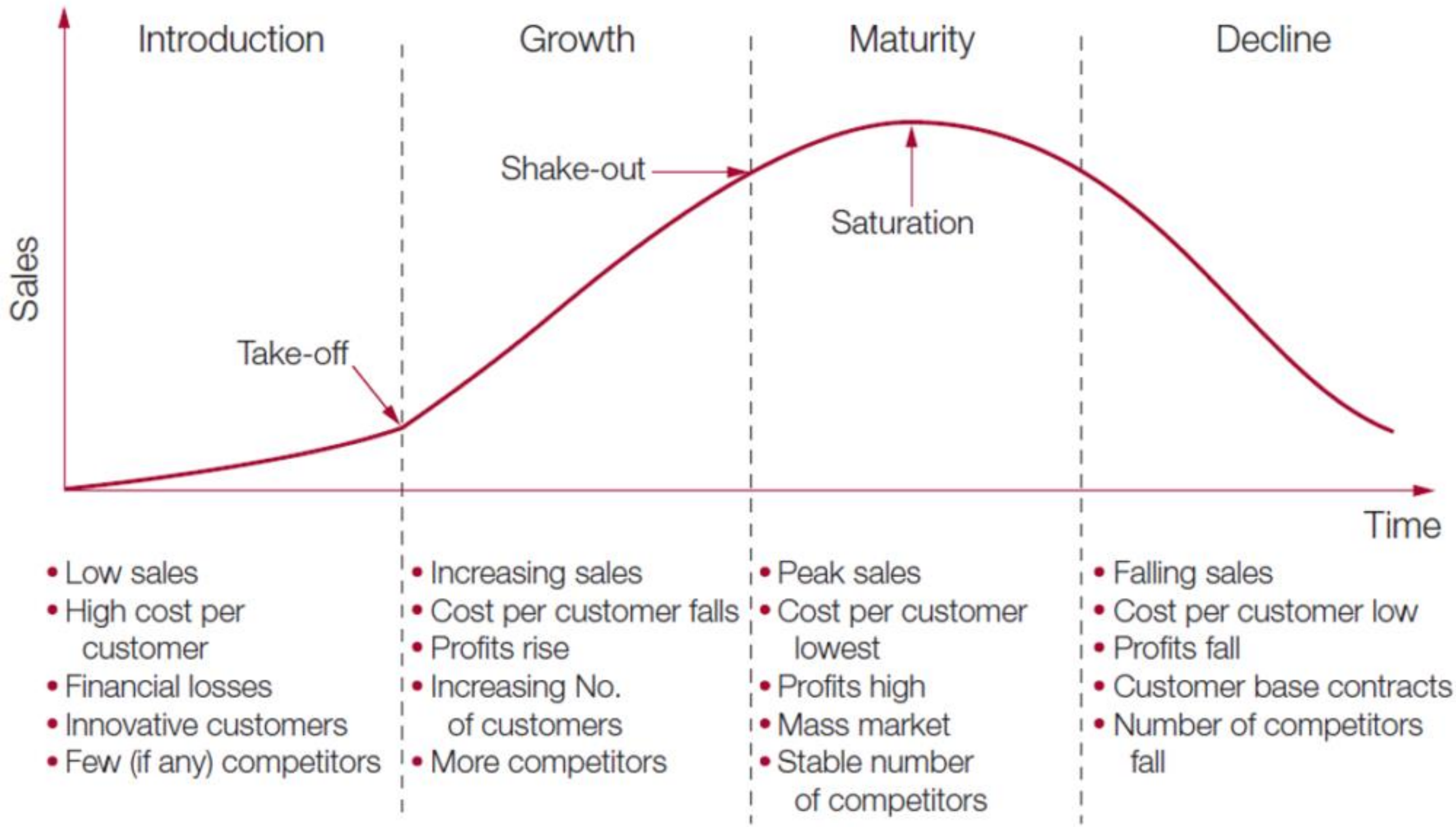

PRODUCT
LIFE CYCLE
DECLINE
INTRODUCTION
GROWTH
MATURITY

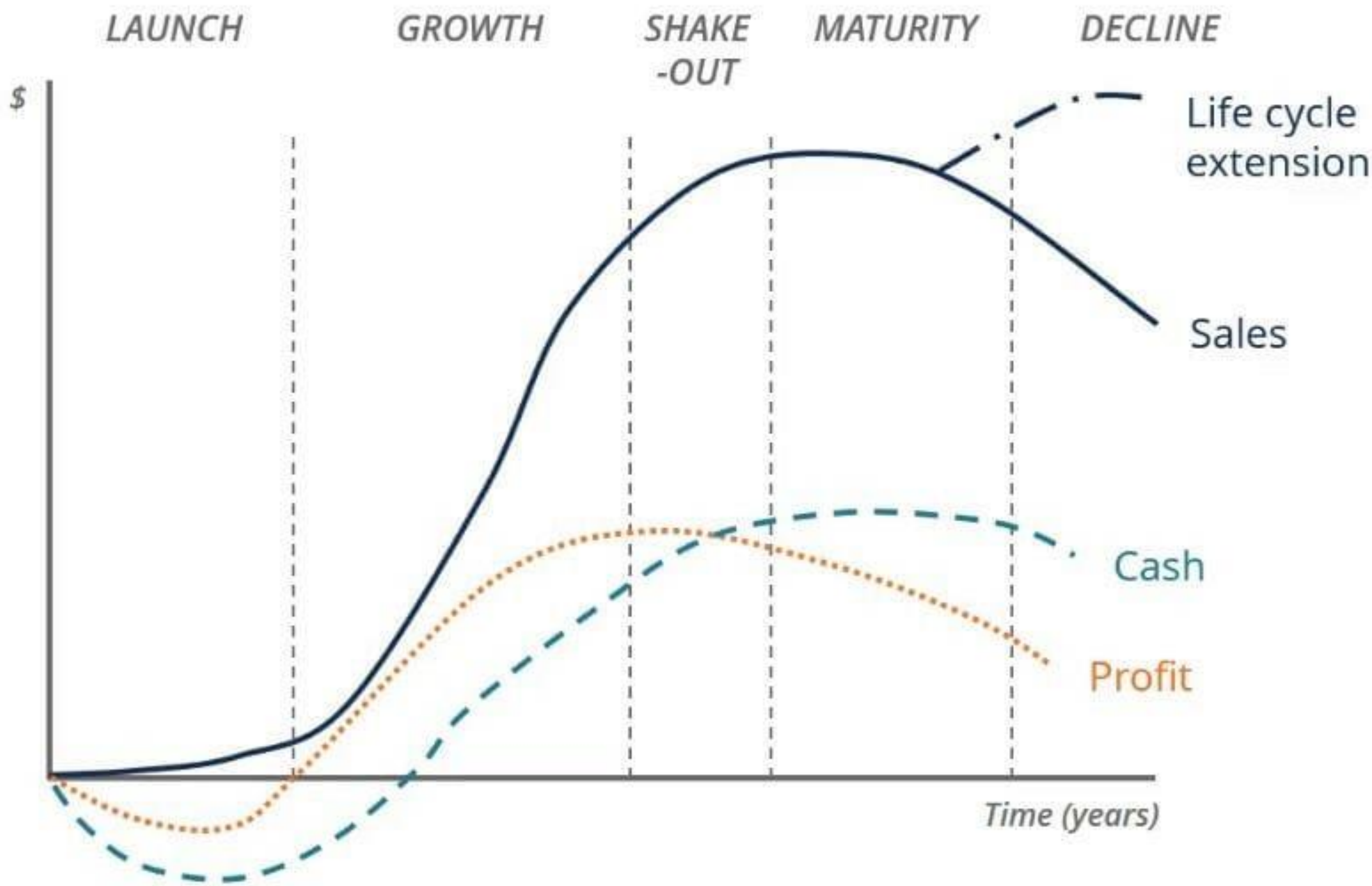

Product Life Cycle
Sales
Product
Extension
Introduction
Growth
Maturity
Decline
TIME

Product Life Cycle



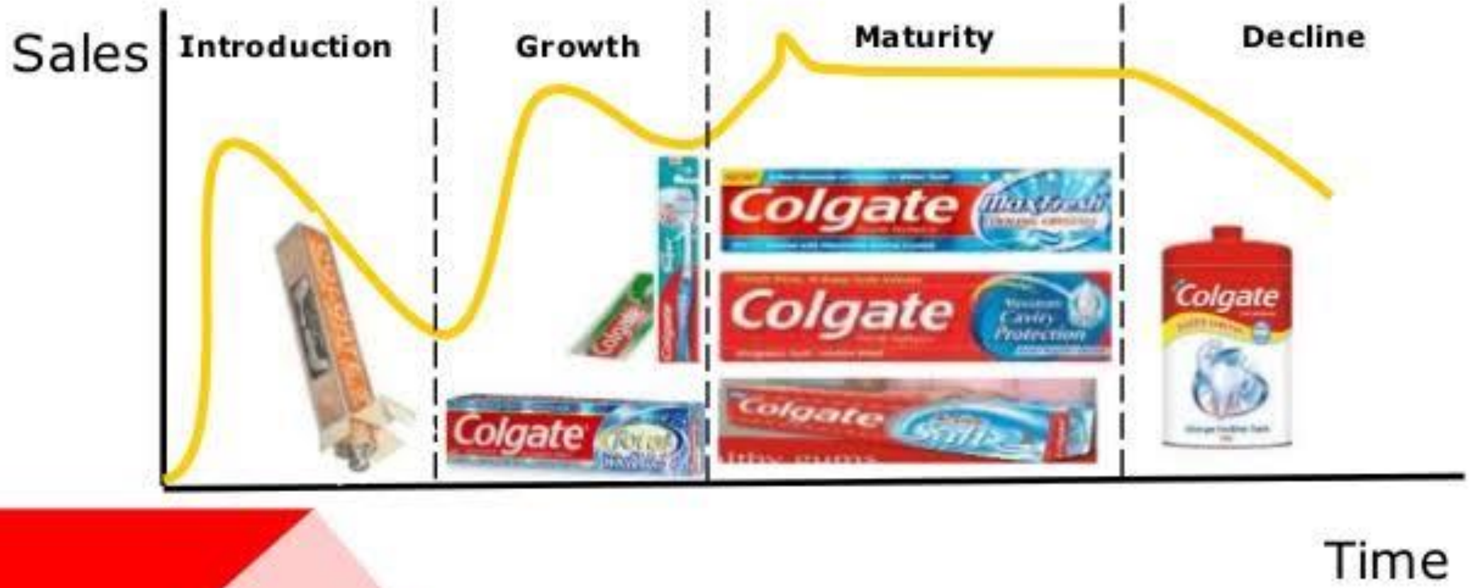
The 4 Life Cycle Stages and their Marketing Implications





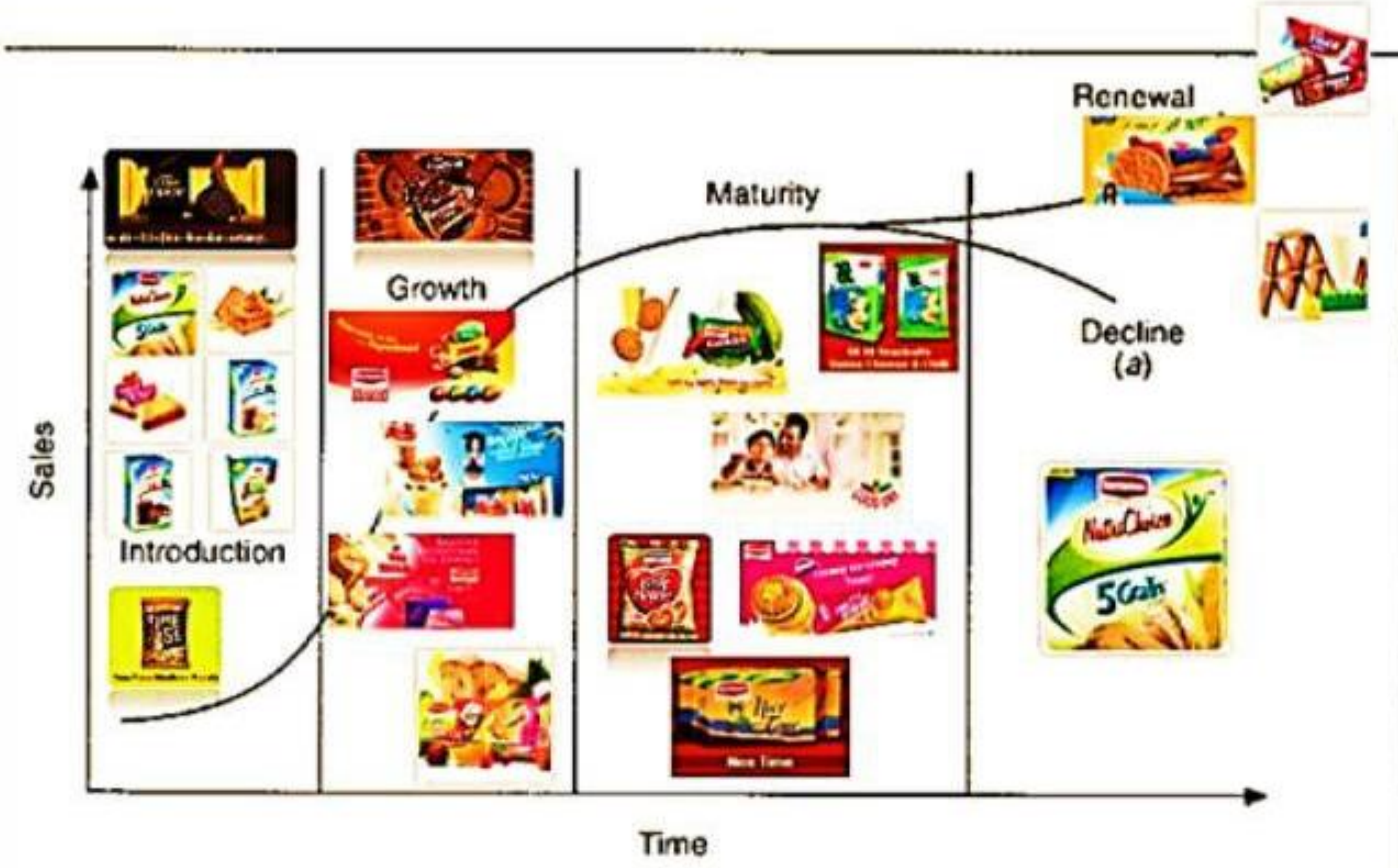
LAUNCH
GROWTH
SHAKE -OUT
MATURITY
DECLINE
$
Life cycle extension
Sales
Cash
Profit
Time (years)

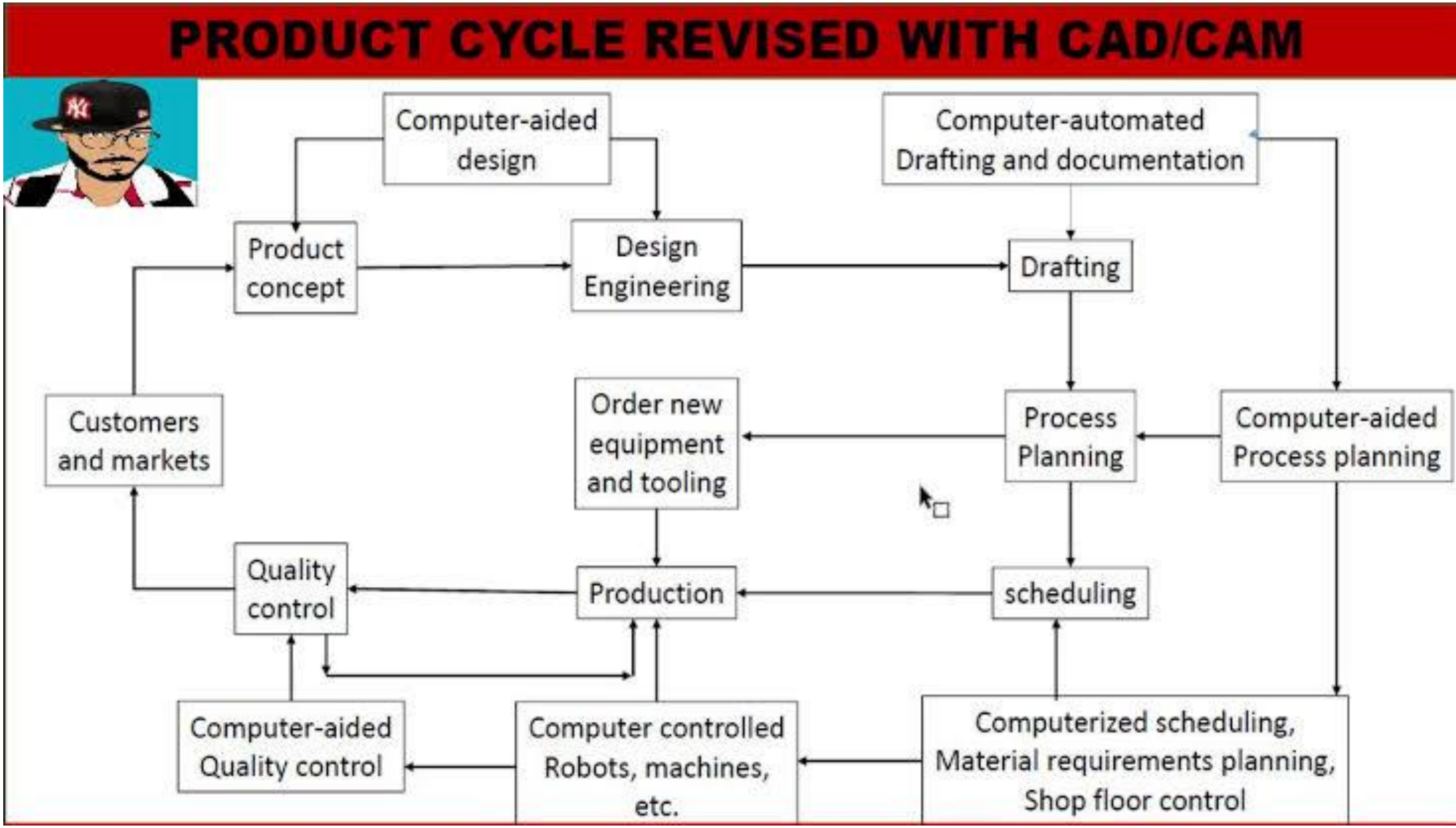
PRODUCT LIFE CYCLE

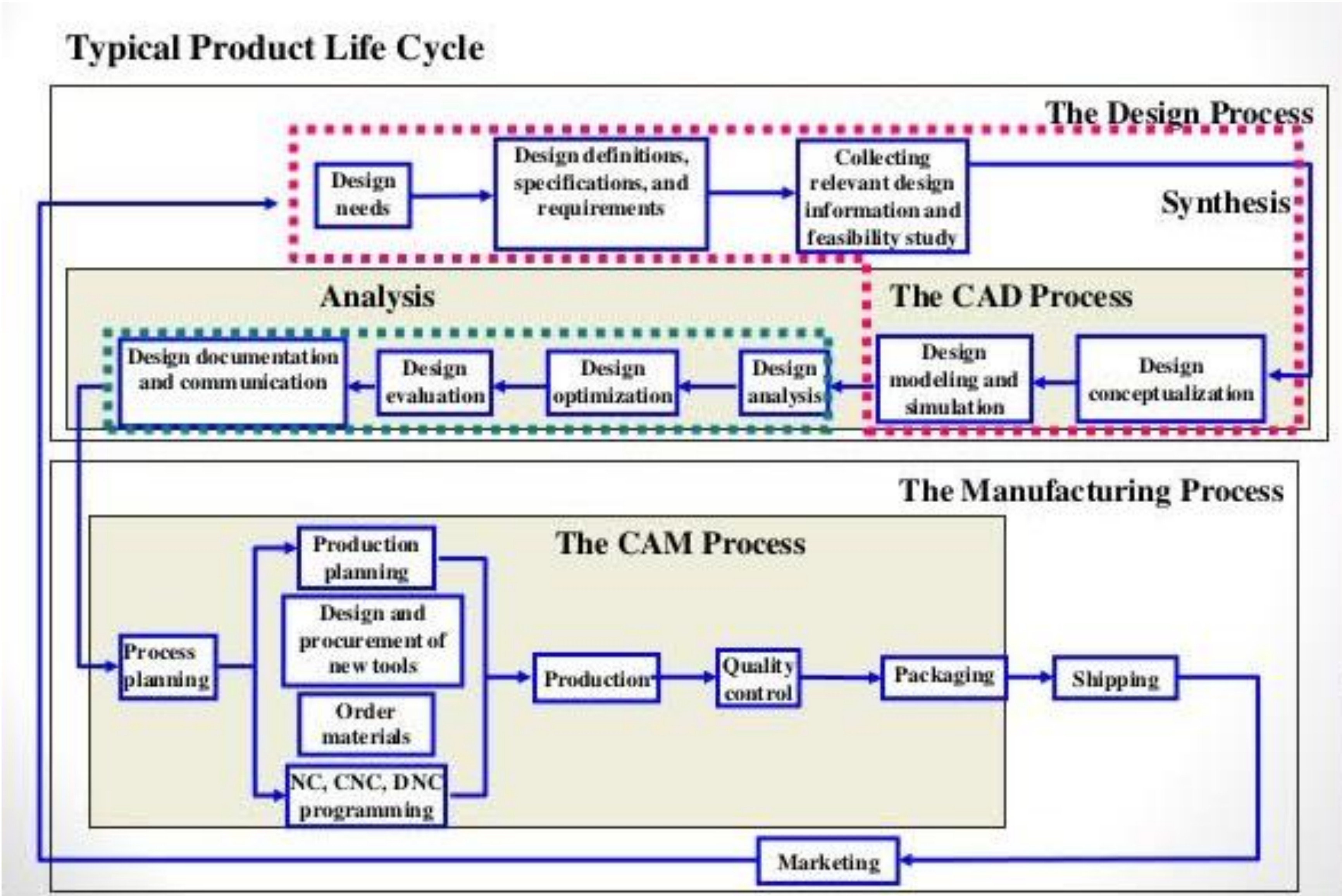



NEW YORK
pepsi
1898
Original
Introduced
NEW YORK
NOW ASPARTAME FREE
diet
pepsi
1964
Diet Pepsi
Added
1893
FROM THE MAKERS OF
ORIGINAL COLA
A bold spin on an original cola
2016
1893 Pepsi Cola
Added
MAX TASTE
pepsi
ZERO SUGAR
2016
Pepsi Zero Sugar
Added

PLC


Renewal
Maturity
Growth
Decline
(a)
Introduction
Sales
Time
5 Grain


PRODUCT CYCLE REVISED WITH CAD/CAM
Computer-aided design
Computer-automated Drafting and documentation
Product concept
Design Engineering
Drafting
Customers and markets
Order new equipment and tooling
Process Planning
Computer-aided Process planning
Quality control
Production
scheduling
Computer-aided Quality control
Computer controlled Robots, machines, etc.
Computerized scheduling, Material requirements planning, Shop floor control


Typical Product Life Cycle
The Design Process
Design needs
Design definitions, specifications, and requirements
Collecting relevant design information and feasibility study
Synthesis
Analysis
The CAD Process
Design documentation and communication
Design evaluation
Design optimization
Design analysis
Design modeling and simulation
Design conceptualization
The Manufacturing Process
The CAM Process
Production planning
Design and procurement of new tools
Order materials
NC, CNC, DNC programming
Process planning
Production
Quality control
Packaging
Shipping
Marketing

Unit -1 INTRODUCTION

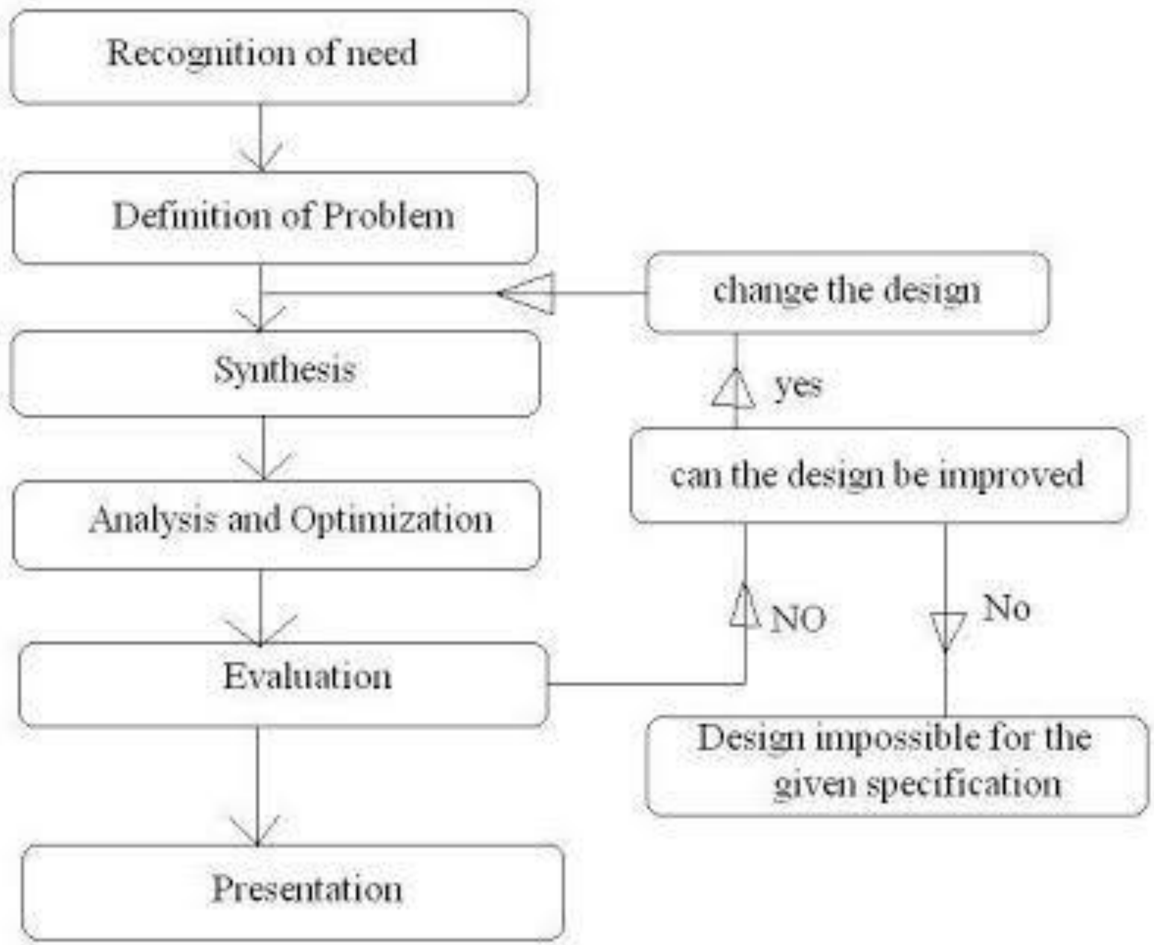
Product cycle- Design process- sequential and concurrent engineering- Computer aided design – CAD system architecture- Computer graphics – coordinate systems- 2D and 3D transformations homogeneous coordinates – Line drawing -Clipping- viewing transformation-Brief introduction to CAD and CAM – Manufacturing Planning, Manufacturing control- Introduction to CAD/CAM – CAD/CAM concepts ––Types of production – Manufacturing models and Metrics – Mathematical models of Production Performance

Shighely Model

1. Recognition of need
2. Definition of Problem
3. Synthesis
4. Analysis and Optimization
5. Evaluation
6. Presentation

recollect

- R
- DP
- S
- AO
- E
- P


Recognition of need
Definition of Problem
change the design
Synthesis
yes
can the design be improved
Analysis and Optimization
NO
No
Evaluation
Design impossible for the given specification
Presentation

Unit -1 INTRODUCTION

Product cycle- Design process- sequential and concurrent engineering- Computer aided design – CAD system architecture- Computer graphics – coordinate systems- 2D and 3D transformations homogeneous coordinates – Line drawing -Clipping- viewing transformation-Brief introduction to CAD and CAM – Manufacturing Planning, Manufacturing control- Introduction to CAD/CAM – CAD/CAM concepts ––Types of production – Manufacturing models and Metrics – Mathematical models of Production Performance

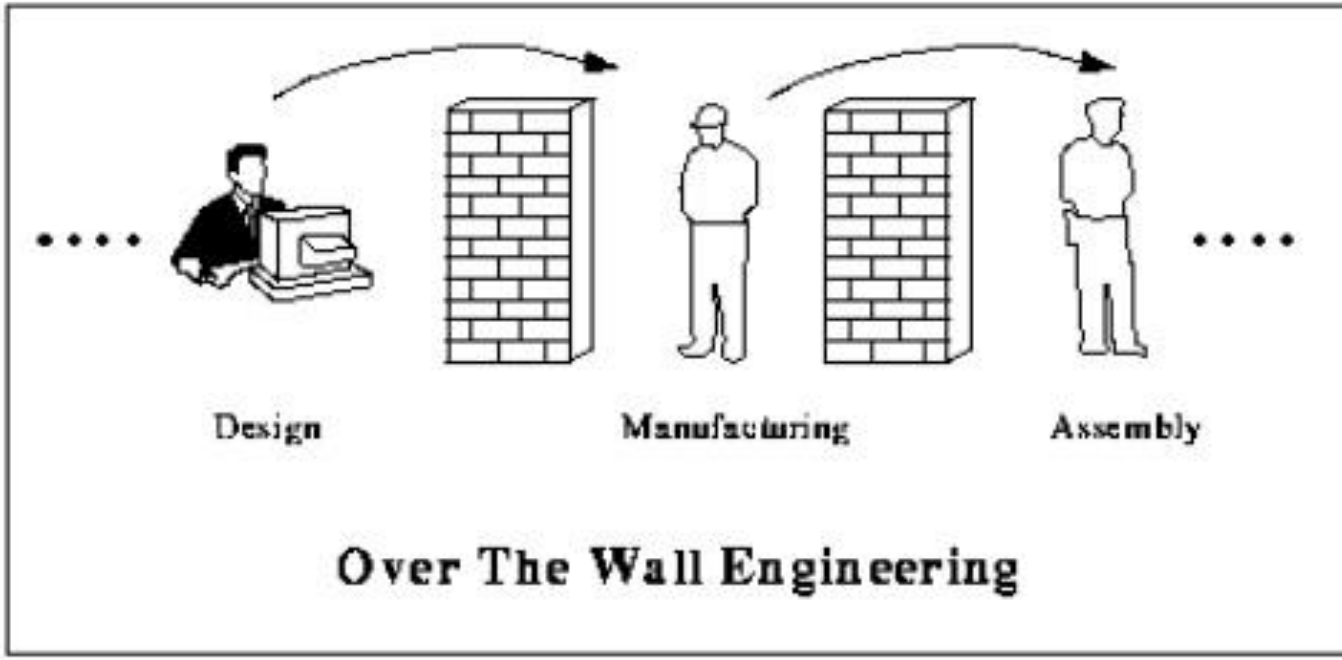
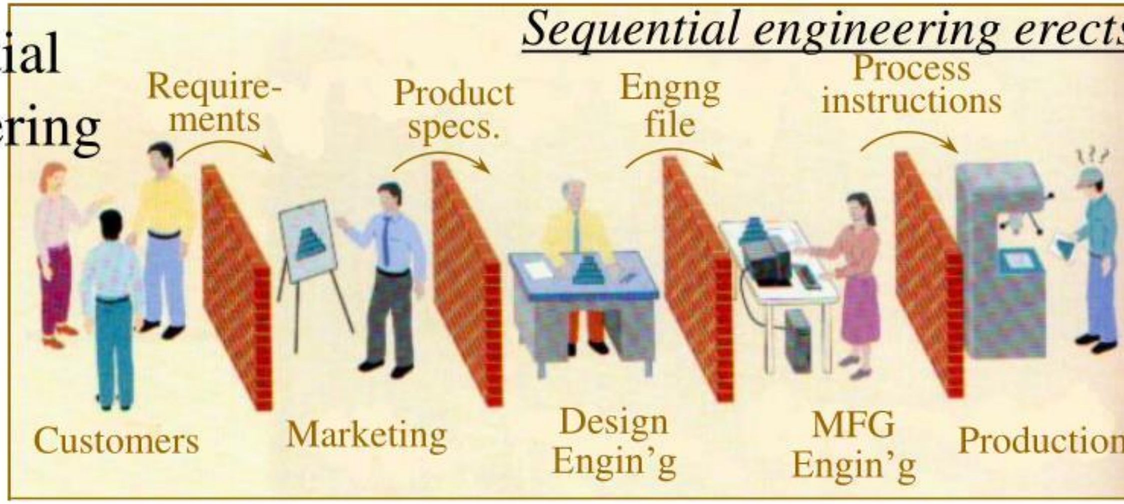


Figure.1.8.Over the Wall Engineering (Sequential Engineering)

Sequential vs. Concurrent Engineering (CE)

Sequential Engineering

Sequential engineering erects walls !!

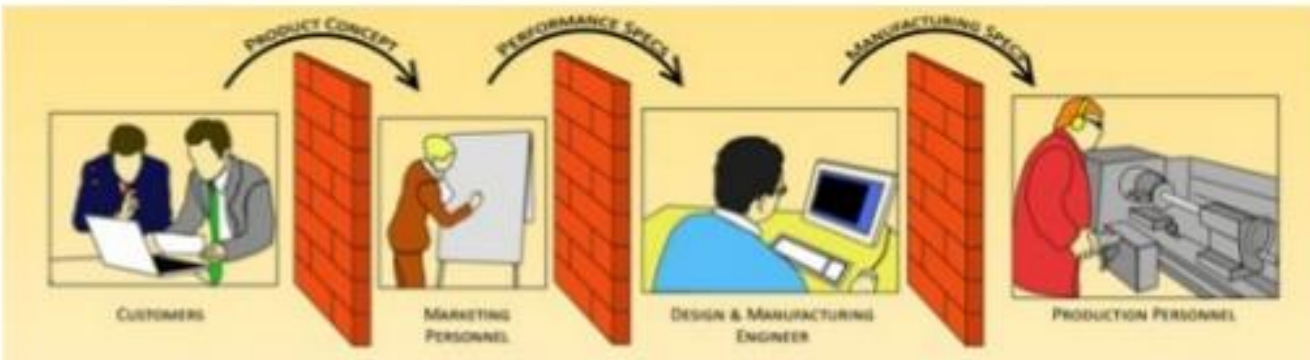


Concurrent Engineering

With CE the walls come tumbling down

ISAT 211 Mod 3-21

Concurrent Engineering



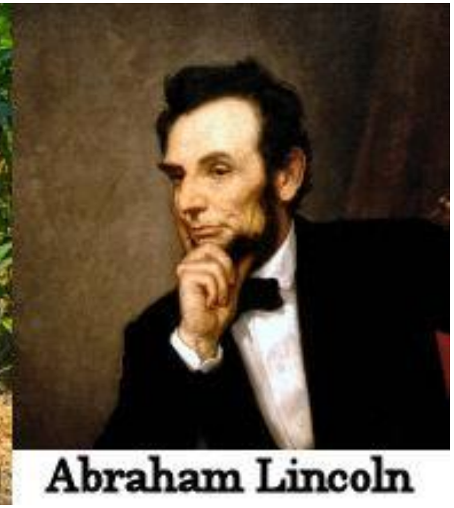
Traditional Process = Linear

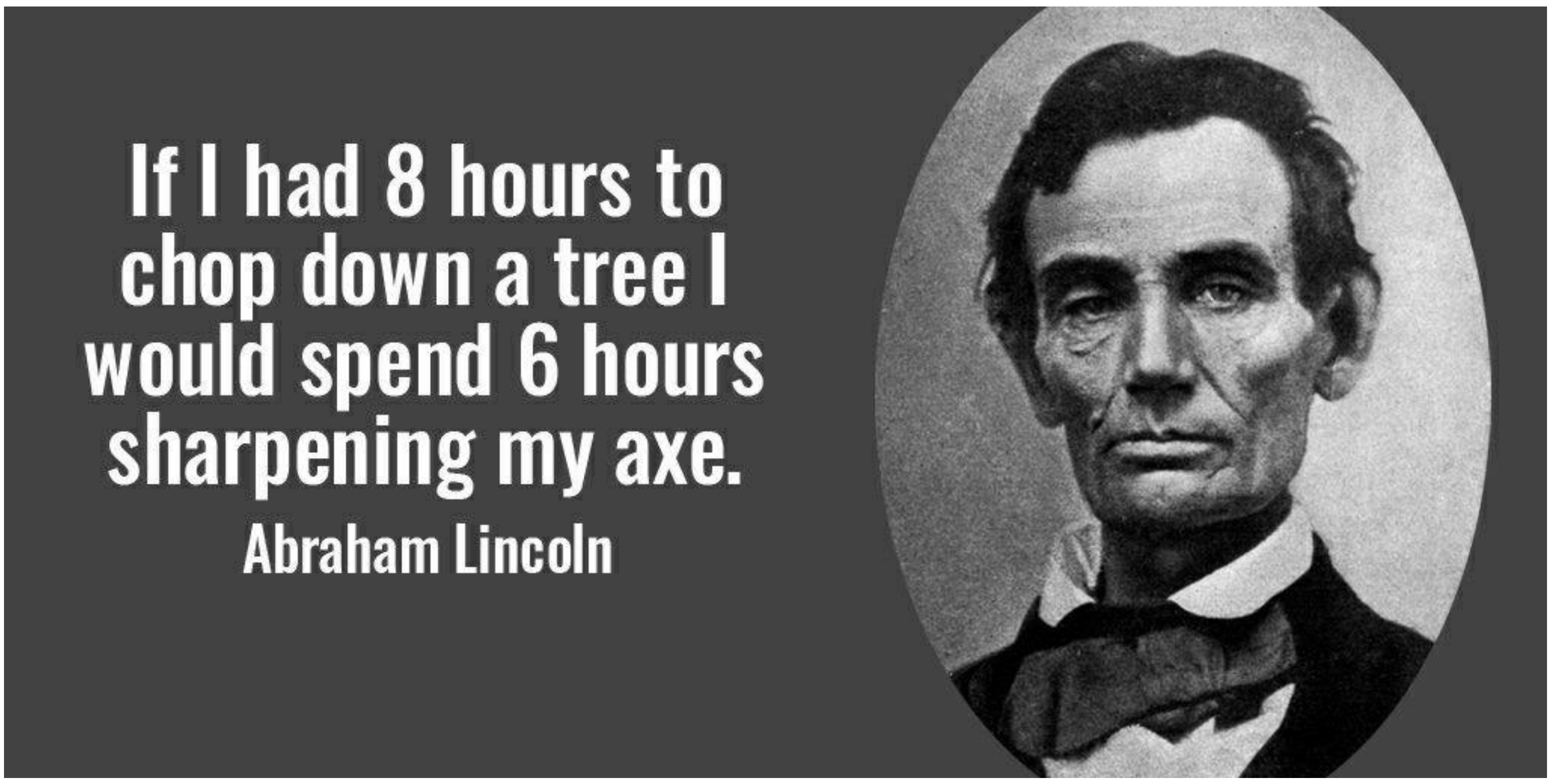
Vs

Concurrent Engineering = Team collaboration

Synergy – Interesting Aspect

- Two horses can pull about 9,000 pounds.
- How many pounds can four horses pull?
- The arithmetical response is 18,000. Sounds reasonable – but it's wrong!
- Four horses can actually pull over 30,000 pounds.
- It's synergy that makes the difference!


Abraham Lincoln


If I had 8 hours to chop down a tree I would spend 6 hours sharpening my axe.
Abraham Lincoln

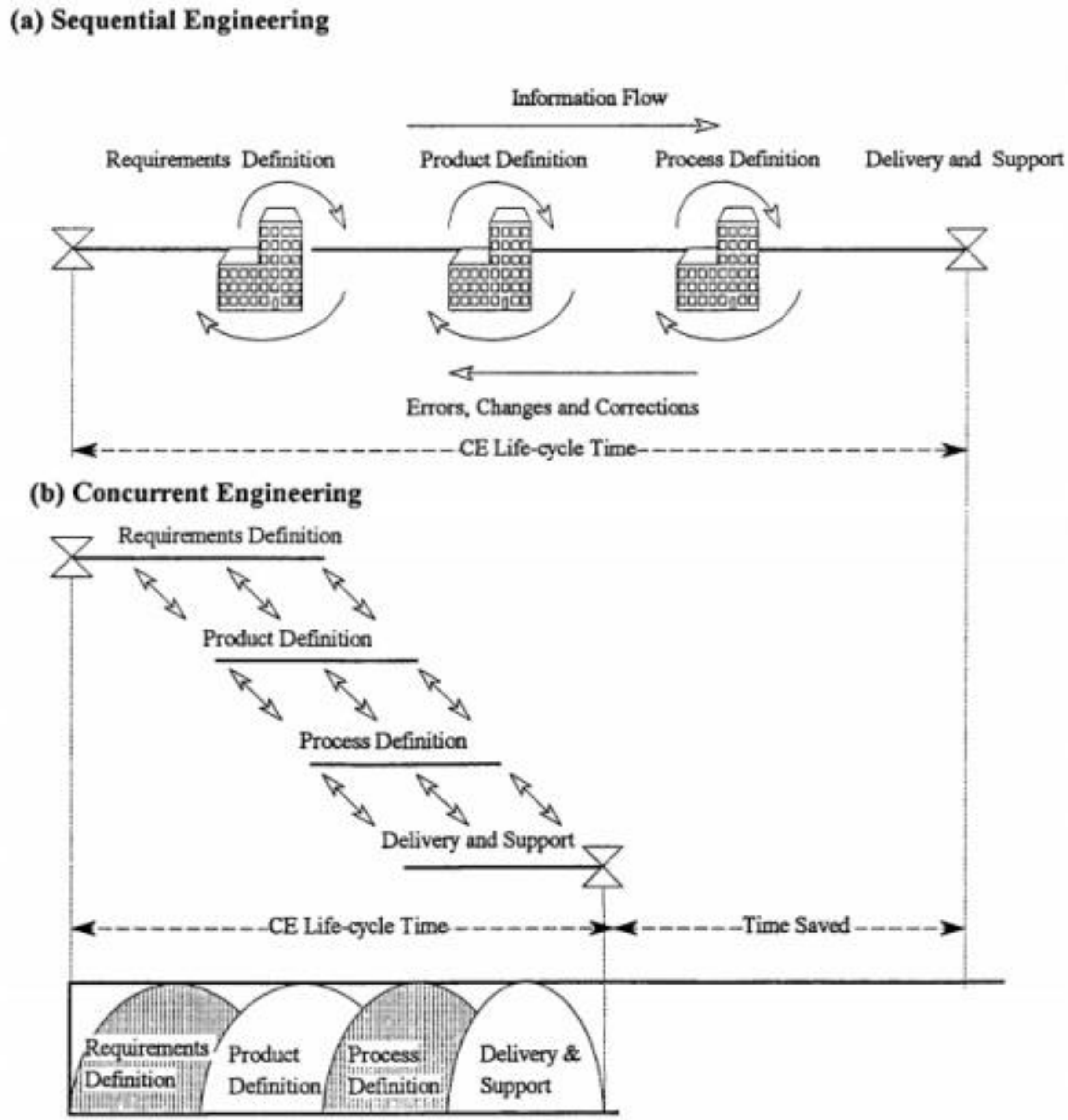
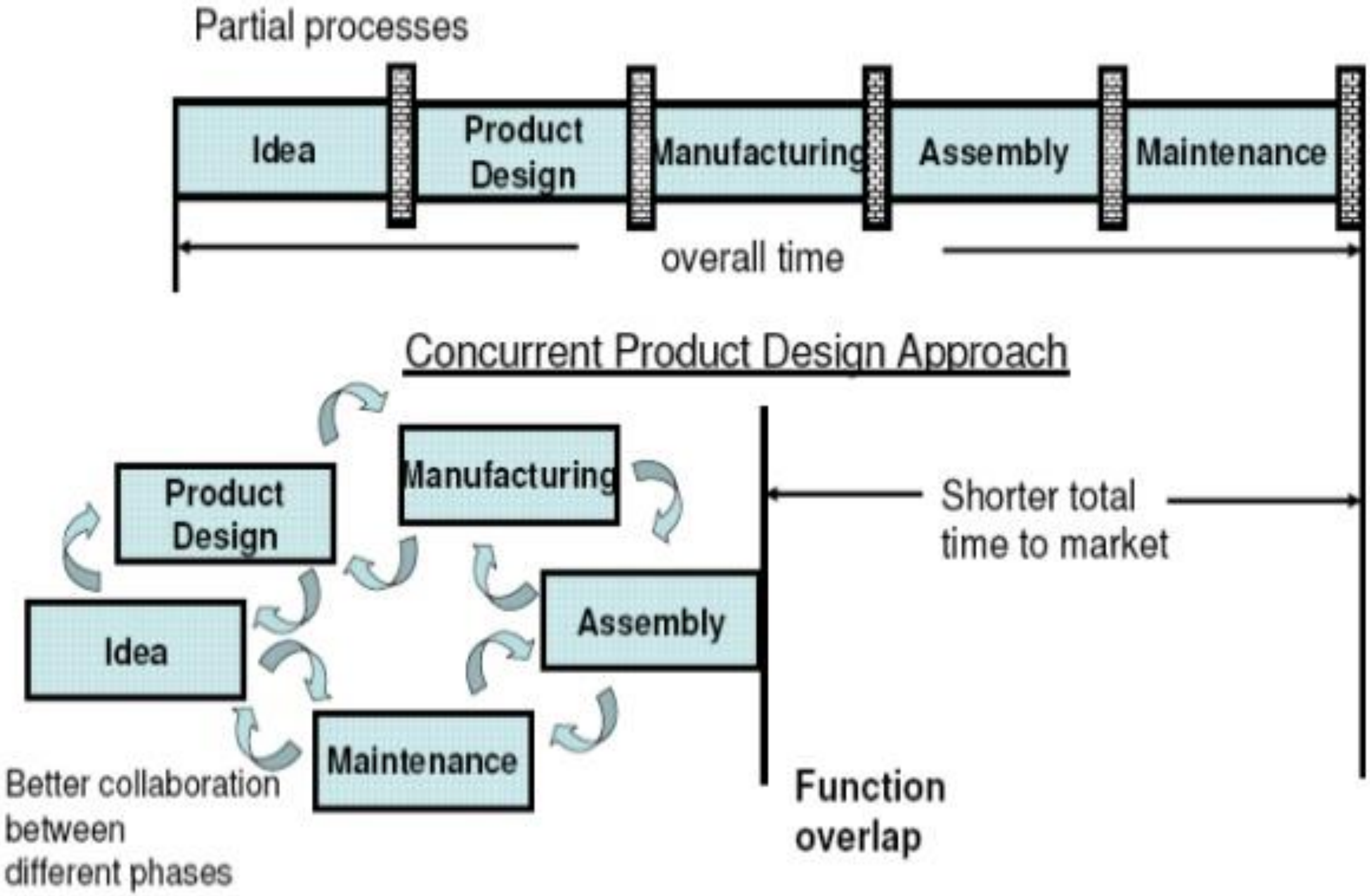


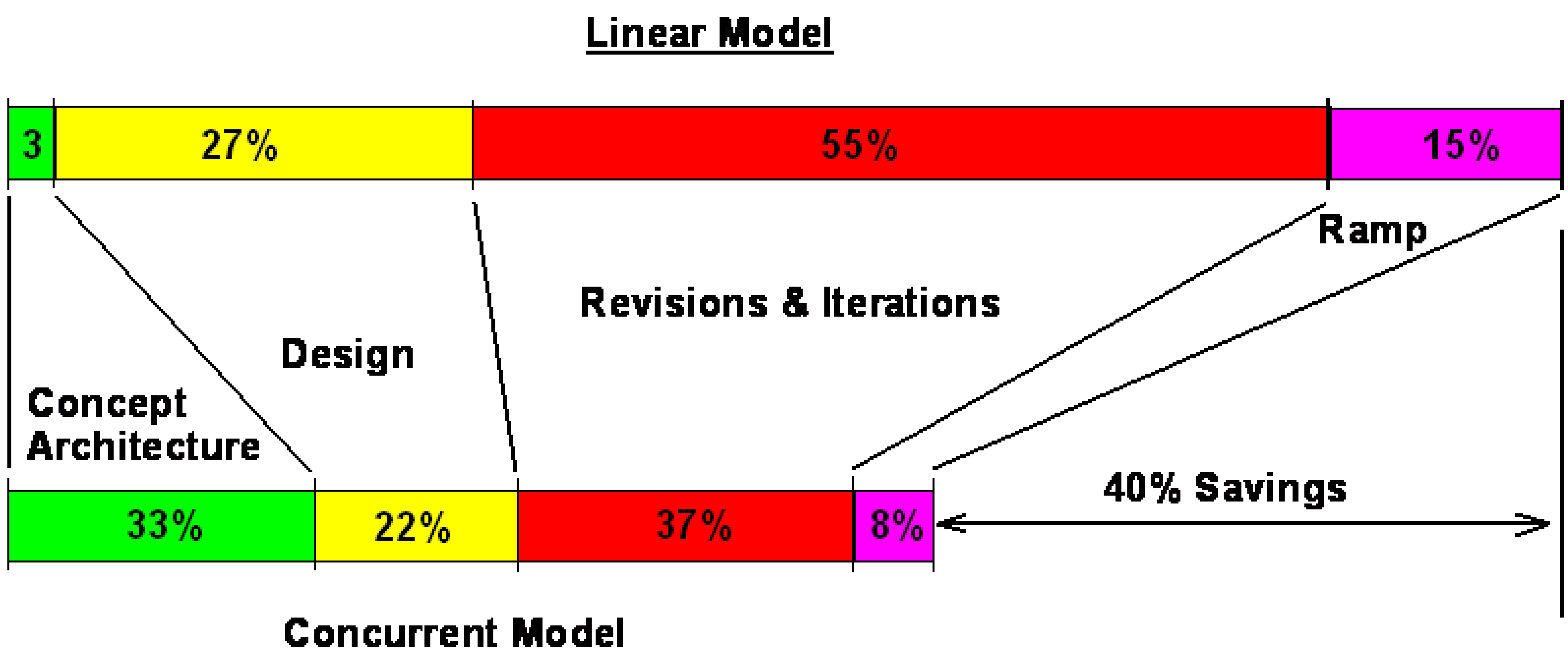
Figure.1.9. Sequential and Concurrent Engineering

Concurrent Engineering



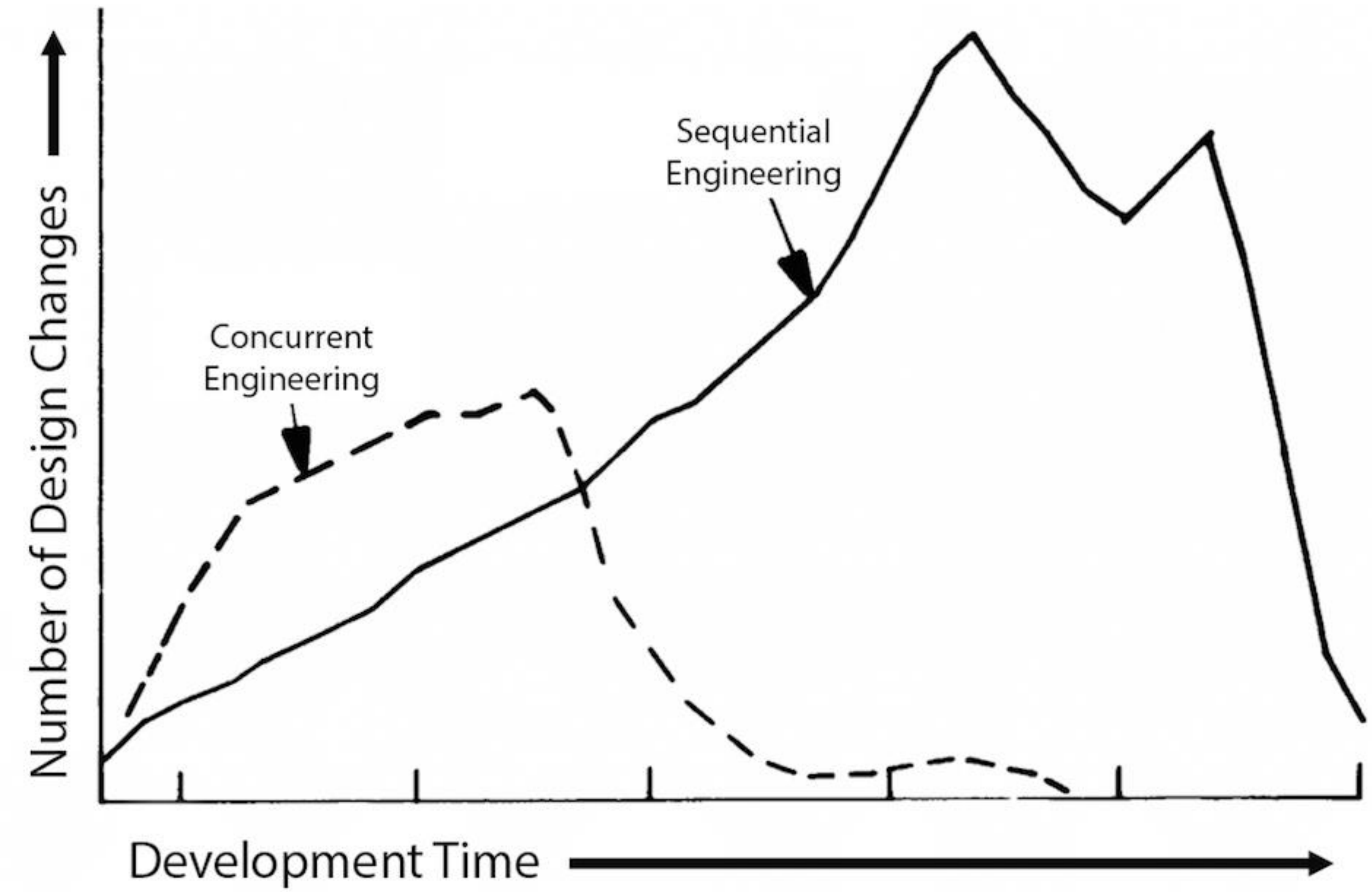
LEXMARK

Concurrent Engineering Time Savings



Mentor Graphic Case Study

Design Changes vs Development Time




T
E
A
M

Table	Chair	Room	Scale	Pen	Train
Mobile	Card	Shirt	Projector	Stand	Bus
Book	Bed	Ring	Bat	Ball	Cycle
Apple	File	Paper	Subject	Bottle	Tap
Cap	Plastic	Tea	Charger	Watch	Water
Diary	Cloth	Shoe	Tie	Car	Juice


T
E
A
M

Table	Chair	Room	Scale	Pen	Train
Mobile	Card	Shirt	Projector	Stand	Bus
Book	Bed	Ring	Bat	Ball	Cycle
Apple	File	Paper	Subject	Bottle	Tap
Cap	Plastic	Tea	Charger	Watch	Water
Diary	Cloth	Shoe	Tie	Car	Juice

Unit -1 INTRODUCTION

Product cycle- Design process- sequential and concurrent engineering- Computer aided design – CAD system architecture- Computer graphics – coordinate systems- 2D and 3D transformations homogeneous coordinates – Line drawing -Clipping- viewing transformation-Brief introduction to CAD and CAM – Manufacturing Planning, Manufacturing control- Introduction to CAD/CAM – CAD/CAM concepts —Types of production – Manufacturing models and Metrics – Mathematical models of Production Performance

- Major classes :
- Main frame
- Mini computer
- Workstation
- Microcomputer
- Based

- Application areas :
- Mechanical
- Architectural
- Construction
- Circuit design
- Chip design

- Cost :
- High end
- Low end

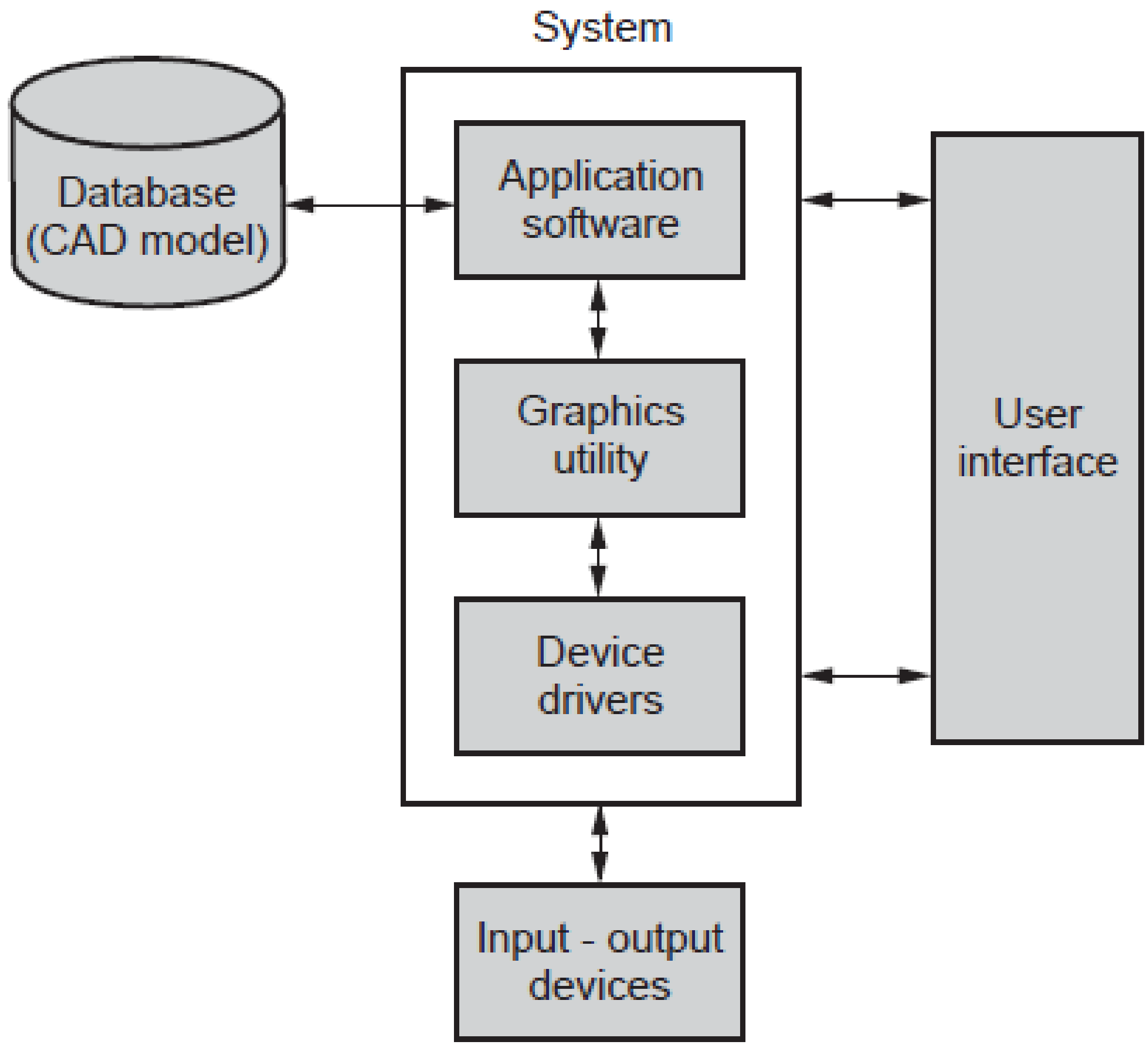


Fig. 1.6.1 CAD system architecture

Unit -1 INTRODUCTION

Product cycle- Design process- sequential and concurrent engineering- Computer aided design – CAD system architecture- Computer graphics – coordinate systems- 2D and 3D transformations homogeneous coordinates – Line drawing -Clipping- viewing transformation-Brief introduction to CAD and CAM – Manufacturing Planning, Manufacturing control- Introduction to CAD/CAM – CAD/CAM concepts ––Types of production – Manufacturing models and Metrics – Mathematical models of Production Performance

Computer Graphics

Computer Graphics involves Creation, displays, manipulation and storage for proper visualization using computer

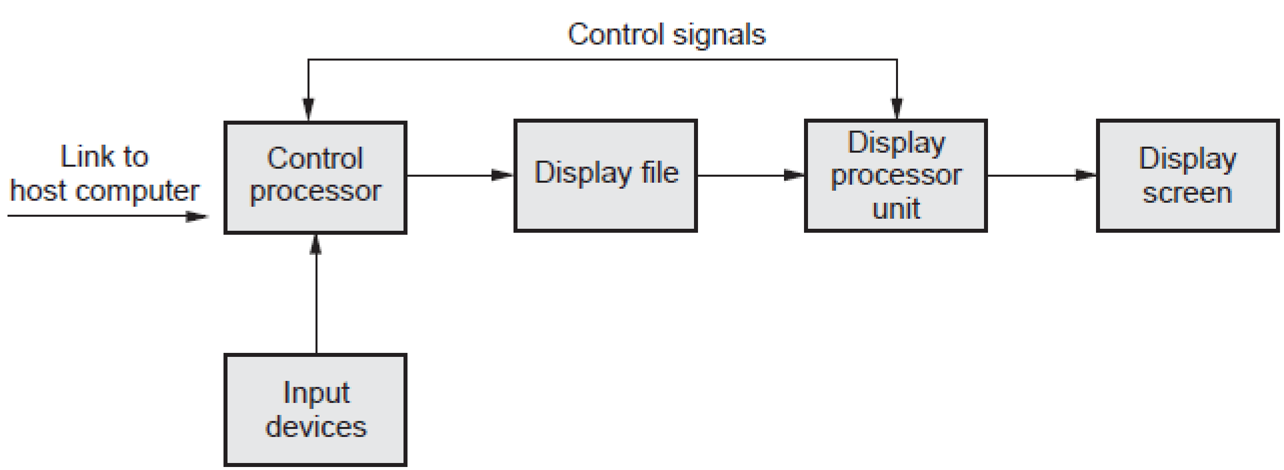
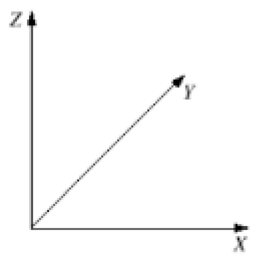


Fig. 1.7.1 A basic computer graphics layout

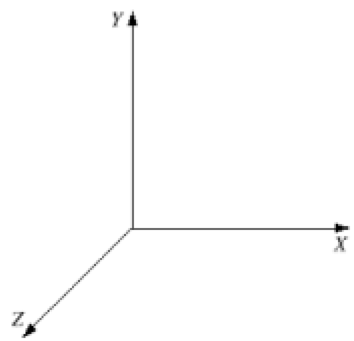
Unit -1 INTRODUCTION

Product cycle- Design process- sequential and concurrent engineering- Computer aided design – CAD system architecture- Computer graphics – coordinate systems- 2D and 3D transformations homogeneous coordinates – Line drawing -Clipping- viewing transformation-Brief introduction to CAD and CAM – Manufacturing Planning, Manufacturing control- Introduction to CAD/CAM – CAD/CAM concepts —Types of production – Manufacturing models and Metrics – Mathematical models of Production Performance

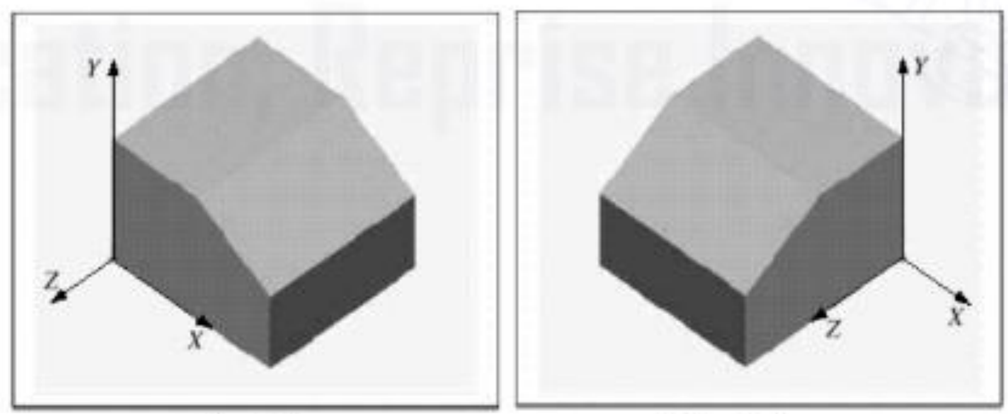
Coordinate systems



(*a*) *XY* plane defines model top view



(*b*) *XY* plane defines model front view



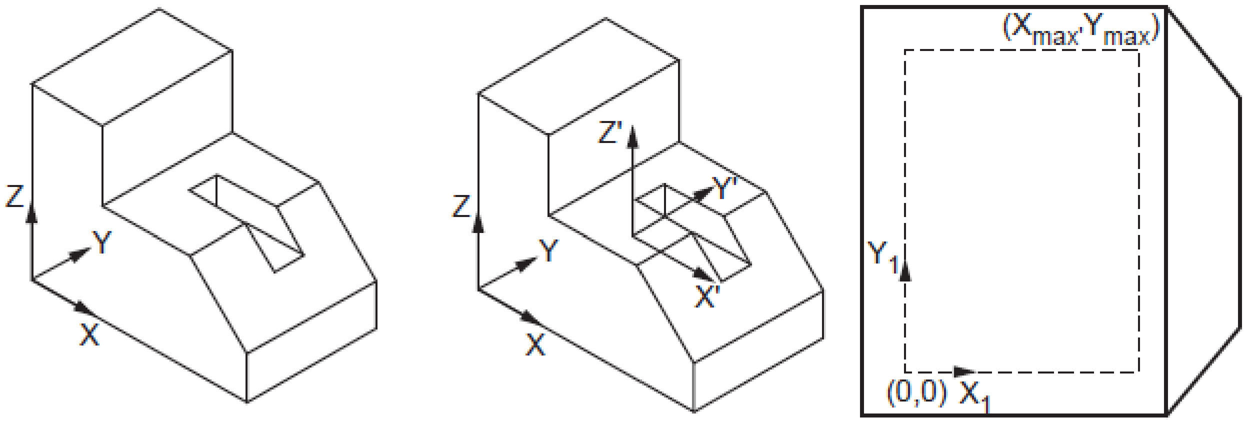
Orientation

Orientation

Coordinate systems

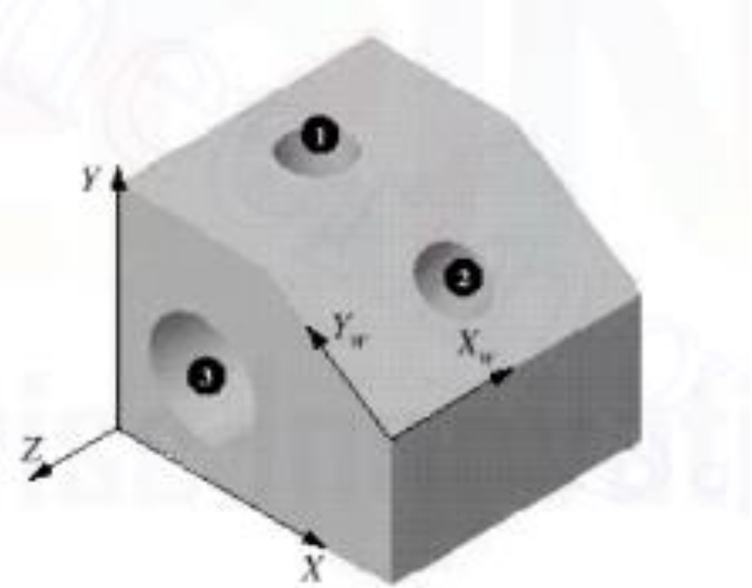
- Model Co-ordinate System / World Co-ordinate System
- Working Co-ordinate System / User Co-ordinate System
- Screen Co-ordinate System / Device Co-ordinate System

Coordinate systems



(a) MCS (b) WCS (c) SCS

Example



Unit -1 INTRODUCTION

Product cycle- Design process- sequential and concurrent engineering- Computer aided design – CAD system architecture- Computer graphics – coordinate systems- 2D and 3D transformations homogeneous coordinates – Line drawing -Clipping- viewing transformation-Brief introduction to CAD and CAM – Manufacturing Planning, Manufacturing control- Introduction to CAD/CAM – CAD/CAM concepts ––Types of production – Manufacturing models and Metrics – Mathematical models of Production Performance

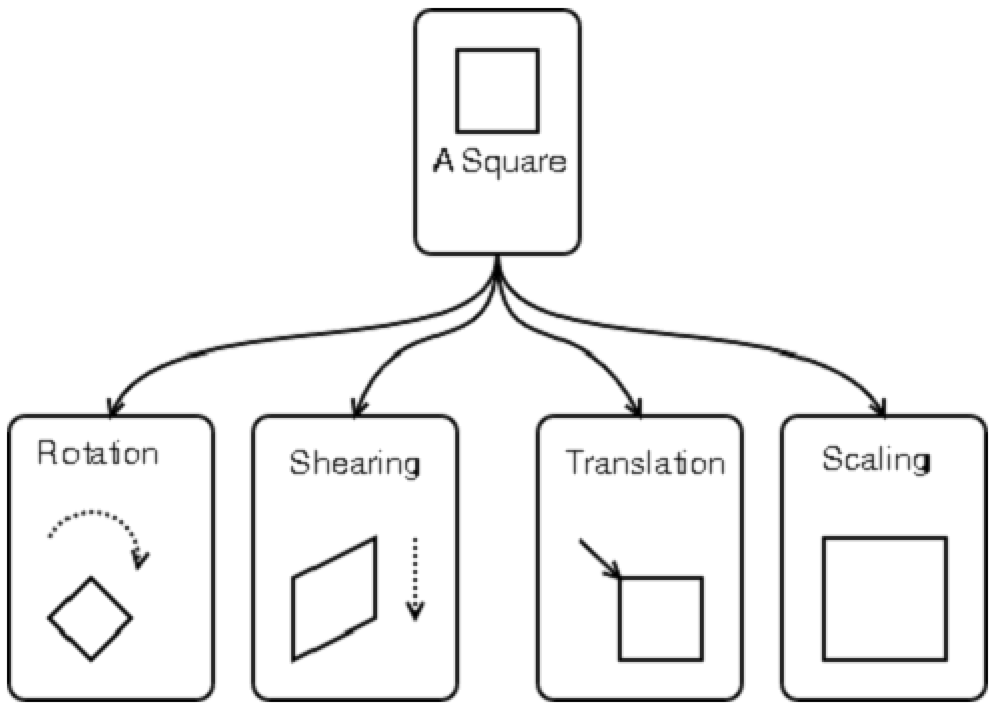
Geometric Transformation

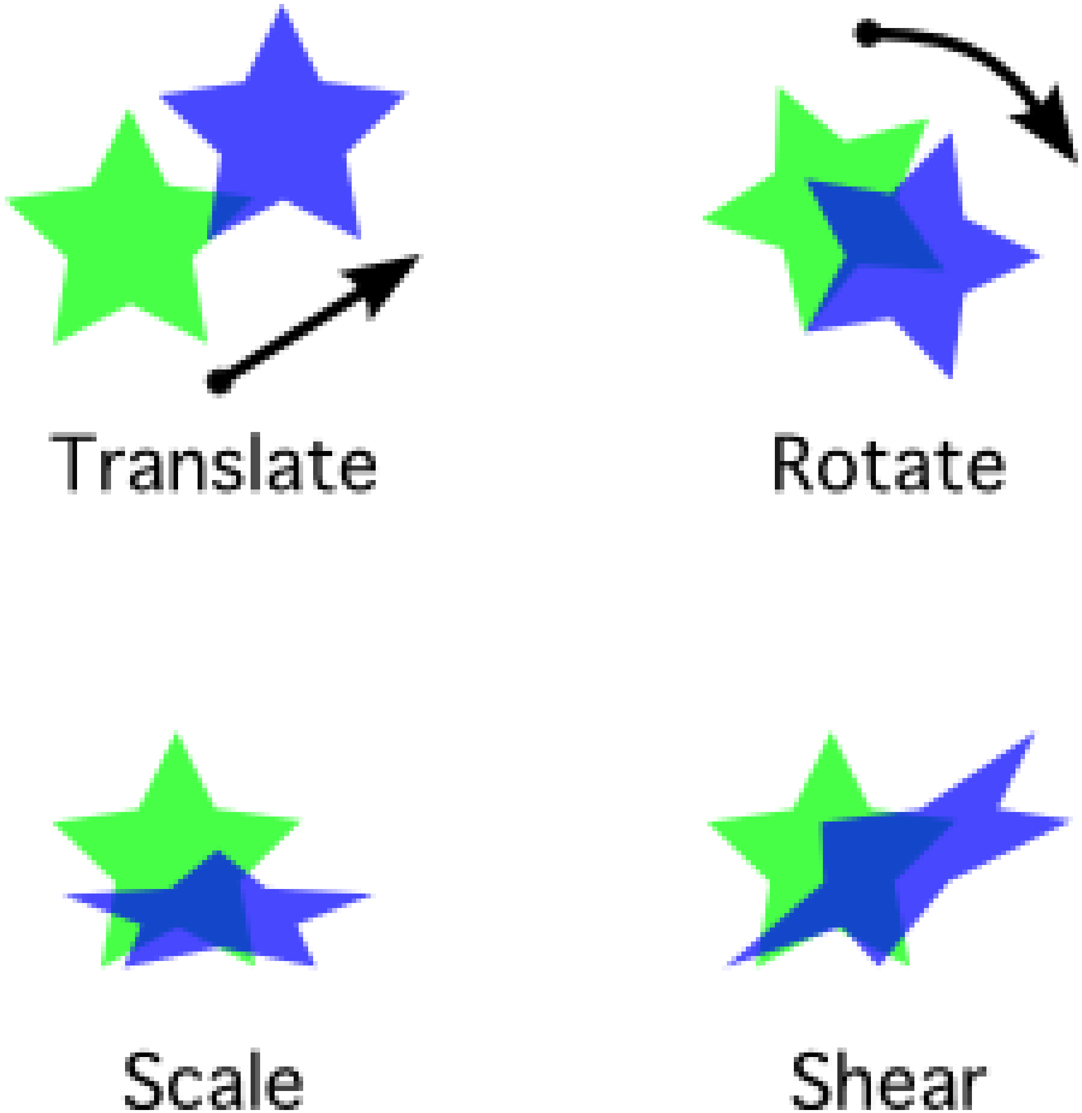
- It plays a major role in geometric modeling and viewing
- Used to express the locations of entities relative to others and to move them around in the modeling space.
- Used to generate different views of a model for visualization and drafting purposes.
- Used to create Animated files of geometric models.

Basic Geometric Transformations

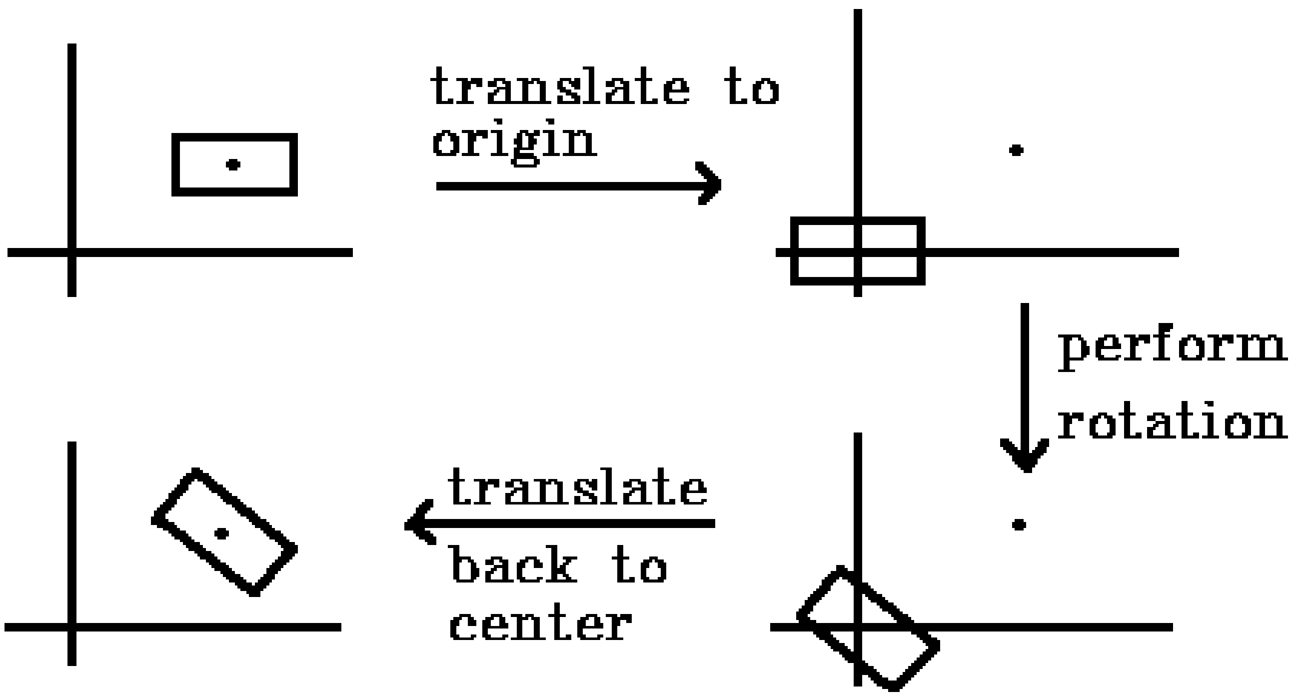
- Translation
- Rotation
- Scaling
- Mirroring
- Shearing

2D TRANSFORMATION

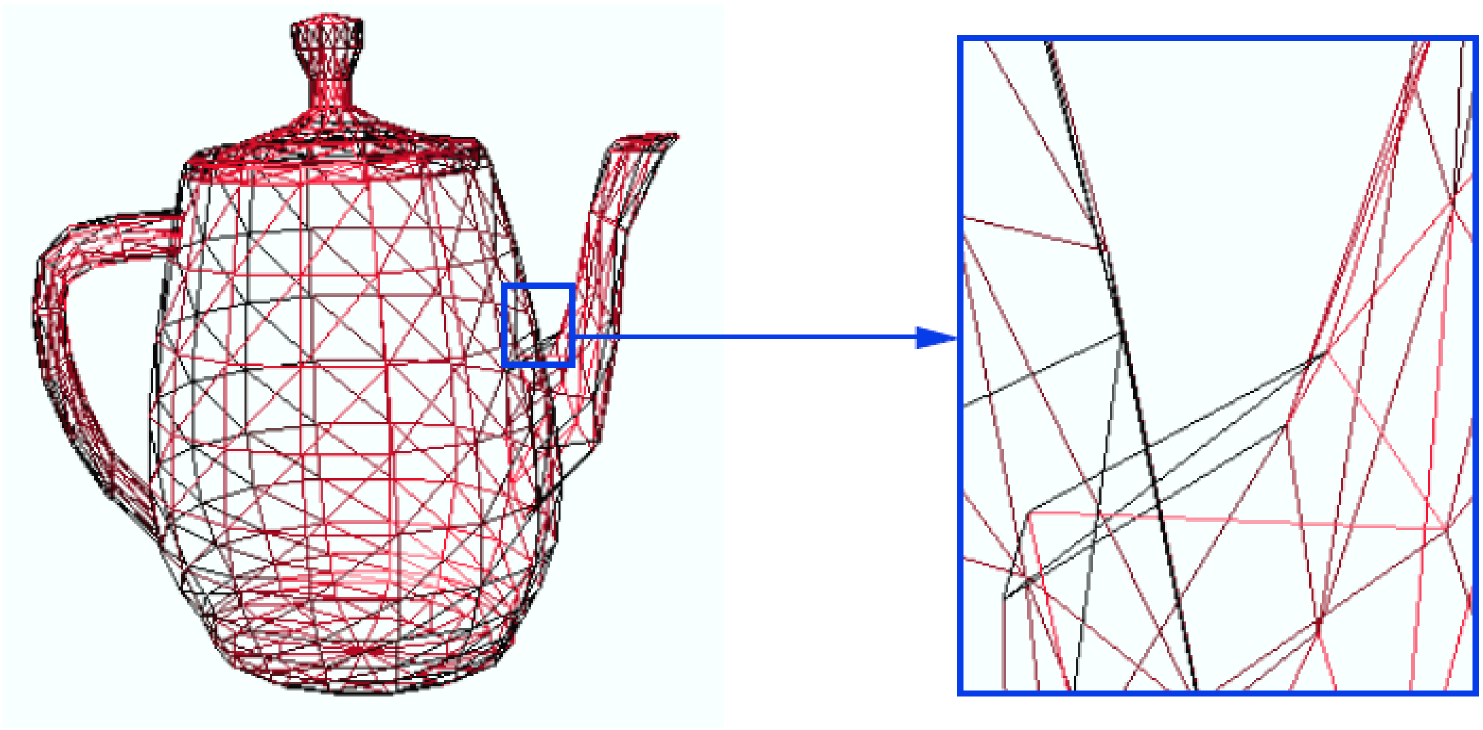



Translate
Rotate
Scale
Shear

Rotation about arbitrary point



Line Drawing Algorithms

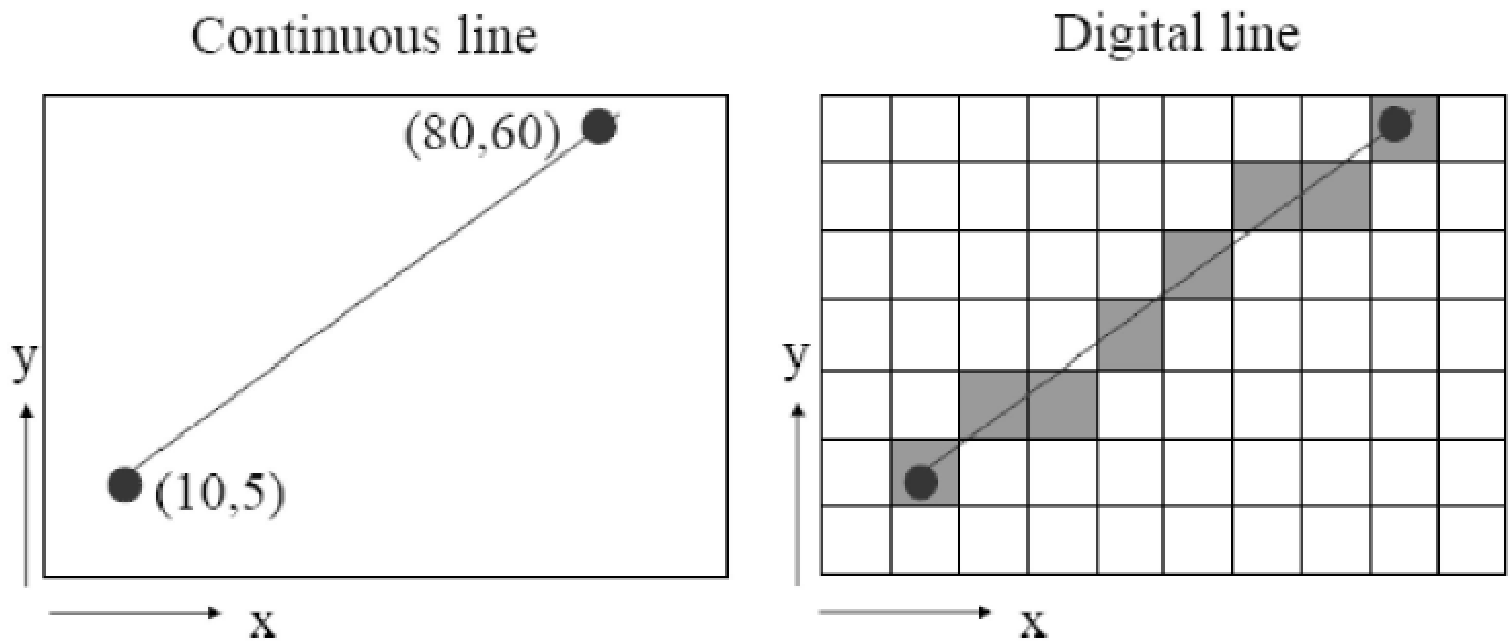


The lines of this object appear **continuous**

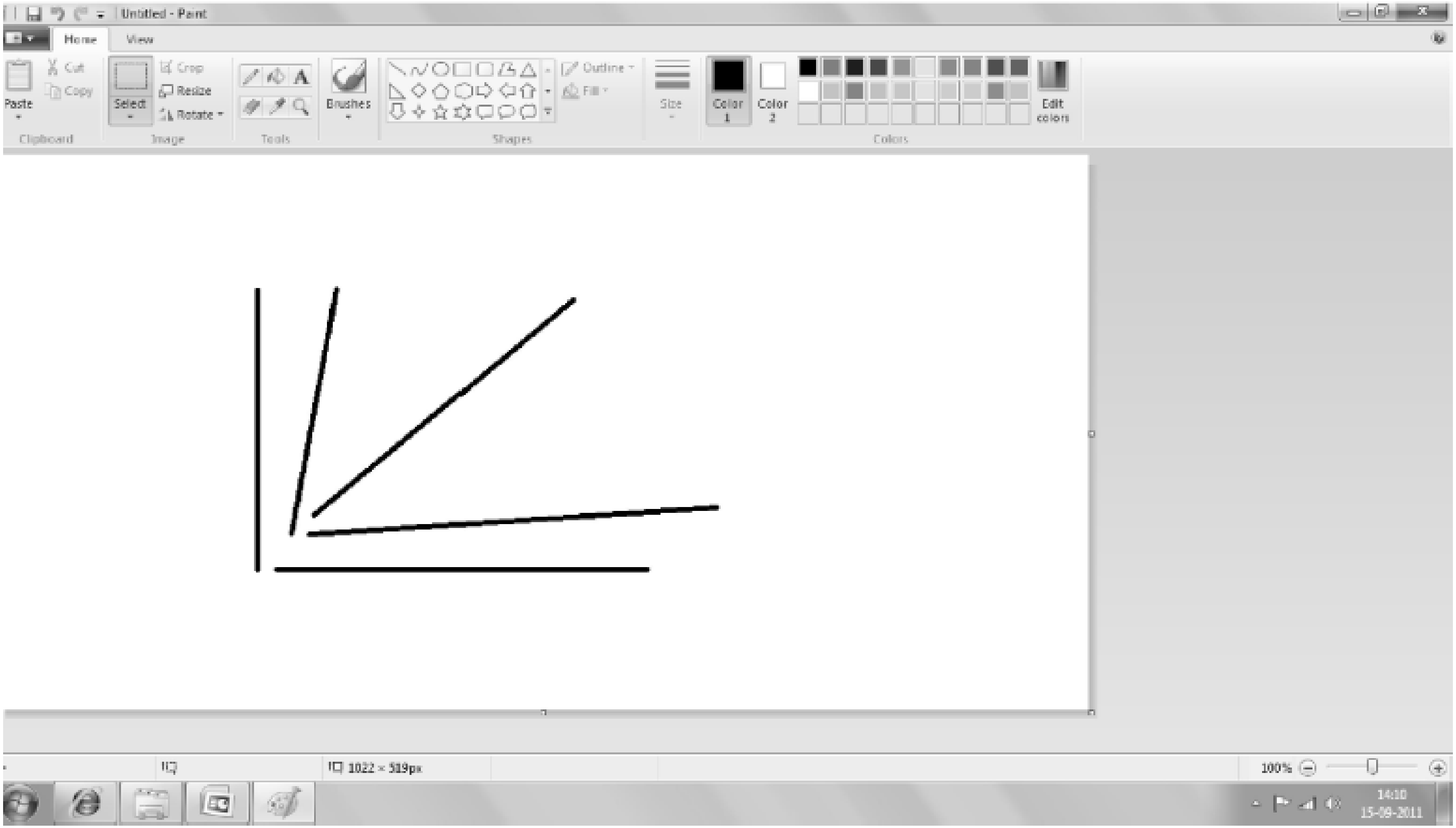
However, they are **made of pixels**

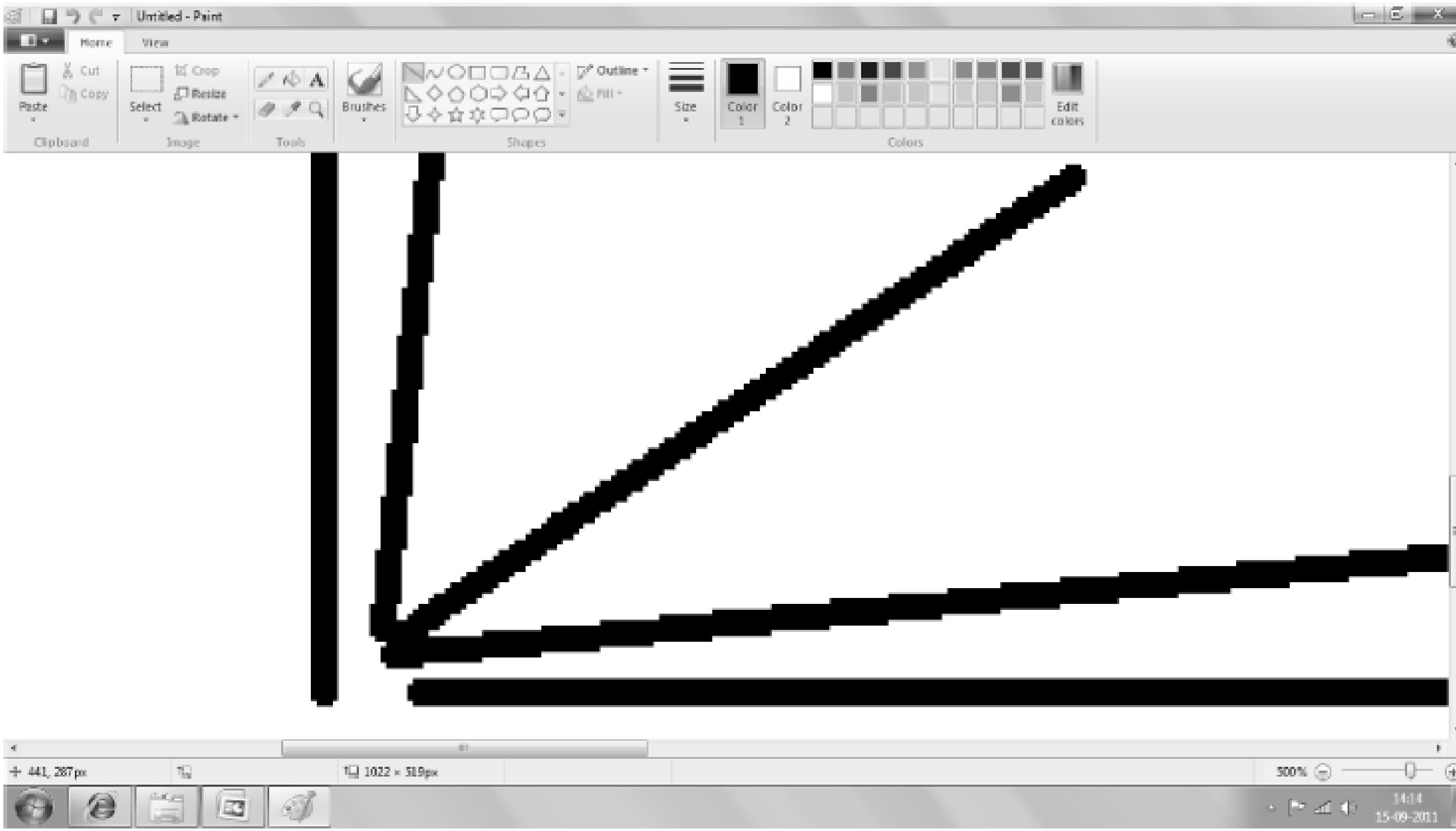
2D Drawing

In reality, photographs are arrays of pixels, not abstract mathematical continuous planes



In graphics, the conversion from continuous to discrete 2D primitives is called scan conversion or rasterization

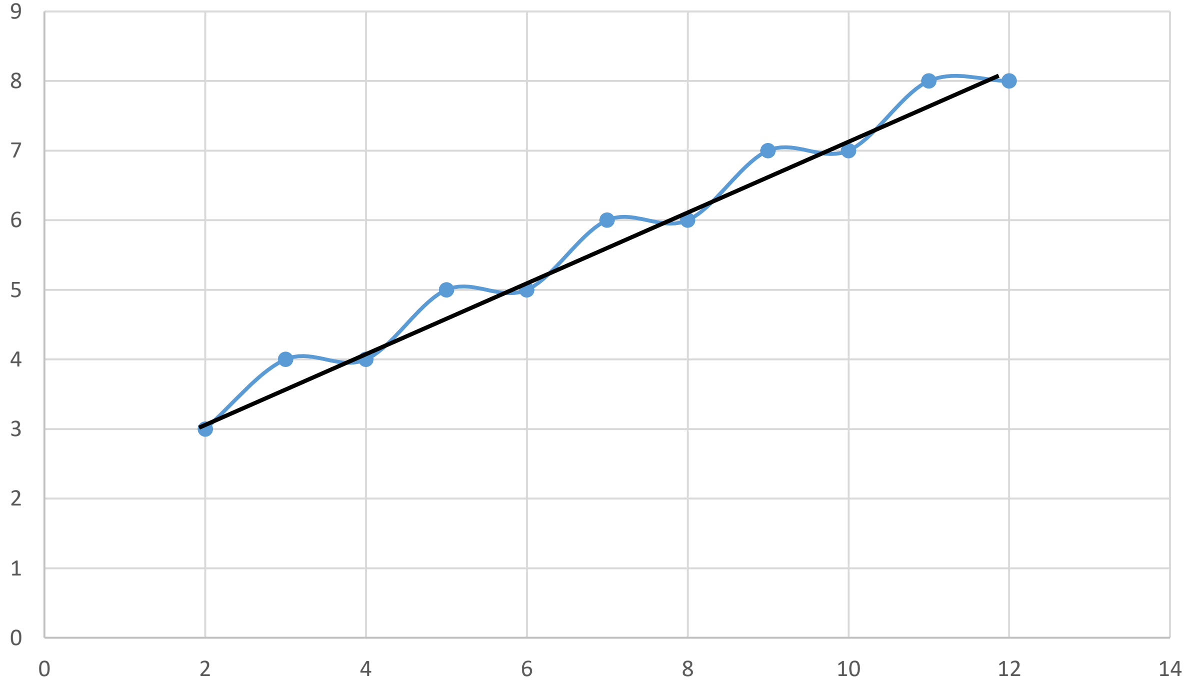

Untitled - Paint
Home
View
Paste
Cut
Copy
Select
Crop
Resize
Rotate
Brushes
Outline
Fill
Size
Color 1
Color 2
Edit colors
Clipboard
Image
Tools
Shapes
Colors
1022 × 519px
100%
14:10
15-09-2011



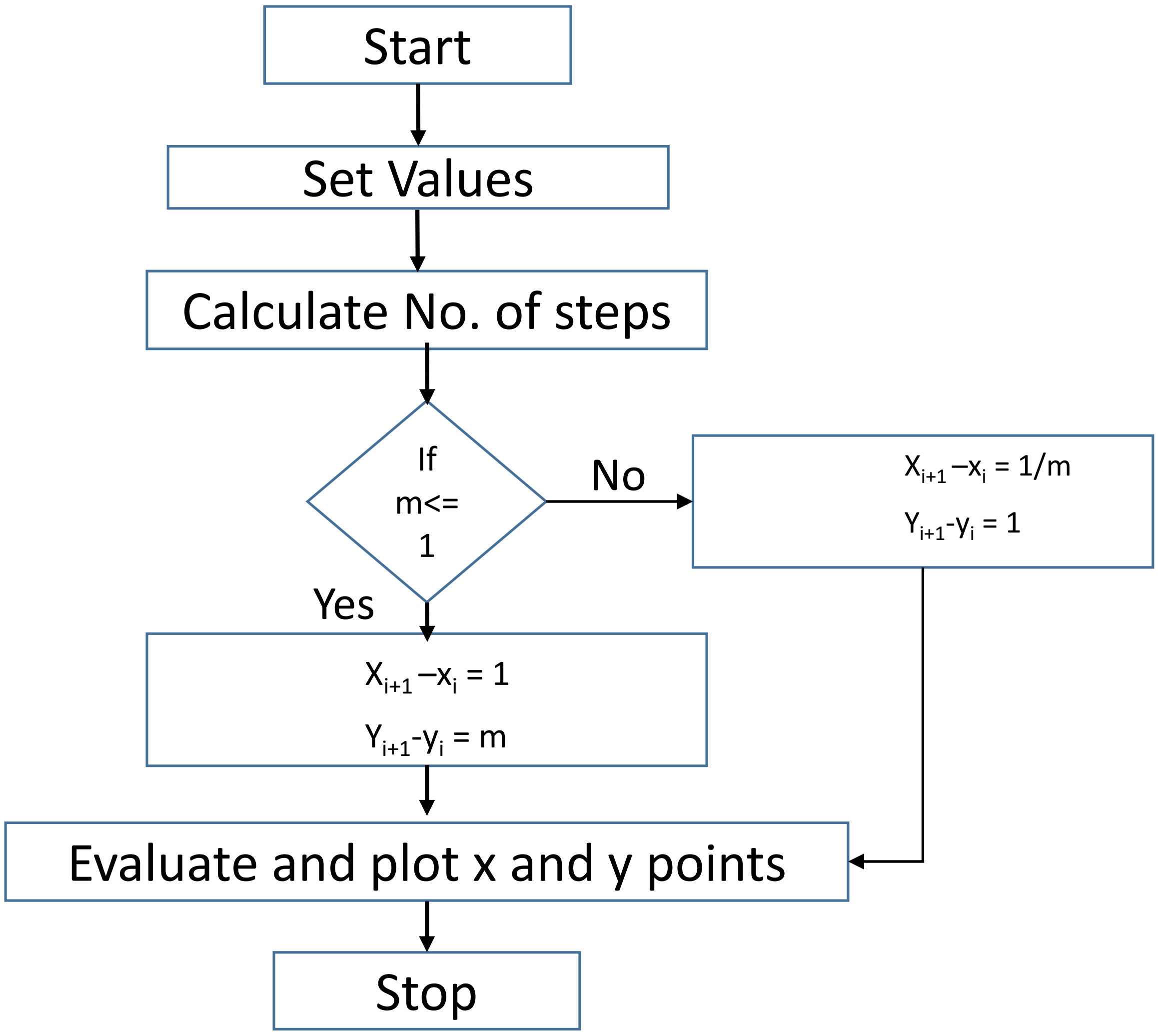
Untitled - Paint
Home
View
Paste
Cut
Copy
Select
Crop
Resize
Rotate
Clipboard
Image
Tools
Brushes
Shapes
Outline
Fill
Size
Color 1
Color 2
Edit colors
Colors
441, 287px
1022 × 519px
500%
14:14
15-09-2011

Rasterize A(2,3) B(12,8)

Xi	Yi	Xp	Yp
2	3	2	3
3	3.5	3	4
4	4	4	4
5	4.5	5	5
6	5	6	5
7	5.5	7	6
8	6	8	6
9	6.5	9	7
10	7	10	7
11	7.5	11	8
12	8	12	8



9
8
7
6
5
4
3
2
1
0
0
2
4
6
8
10
12
14



Start
Set Values
Calculate No. of steps
If
m<=
1
No
Xi+1 –xi = 1/m
Yi+1-yi = 1
Yes
Xi+1 –xi = 1
Yi+1-yi = m
Evaluate and plot x and y points
Stop

DDA line algorithm : Disadvantages

- Key disadvantage:

  it relies on floating point operations to compute pixel positions.

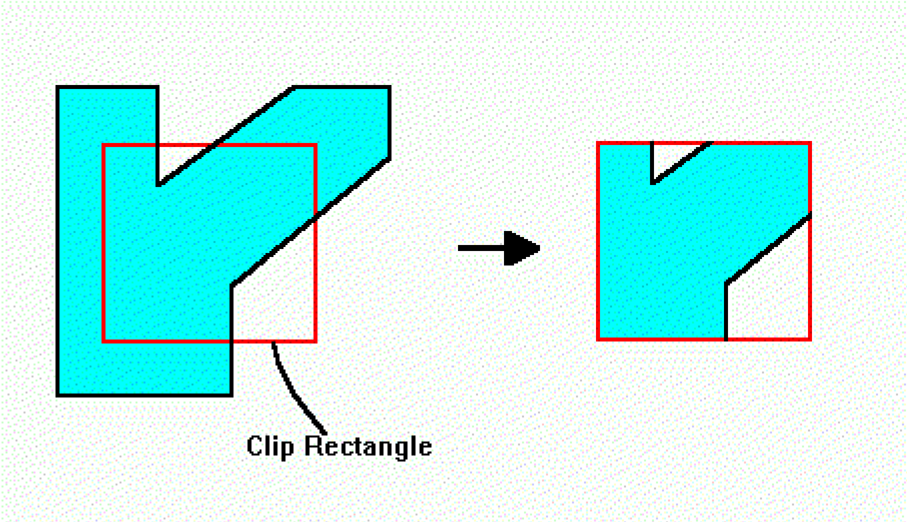
- Implications:

❑ Computationally inefficient because floating-point operations are slow.

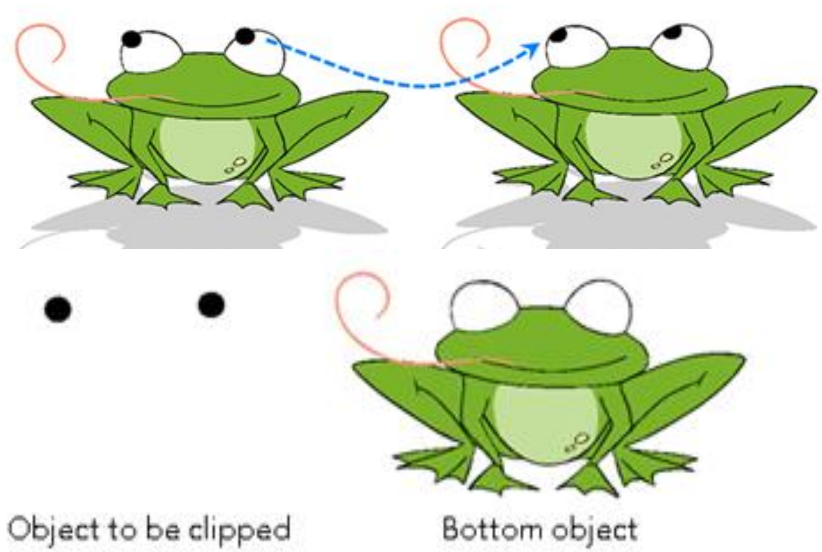
❑ Round-off errors accumulate, producing incorrect line drawings (e.g., if m is rounded to 0.9 even though it is equal to 0.99, lines of length > 10 will be drawn inaccurately)

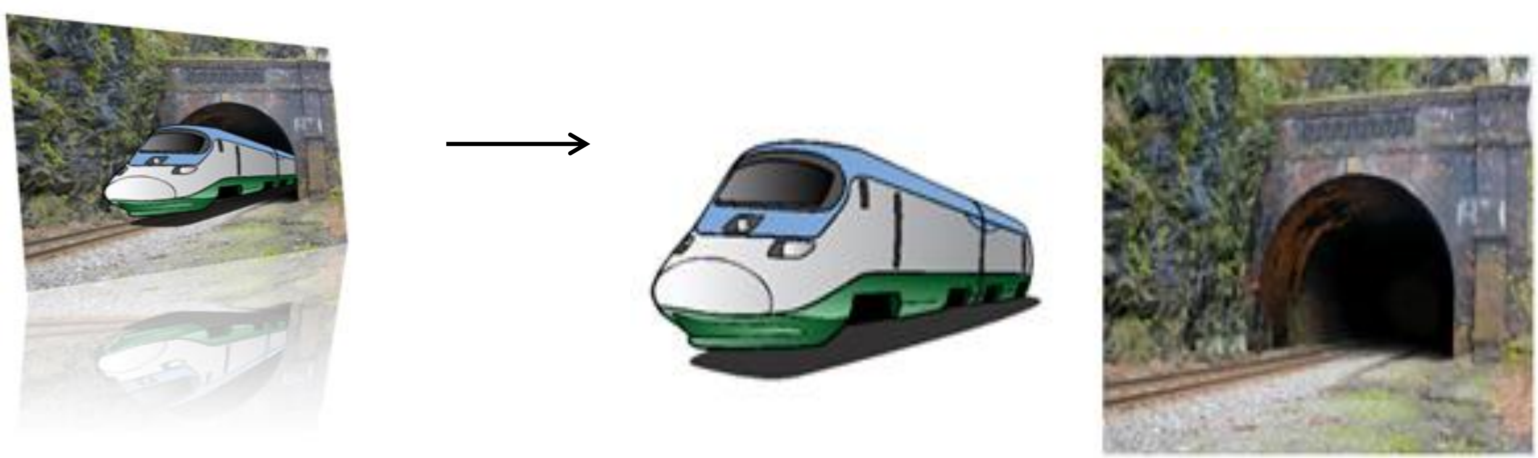
Clipping

- Any Procedure that identifies those portions of a picture that are either inside or outside of a specified region of a space is referred to as a Clipping algorithm or simply Clipping.
- The region against which an object is to be clipped is known as Clipping Window.



CLIPPING



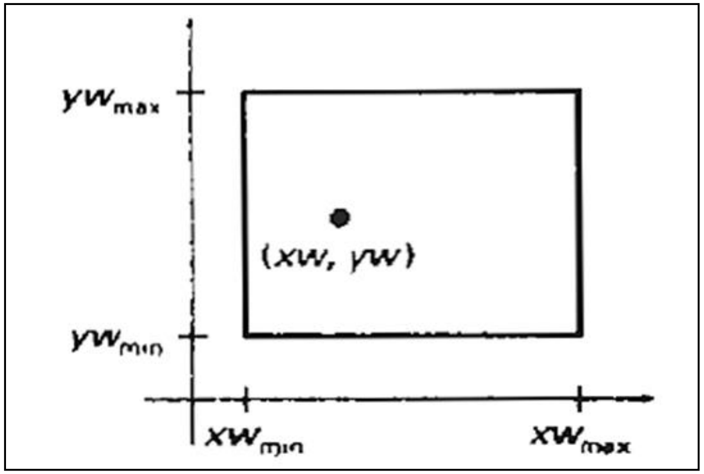


Types of Clipping

- Point Clipping
- Line Clipping
- Polygon Clipping
- Curve Clipping
- Text Clipping

Point Clipping

- We have a point P=(xw, yw) for display if the following inequalities are satisfied

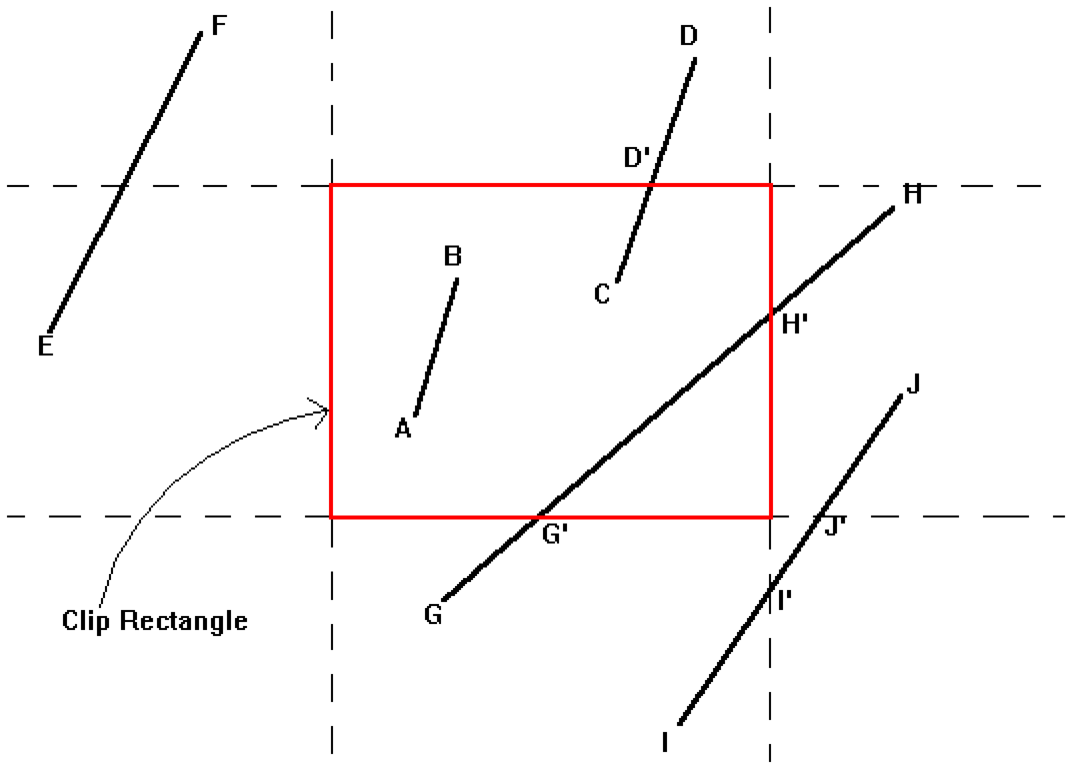


$$xw_{min} \le xw \le xw_{max}$$
$$yw_{min} \le yw \le yw_{max}$$

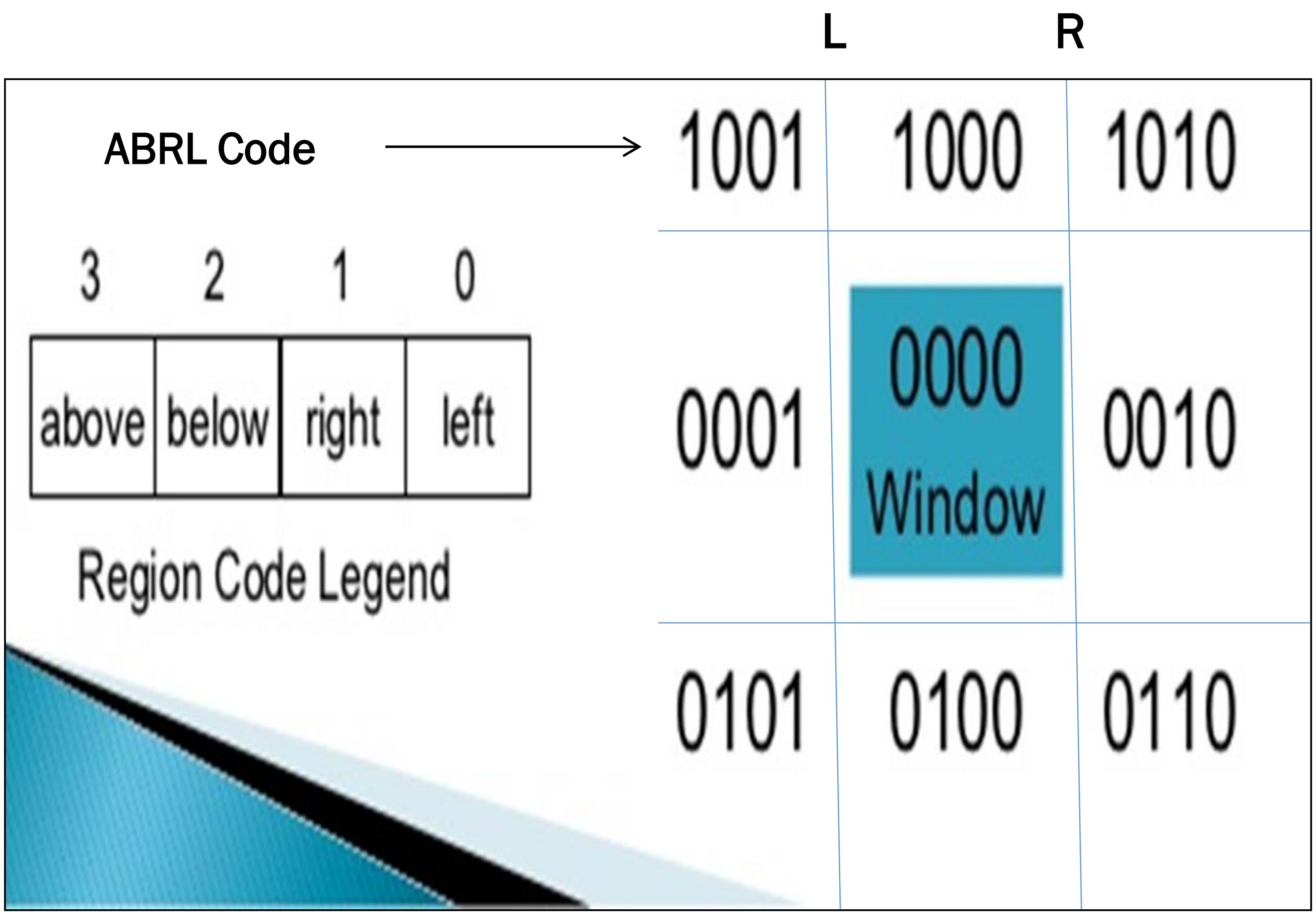
where the xw_{min}, y_{wmin}, xw_{max}, yw_{max} are the edge of the Clip Window.

- If any one of these four inequalities is not satisfied, the point is clipped

Line Clipping

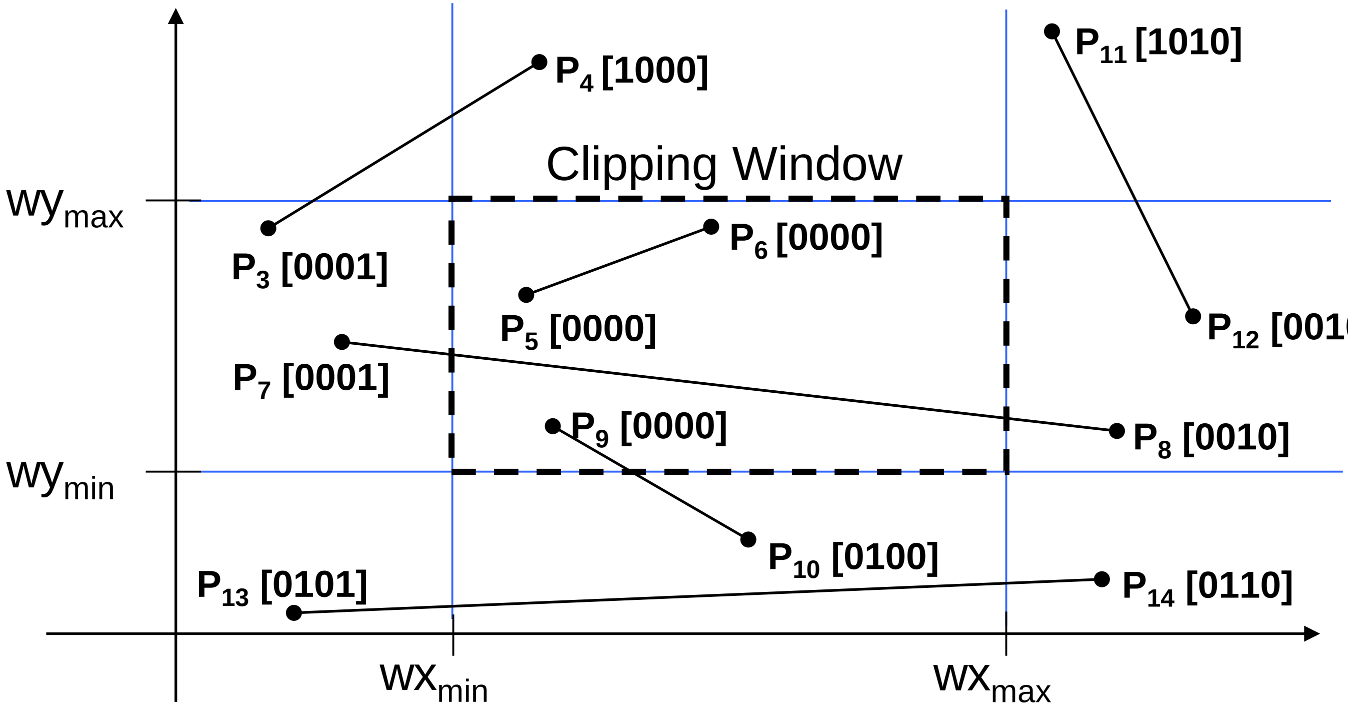


Cohen - Sutherland Algorithm



ABRL coding

1. **TRIVIALLY ACCEPT**
2. **TRIVIALLY REJECT**

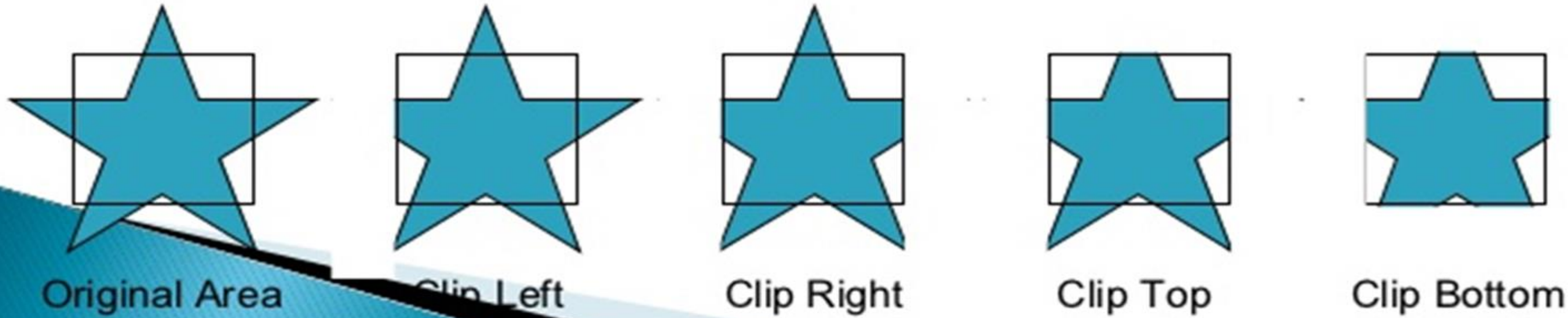


Sutherland-hodgman area clipping algorithm

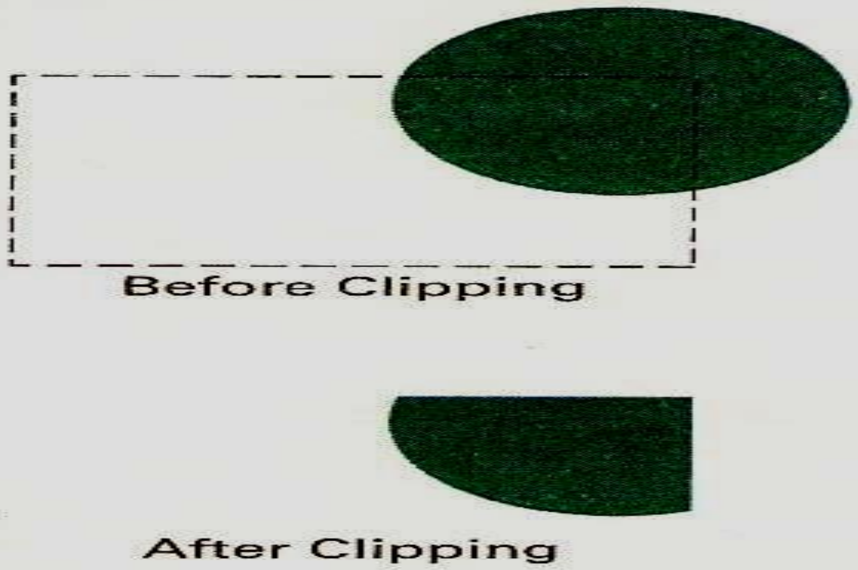
A technique for clipping areas developed by Sutherland & Hodgman

Put simply the polygon is clipped by comparing it against each boundary in turn

Sutherland turns up again. This time with Gary Hodgman with whom he worked at the first ever graphics company Evans & Sutherland

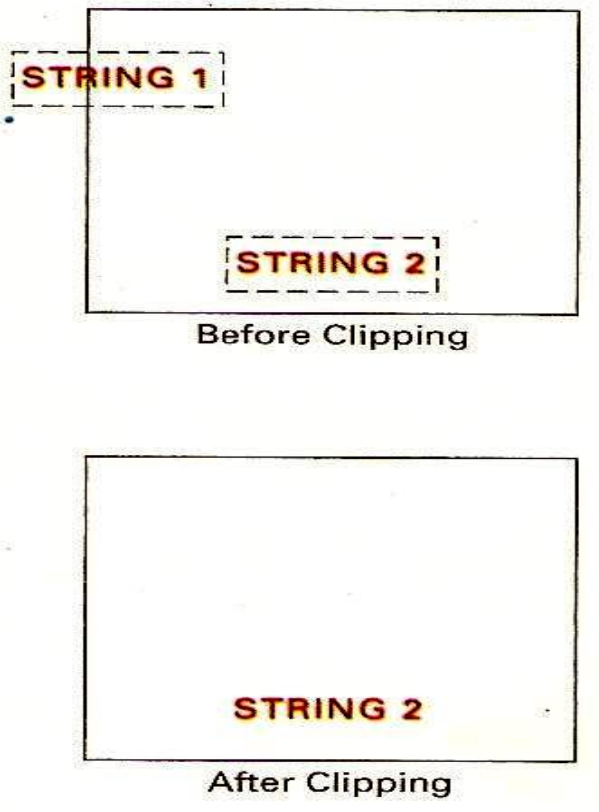


2.Curve clipping

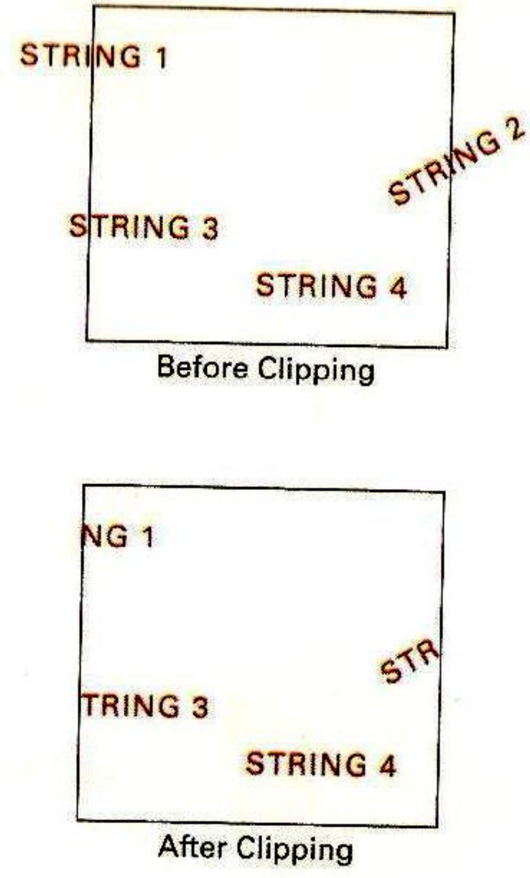


Clipping a filled circle.

TEXT CLIPPING



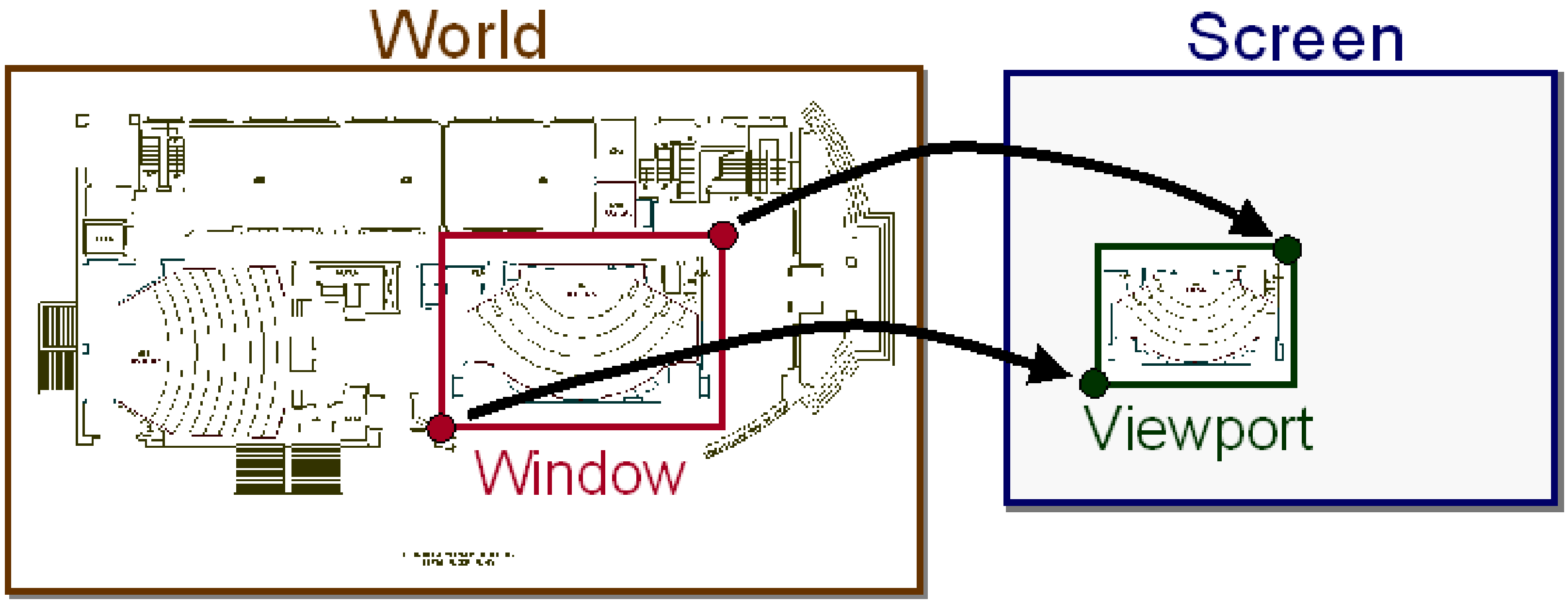
Text clipping using a bounding rectangle about the entire string.



Text clipping using a bounding rectangle about individual characters.

2D Viewing Transformation

- Transform 2D geometric primitives from application's world coordinate system to device's screen coordinate system



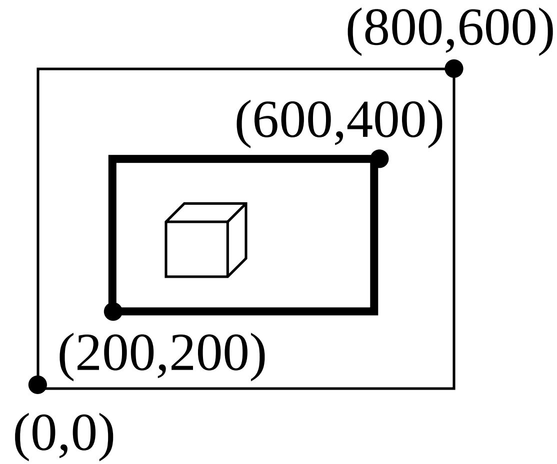
Window and Viewport

Viewport:

Area on screen to be used for drawing.

Unit: pixels (screen coordinates)

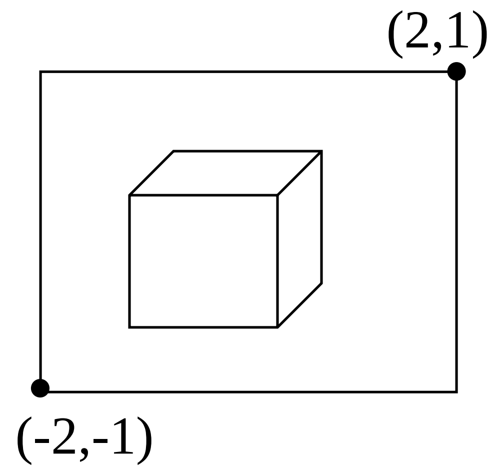
Note: *y*-axis often points down

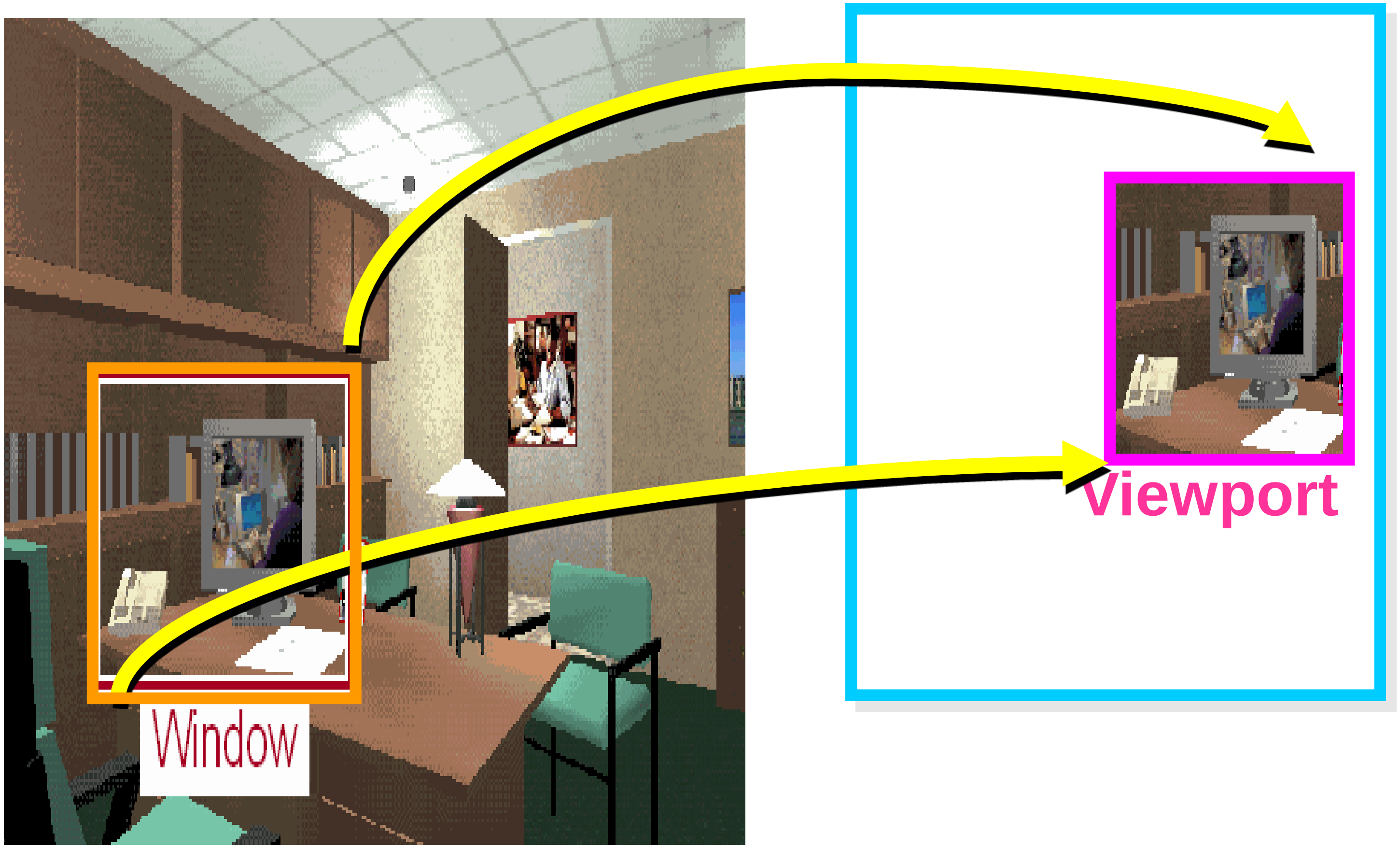


Window:

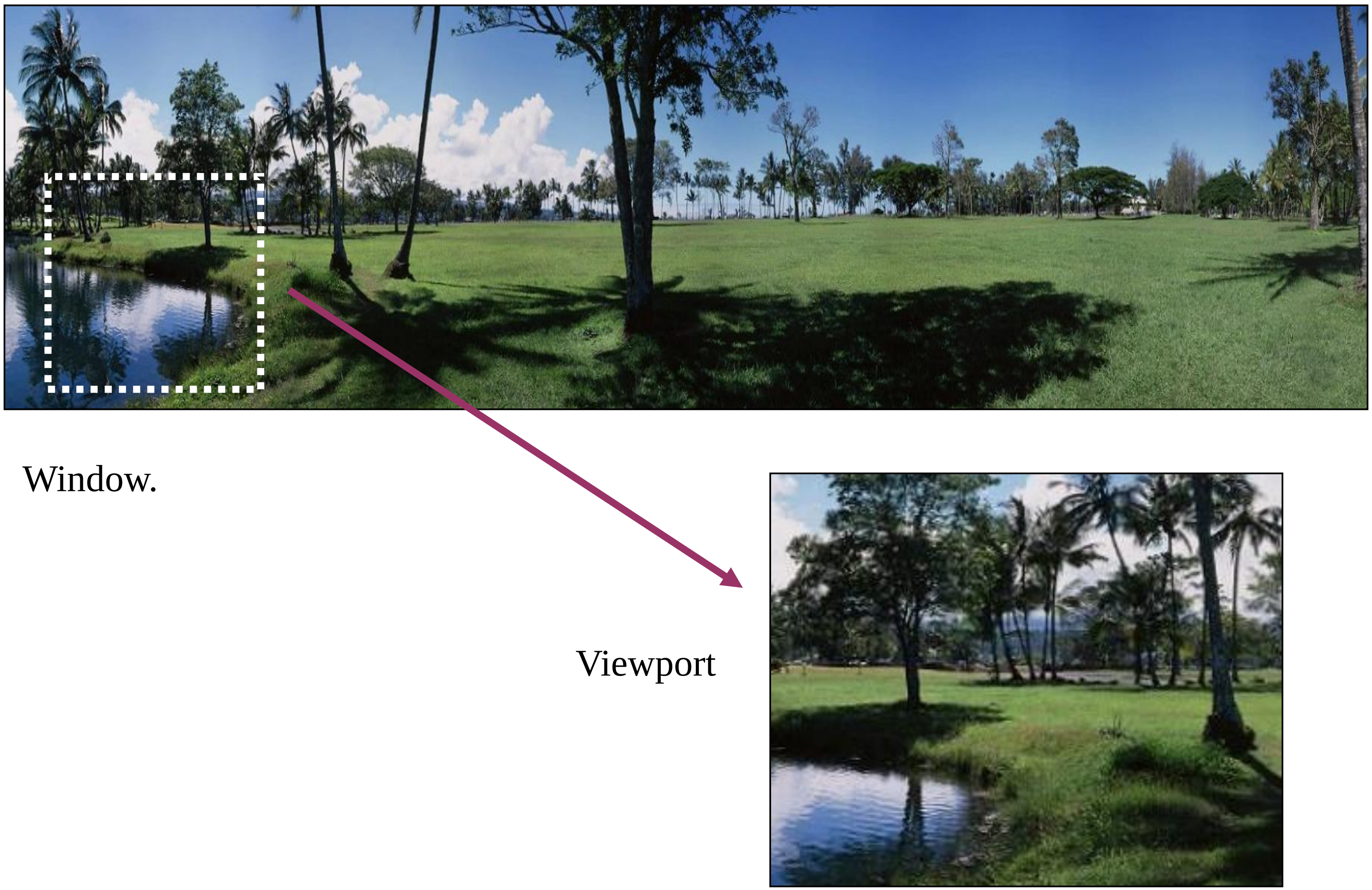
Virtual area to be used by application

Unit: km, mm,… (world coordinates)




Window
Viewport

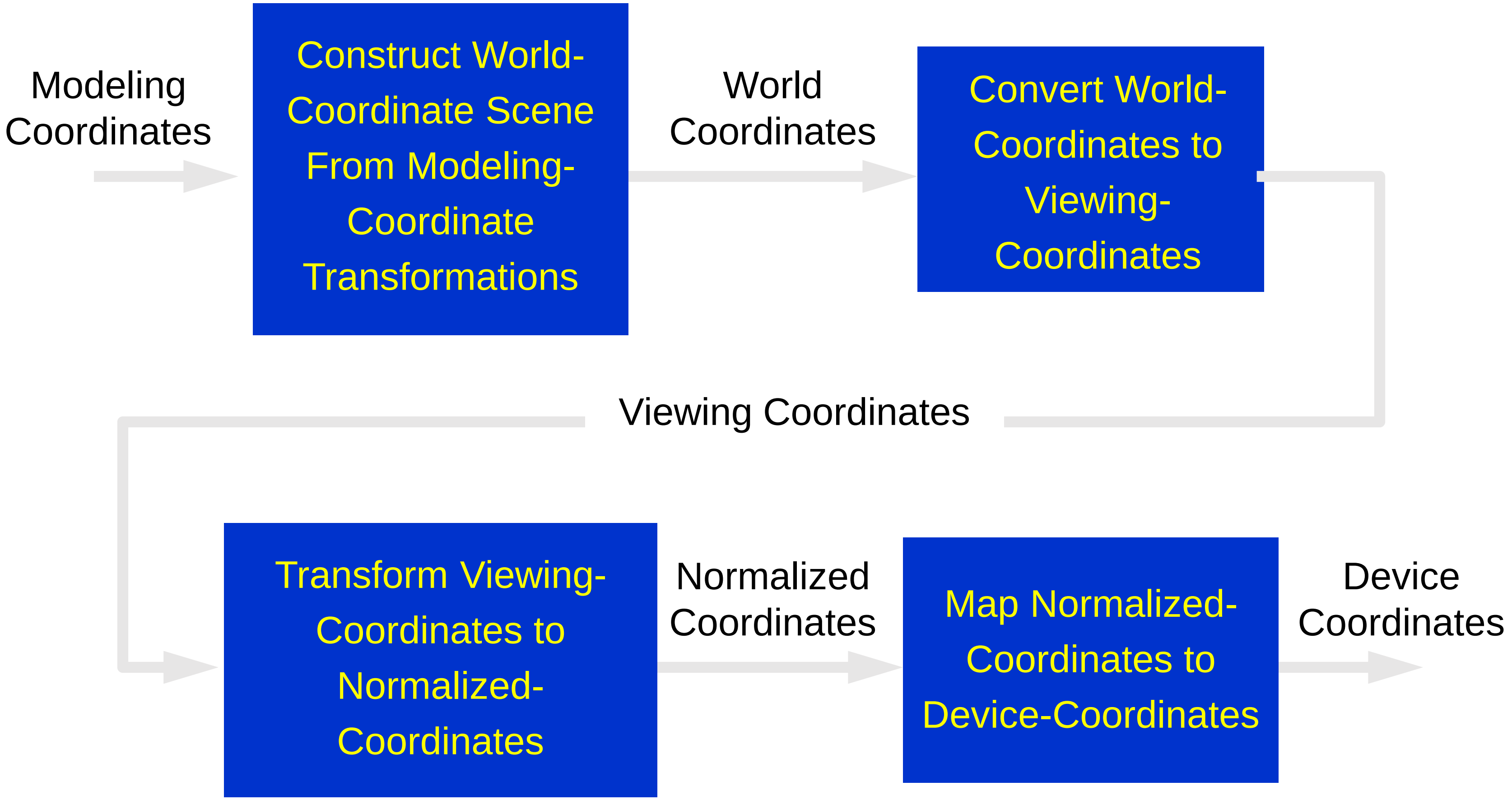
Viewing in 2D - Viewport



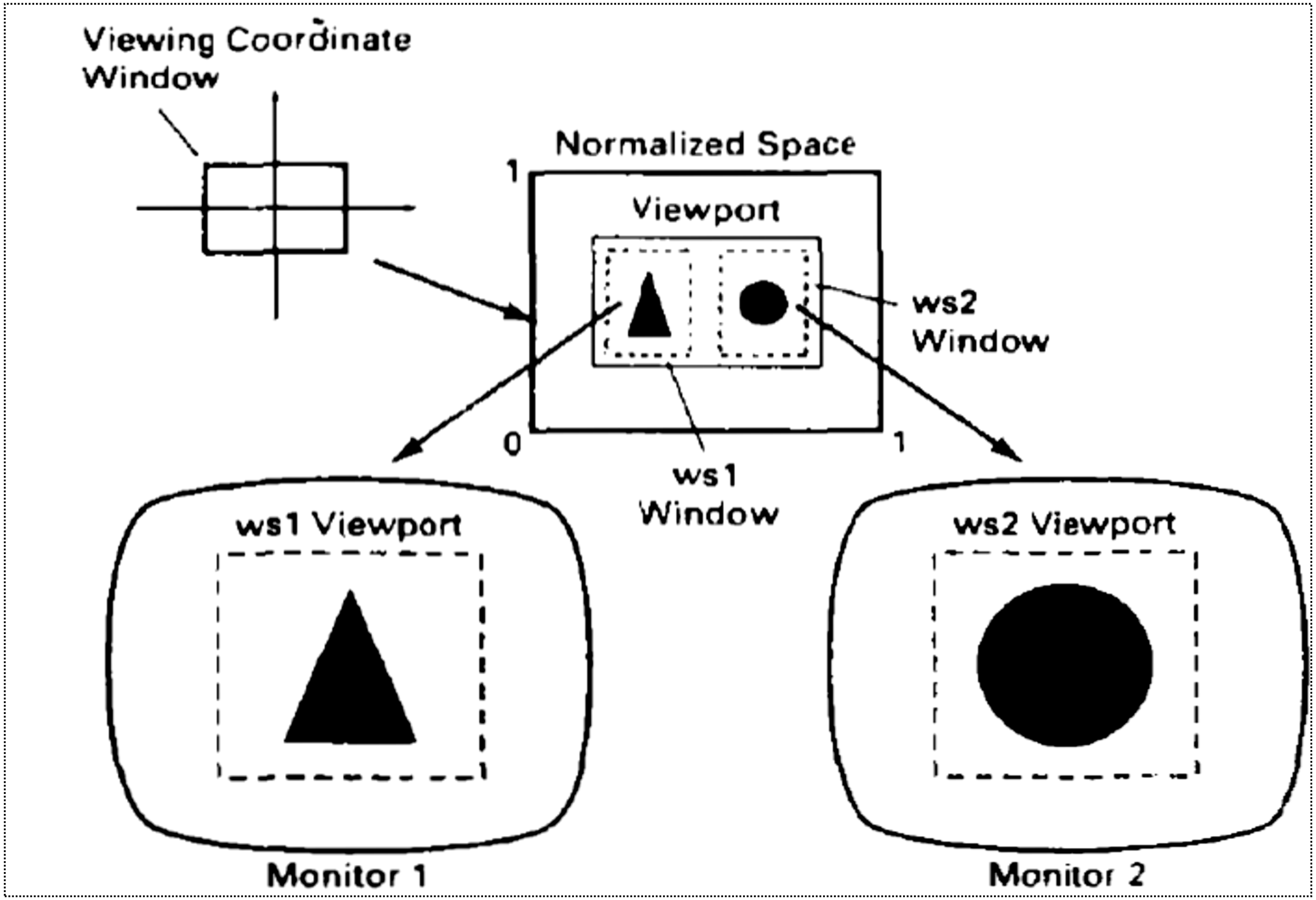
Window -to-Viewport Transformation

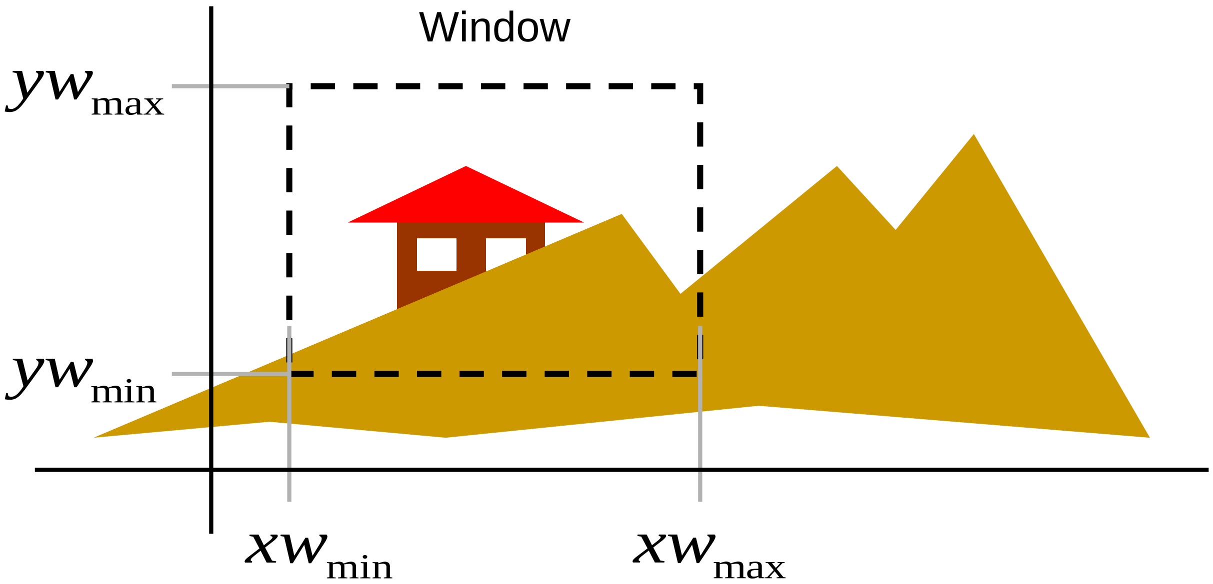
- **Window-to-Viewport mapping** is the process of mapping or transforming a two-dimensional, world-coordinate scene to device coordinates.
- In particular, objects inside the world or clipping window are mapped to the viewport.
- The viewport is displayed in the interface window on the screen.

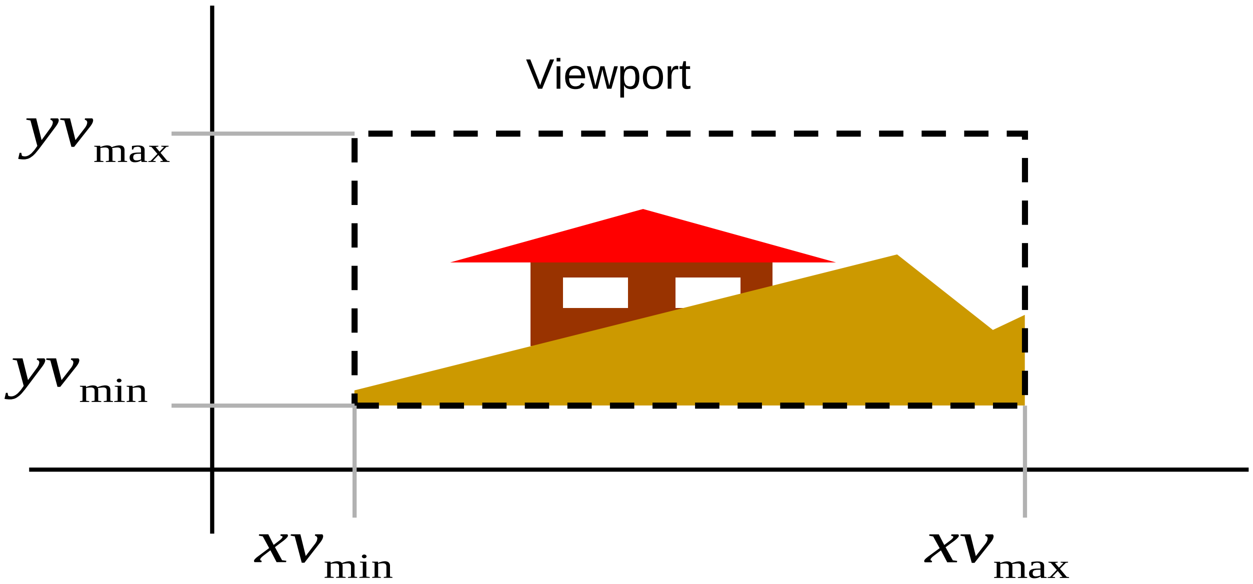
2D viewing transformation pipeline



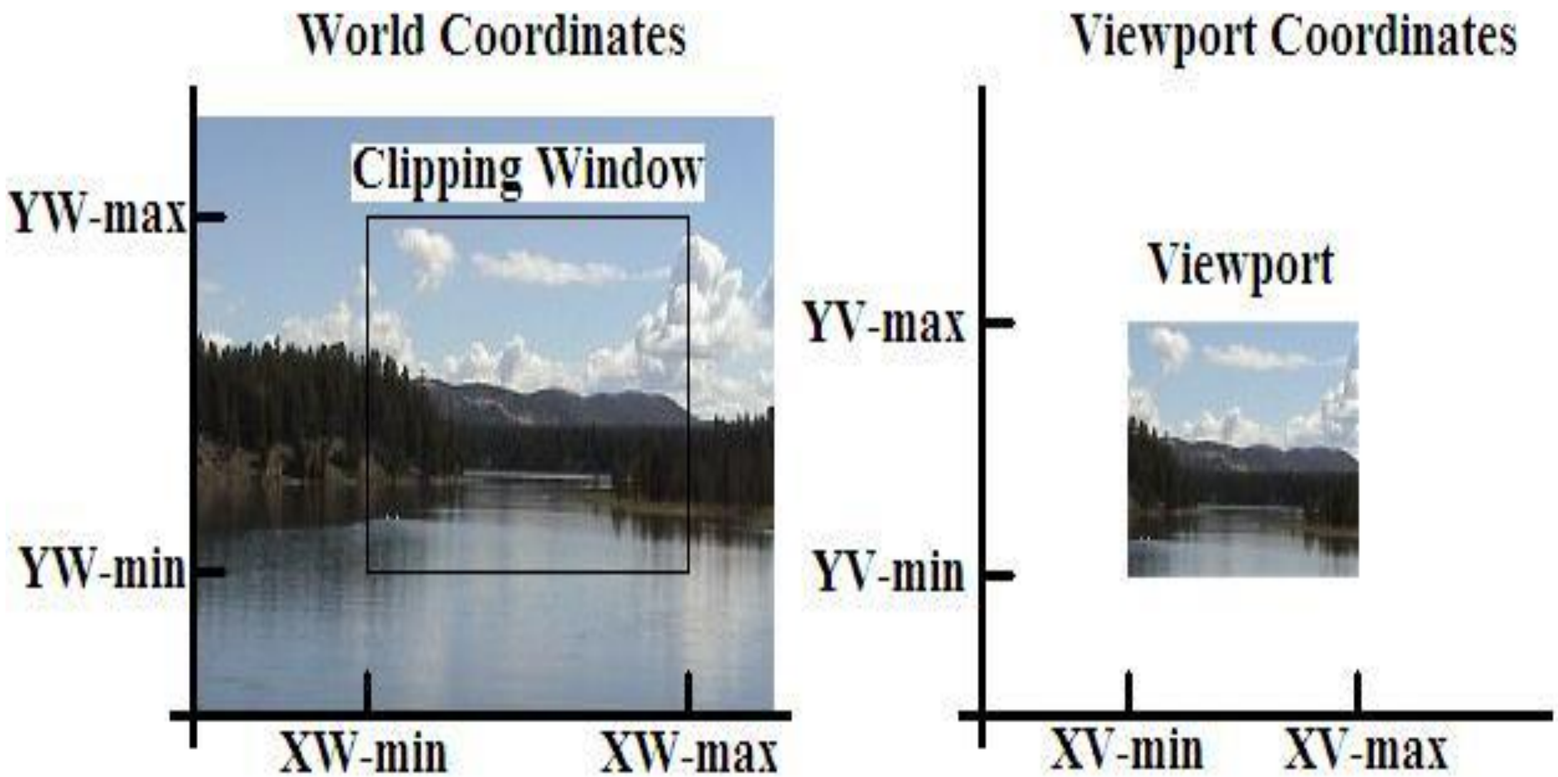
2D Viewing Transformation Representation




Window
yw_{max}
yw_{min}
xw_{min}
xw_{max}


Viewport
yv_{max}
yv_{min}
xv_{min}
xv_{max}

Example:


World Coordinates
Clipping Window
YW-max
YW-min
XW-min
XW-max
Viewport Coordinates
Viewport
YV-max
YV-min
XV-min
XV-max

Unit -2 Geometric Modeling

Representation of curves- Hermite curve- Bezier curve- B-spline curves-rational curves-Techniques for surface modeling – surface patch- Coons and bicubic patches- Bezier and B-spline surfaces. Solid modeling techniques- CSG andB-rep

Unit -2 Geometric Modeling

- Curves
- Surfaces
- Solid

Introduction to curves

Curve Entities

All CAD/CAM systems provide users with curve entities

Curve entities are divided into two categories,

Analytic

The Curves Which are defined as those that can be expressed by analytic equation.

Points, lines, arcs, fillets, chamfers, and conics (ellipses, parabolas, and hyperbolas)

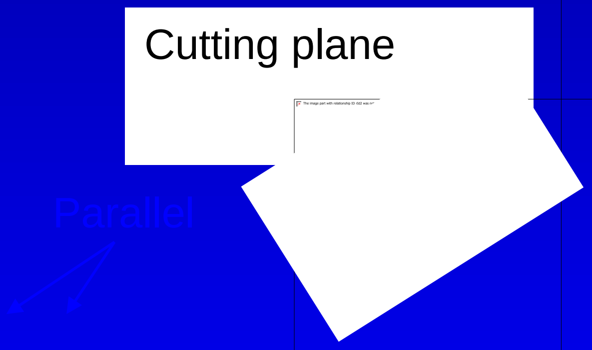
Synthetic

The curves which are described by a set of data points or the control points such as spline; Cubic spline, B-spline and Bezier curve

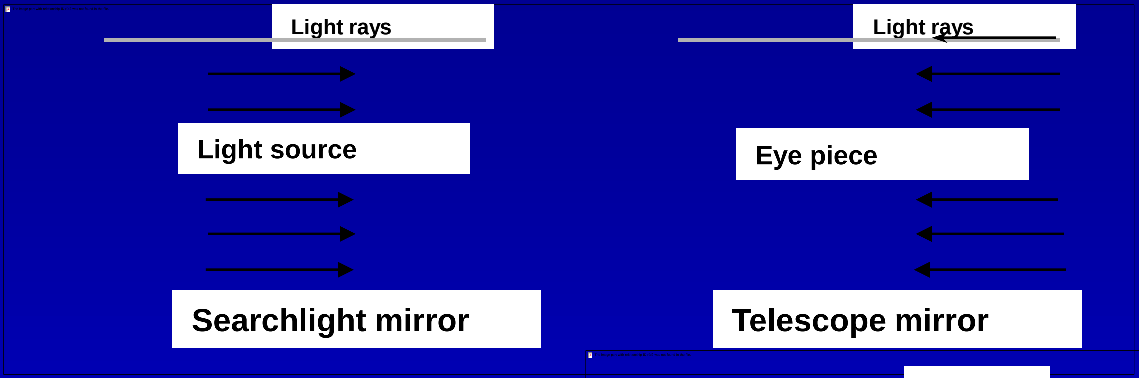
Conic Curves - Parabolas

Conic curves or conics are the curves formed by the intersection of a plane with a right circular cone (parabola, hyperbola and sphere).

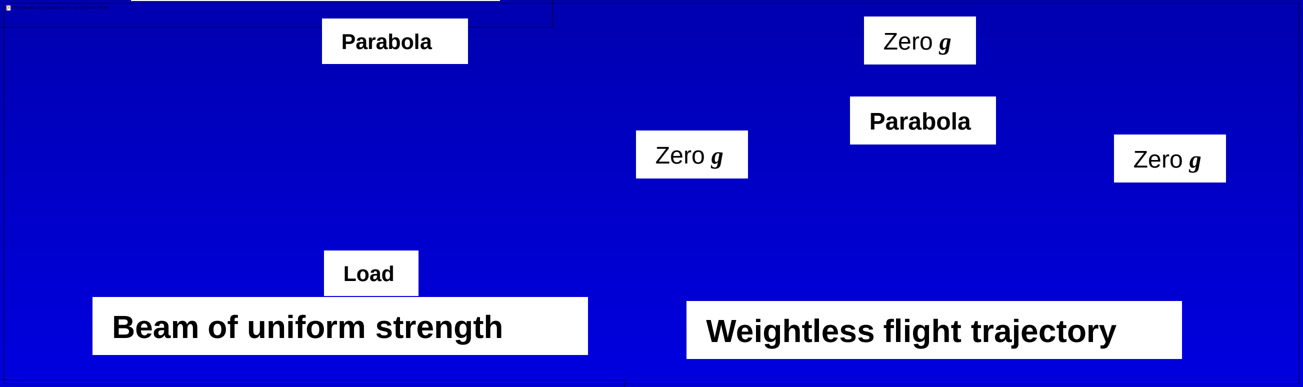
A ***parabola*** is the curve created when a plane intersects a right circular cone parallel to the side (elements) of the cone



Conic Curves - Parabolas



Applications of ***parabola***



A parabola revolved about its axis creates a surface called paraboloid. An auditorium ceiling in shape of paraboloid reduces reverberations if the speaker stands near the focus

Conic Curves - Hyperbolas

A ***hyperbola*** is the curve created when a plane parallel to the axis and perpendicular to the base intersects a right circular cone.

Element (side)

Hyperbola

Orthographic view

Conic Curves - Hyperbolas

Cooling Towers of Nuclear Reactors

The hyperboloid is the design standard for all nuclear cooling towers. It is structurally sound and can be built with straight steel beams.

For a given diameter and height of a tower and a given strength, this shape requires less material than any other form.

Dulles Airport - shape of a hyperbolic paraboloid

Conic Curves - Ellipse

An ***ellipse*** is the curve created when a plane cuts all the elements (sides) of the cone but its not perpendicular to the axis.

Conic Curves - Ellipse

In New York's Grand Central Station, underneath the main concourse there's a special place known as The ***Whispering Gallery.***

The Statuary Hall in the Rotunda (Washington) has a ceiling curved as an ellipse.

Conic Curves - Ellipse

Some tanks are in fact elliptical (not circular) in cross section. This gives them a high capacity, but with a lower center-of-gravity. They're shorter, so that they can pass under a low bridge. You might see these tanks transporting heating oil or gasoline on the highway

Ellipses (or half-ellipses) are sometimes used as fins, or airfoils in structures that move through the air. The elliptical shape reduces drag.

Elliptical gears are used for certain applications

Conic Curves

The image part with relationship ID rId2 was not found in the file.

Curve Entities – Synthetic Curves

Analytical curves are usually not sufficient to meet the design requirements of complex mechanical parts, car bodies, ship hulls, airplane fuselages and wings, shoe insoles, propeller blades, bottles, plastic enclosures for household appliances and power tools, ….

The image part with relationship ID rId2 was not found in the file.

The image part with relationship ID rId2 was not found in the file.

The image part with relationship ID rId2 was not found in the file.

Radio

Thermos

Coffee Press

Types of curve equations

1. Parametric equation:

   Example: circle equation:

   $X = R\cos\theta \quad Y = R\sin\theta \quad z = 0 \quad 0 \leq \theta \leq 2\pi$

   (the coordinates are defined with the help of the extra parameter θ).

2. Non-parametric equation

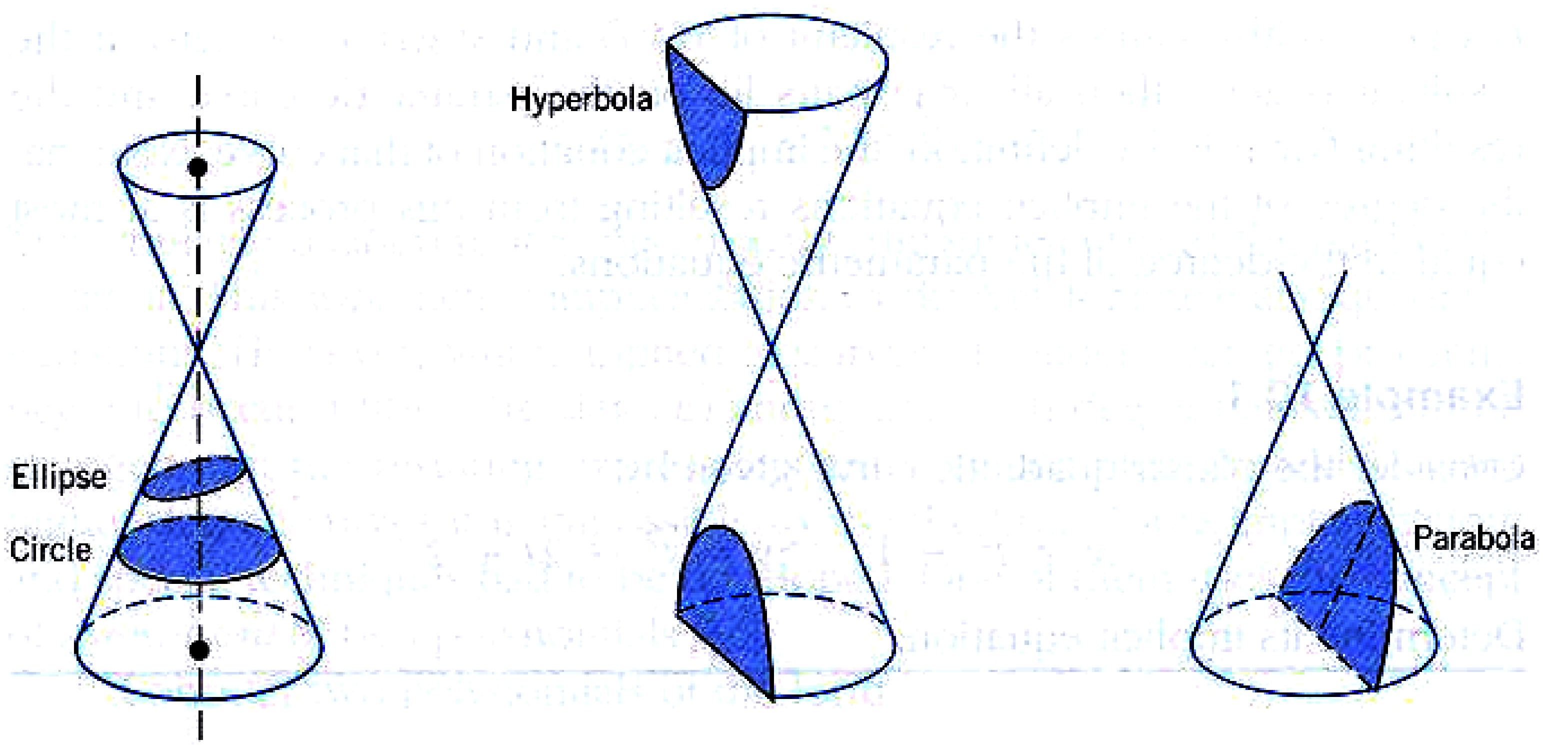
   1. Implicit form: $x^2+y^2-R^2=0 \quad z=0.$
   2. Explicit form $y=\pm\sqrt{(R^2-x^2)} \quad z=0.$

   (these equations defined the x , y and z coordinates without the assistance of extra parameters)

The parametric equation is the most popular form for representing curves and surfaces in CAD systems.

Conic Sections

The curves or portion of curves obtained by cutting a cone with a plane are referred to as conic sections.



Circle or Circular Arc

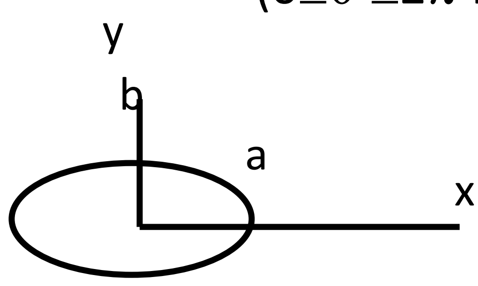
A circle or its portion on the xy plane with radius R and center at (X_c , Y_c) can be represented by the equations:

$x = R\cos\theta + X_c$ $y = R\sin\theta + Y_c$ $(0 \leq \theta \leq 2\pi$ for a circle)

Ellipse or Elliptic Arc

The parametric equation on an ellipse centered at the origin and located in the xy plane. The major axis is in the x direction with length a and the minor axis is in the y direction with length b:

$x = a\cos\theta$ $y = b\sin\theta$ $z = 0$ $(0 \leq \theta \leq 2\pi$ for an ellipse)



Hyperbola

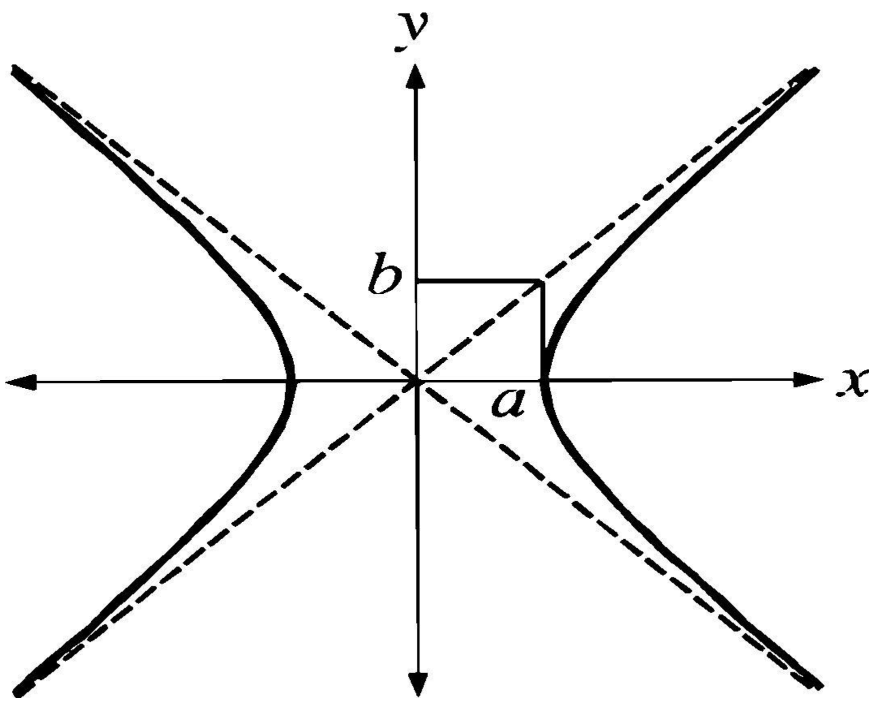
Implicit equation:

$$\frac{x^2}{a^2} - \frac{y^2}{b^2} = 1$$

The parametric equations:

x=a cosh u

y=b sinh u



Parabola

A reference parabola that is symmetric about the x axis and passes through the origin can be represented by the following explicit equation:

$x = cy^2$

This equation can be converted to the following parametric equations:

$x=cu^2$ $y=u$

Types of curves

1. Analytical curves

The curves which are having **rigid form of equation with out any flexibility** to modify its original shapes after display.

Example:

1. Point
2. Line, **Line segment**
3. Conic sections – **Circle, Ellipse**, parabola, hyperbola.
4. Fillet
5. Chamfer

2. Synthetic curves

- The curves which are usually described by **a polynomial equation having flexibility** to modify its original shape after display.
- A **parameter (u)** is used to control its shape and degrees-of-curves.

Examples:

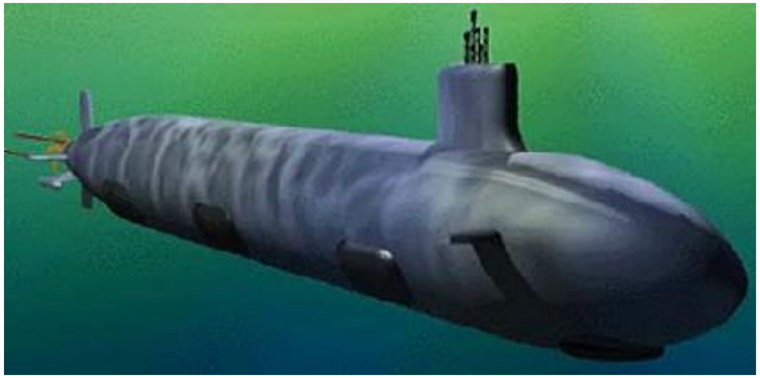
1. Hermite cubic spline curve
2. Bezier curve
3. B-spline curve
4. Rational curve
    a) Rational Hermite cubic spline
    b) Rational Bezier curve
    c) Rational B-spline curve

Synthetic Curves

- Analytic curves are usually not sufficient to meet geometric design requirements of mechanical parts.

  Example: Car bodies, ship hulls, airplane wings, propeller blades, shoe insoles, and bottles etc.






Need for synthetic curves

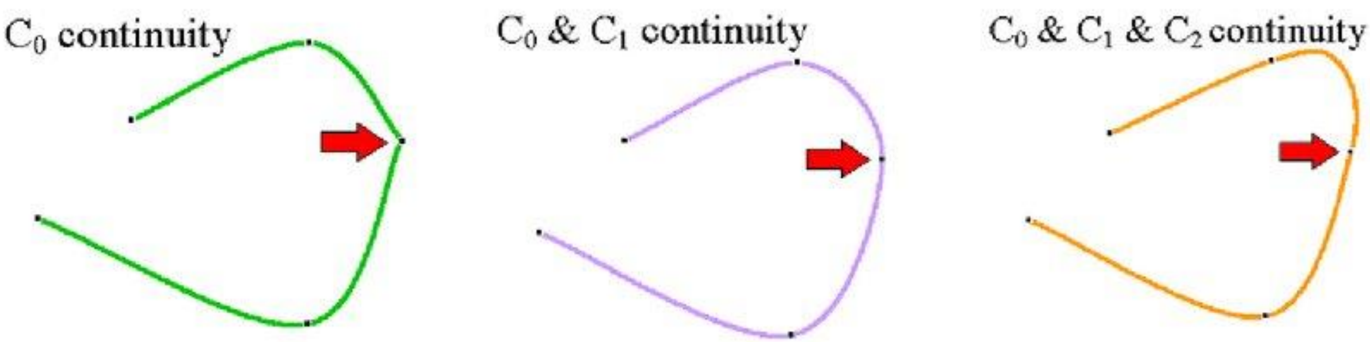
- When a curve is represented by a collection of measured data points.

- When an existing curve must change to meet new design requirements.

Continuity

The smoothness of the connection of two curves or surfaces at the connection points or edges.

- C^0: simple connection of two curves
- C^1: the geometric slopes at the joint must be same
- C^2: curvature continuity that not only the gradients but also the center of curvature is the same



1. Hermite cubic spline curve

- Parametric spline curves are defined as **piecewise polynomial curves** with certain order of continuity.
- The parametric cubic spline curve curves connects two data (end) points and utilizes a cubic equation.

General condition required:

1. Two end points
2. Two end slopes

- The parametric equation of a cubic spline segment is given by:

$$P(u) = \sum_{i=0}^{3} C_i u^i \qquad 0 \leq u \leq 1$$

where, u – parameter

C_i - Polynomial coefficients

The scalar form of equation:

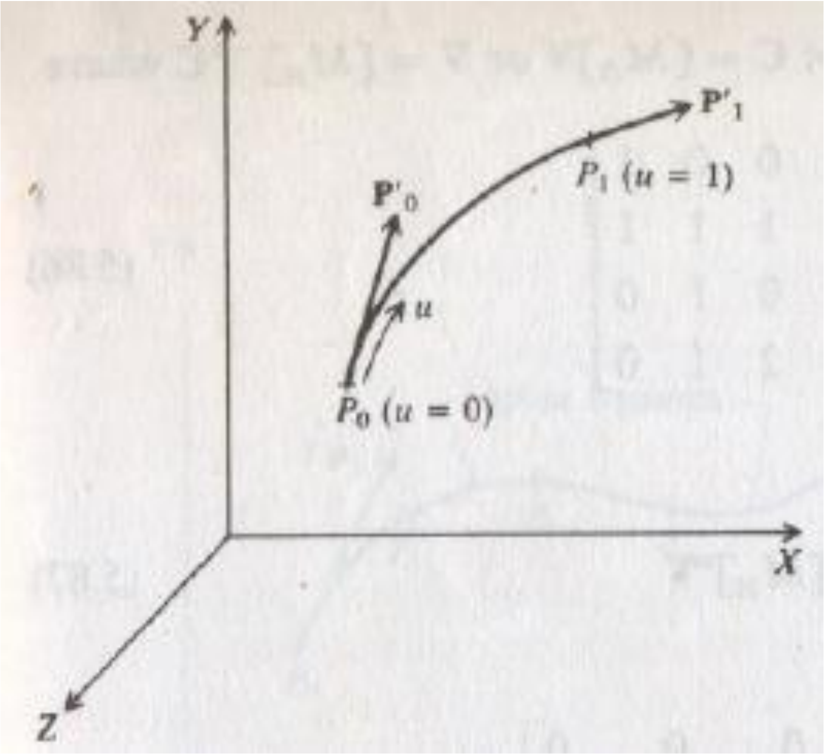
$X(u) = C_{3x} u^3 + C_{2x} u^2 + C_{1x} u + C_{0x}$

$y(u) = C_{3y} u^3 + C_{2y} u^2 + C_{1y} u + C_{0y}$

$z(u) = C_{3z} u^3 + C_{2z} u^2 + C_{1z} u + C_{0z}$

The vector form of equation:

$P(u) = C_3 u^3 + C_2 u^2 + C_1 u + C_0$



P_0 – Starting point

P_1 – End point

P'_0 – Starting slope

P'_1 – Starting slope

The matrix form of equation:

$$P(u) = U^T C$$

where,

$U = [u^3 \;\; u^2 \;\; u^1 \;\; 1]^T$

$C = [C_3 \;\; C_2 \;\; C_1 \;\; C_0]^T$ (coefficient vector)

Tangent Vector

$$P'(u) = \sum_{i=0}^{3} C_i \, i \, u^{i-1} \qquad 0 \le u \le 1$$

To find the coefficients C_i

Apply the known boundary conditions

P_0 , P'_0 at $u = 0$

P_1 , P'_1 at $u = 1$

Position and slope equation are:

$P(u) = C_3u^3 + C_2u^2 + C_1u + C_0$

$P'(u) = 3C_3u^2 + 2C_2u + C_1$

When applying u = 0 and u = 1 on the position and slope equations:

- $P_0 = C_0$
- $P'_0 = C_1$
- $P_1 = C_3 + C_2 + C_1 + C_0$
- $P'_1 = 3C_3 + 2C_2 + C_1$

After solving the above four equations by simultaneous solution method the coefficients are:

$$C_0 = P_0$$

$$C_1 = P'_0$$

$$C_2 = 3(P_1 - P_0) - 2(P'_0 - P'_1)$$

$$C_3 = 2(P_0 - P_1) + P'_0 + P'_1$$

After substituting all four coefficient, the final **blending functions** are:

$$P(u) \quad = \quad (2u^3 - 3u^2 + 1)P_0 + (-2u^3 + 3u^2)\ P_1 + (u^3 - 2u^2 + u)P'_0 + (u^3 - u^2)\ P'_1$$

$$P'(u) \quad = \quad (6u^3 - 6u)P_0 + (-6u^2 + 6u)\ P_1 + (3u^2 - 4u + 1)P'_0 + (3u^2 - 2u)\ P'_1$$

$P(u) = U^T C$

$P(u) = U^T [M_H] V$

Where,

$[M_H]$ = Hermite matrix

V= Geometry (Boundary) vector

$$[M_H] = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

$V = [P_0 \ P_1 \ P'_0 \ P'_1]^T$

Comparing the above two equations:

$$U^T C = U^T [M_H] V$$

$$C = [M_H] V$$

Final position and slope equation in the matrix are:

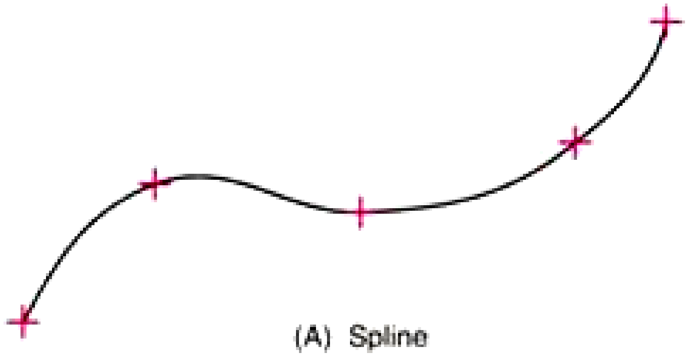
$$P(u) = U^T [M_H] V$$

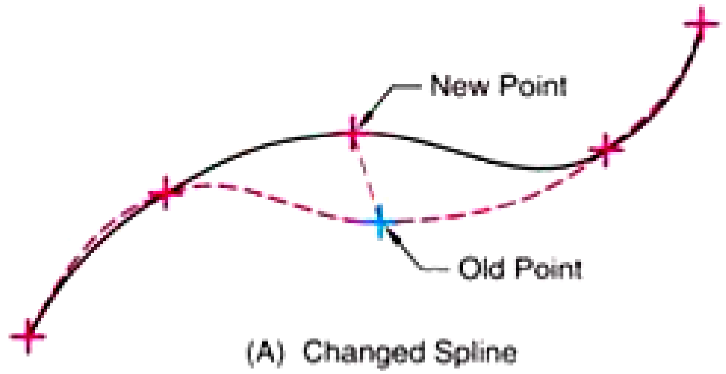
$$P'(u) = U^T [M_H]^u V$$

Modification of resultant curve shape

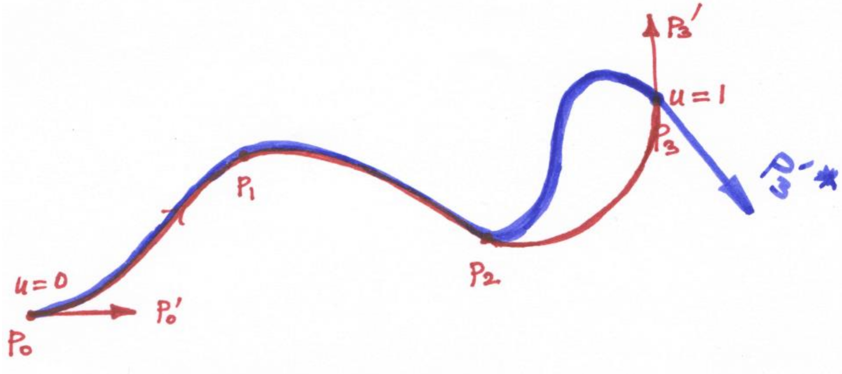
1. By changing control point(s)
2. By changing end slope(s)

1. By changing control point(s)

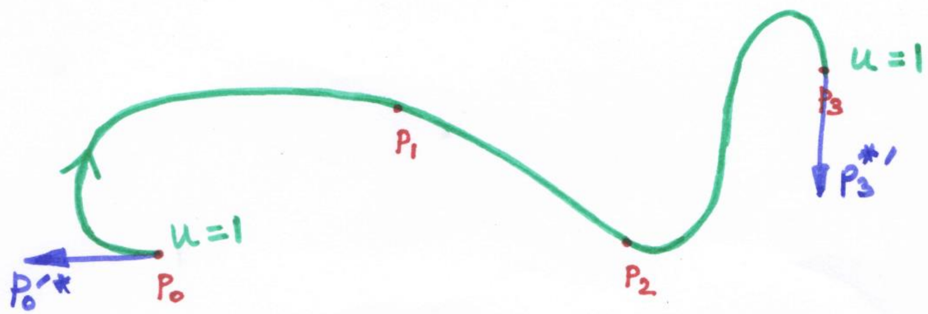




2. By changing end slope(s)

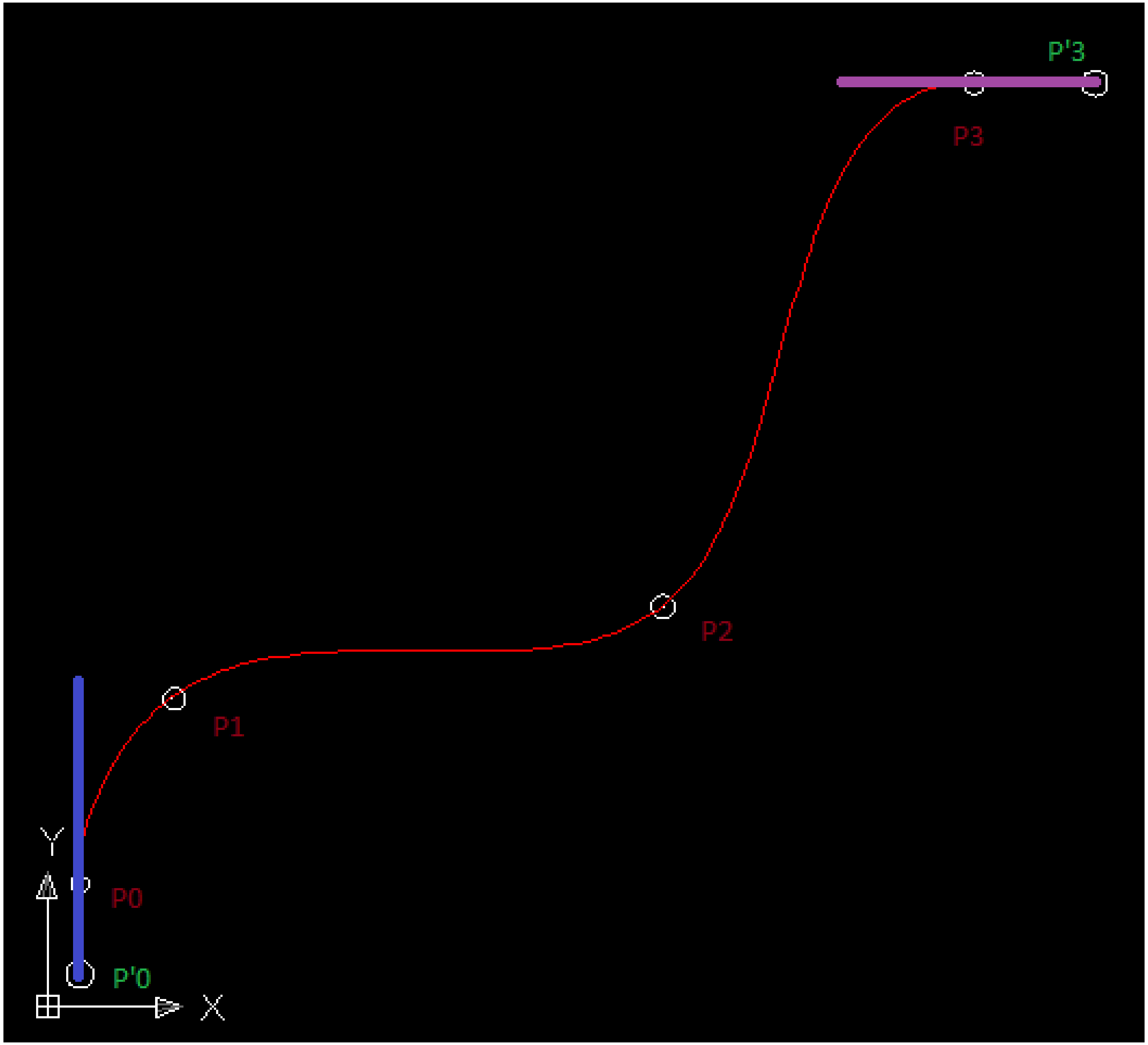


By changing one slope



By changing two slopes

Hermite Cubic Spline curve using AutoCAD software



Characteristics of Hermite cubic spline curve

1. The resultant shape will pass through all the given data or control points.
2. It uses interpolation technique for curve general.
3. The resultant curve has tangential property with start and end slopes.
4. When reversing the parametric direction, the shape of the resultant curve not be altered.
5. It has only global control.
6. The resultant curve has always cubic curve.

Limitations

1. It has only global control (or) Lack of local.
2. Always it is a cubic curves.
3. It is not possible to apply, when higher degree of curves required.

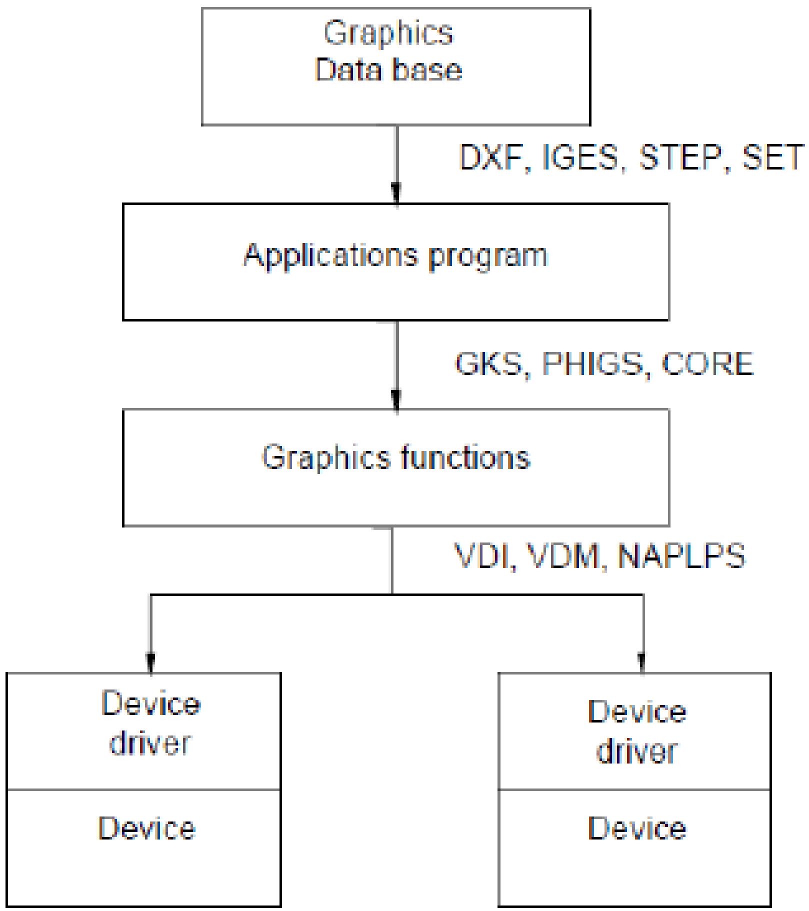
UNIT - III

- **CAD STANDARDS**

Standardization in Graphics

- GKS (Graphical Kernel System)
- PHIGS (Programmer's Hierarchical Interface for Graphics)
- CORE (ACM-SIGGRAPH)
- GKS-3D
- IGES (Initial Graphics Exchange Specification)
- DXF (Drawing Exchange Format)
- STEP (Standard for the Exchange of Product Model Data)
- DMIS (Dimensional Measurement Interface Specification)
- VDI (Virtual Device Interface)
- VDM (Virtual Device Metafile)
- GKSM (GKS Metafile)
- NAPLPS (North American Presentation Level Protocol Syntax)

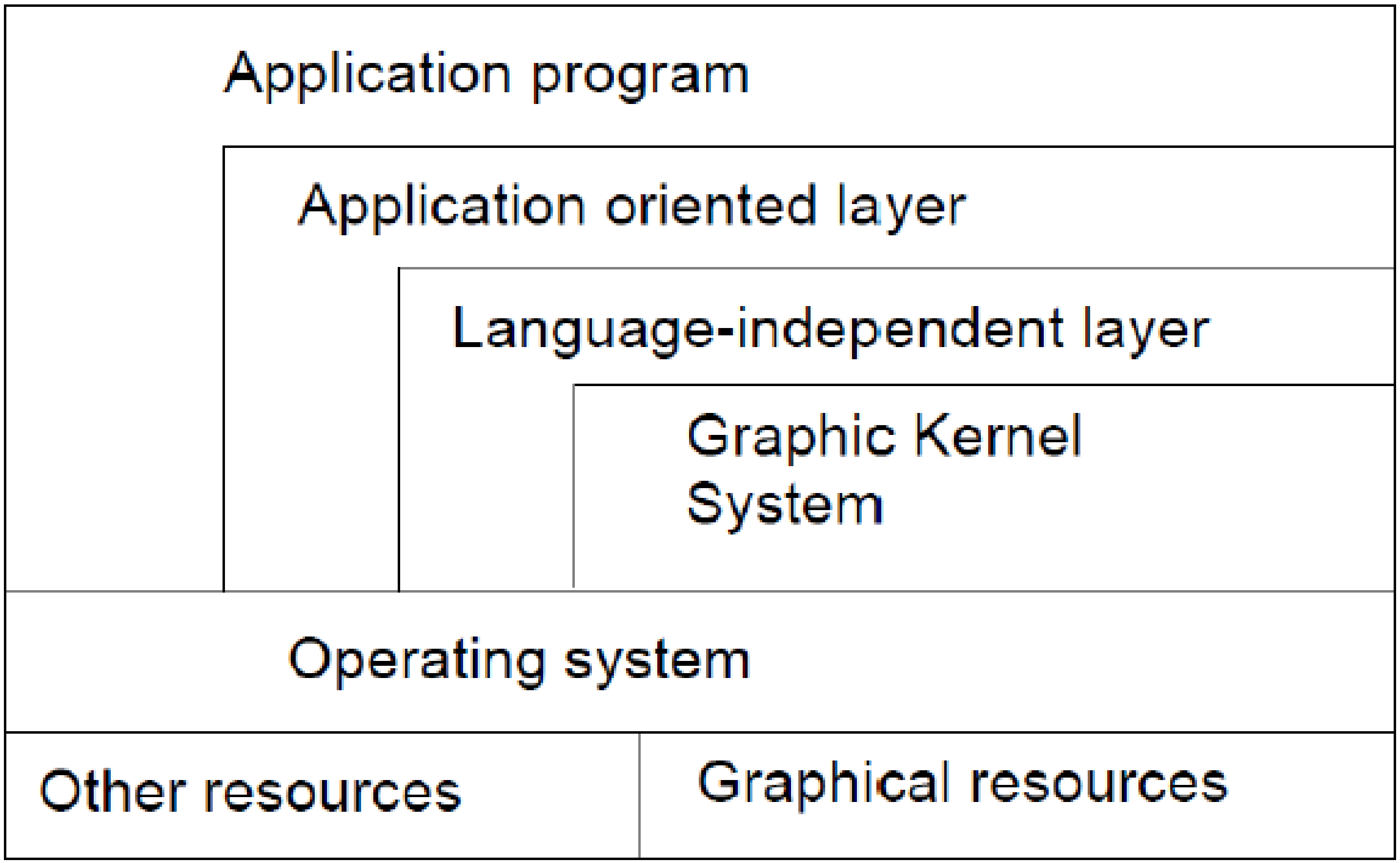
Various standards in graphics programming



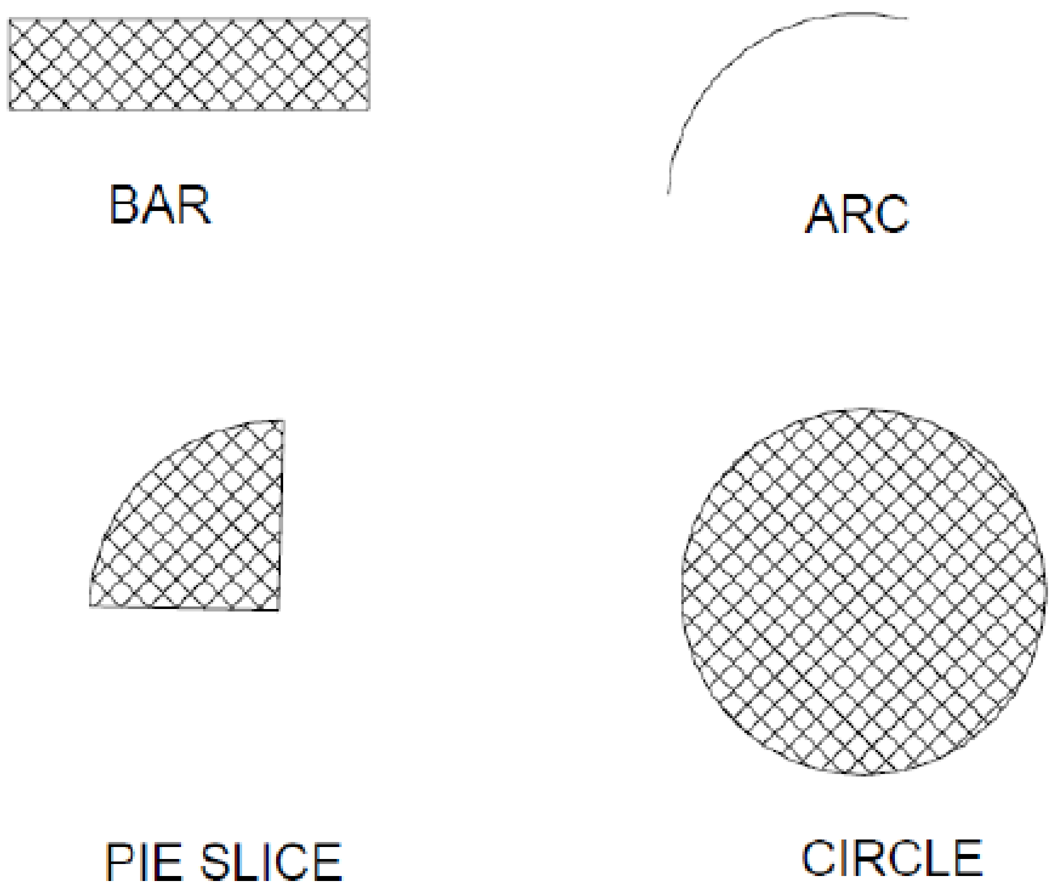
objectives

- The main objectives that were put forward for GKS are:
- –To provide the complete range of graphical facilities in 2D, including the interactive capabilities,
- To control all types of graphic devices such as plotters and display devices in a consistent manner,
- To be small enough for a variety of programs

Layer model of Graphics Kernel System

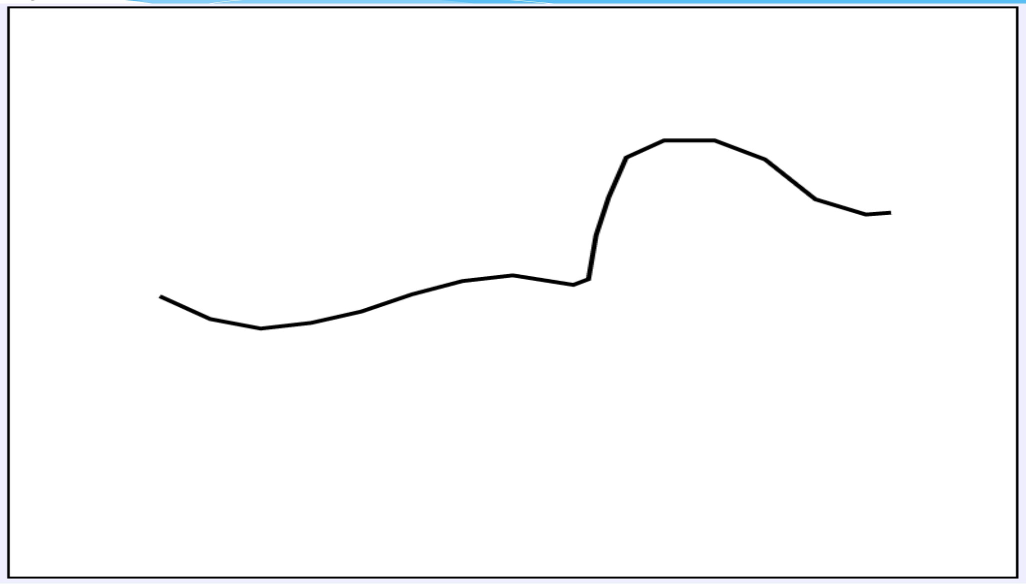


Graphics primitives in IBM GKS



Primitives in GKS

- polyline:. The GKS function for drawing line segments is called polyline.

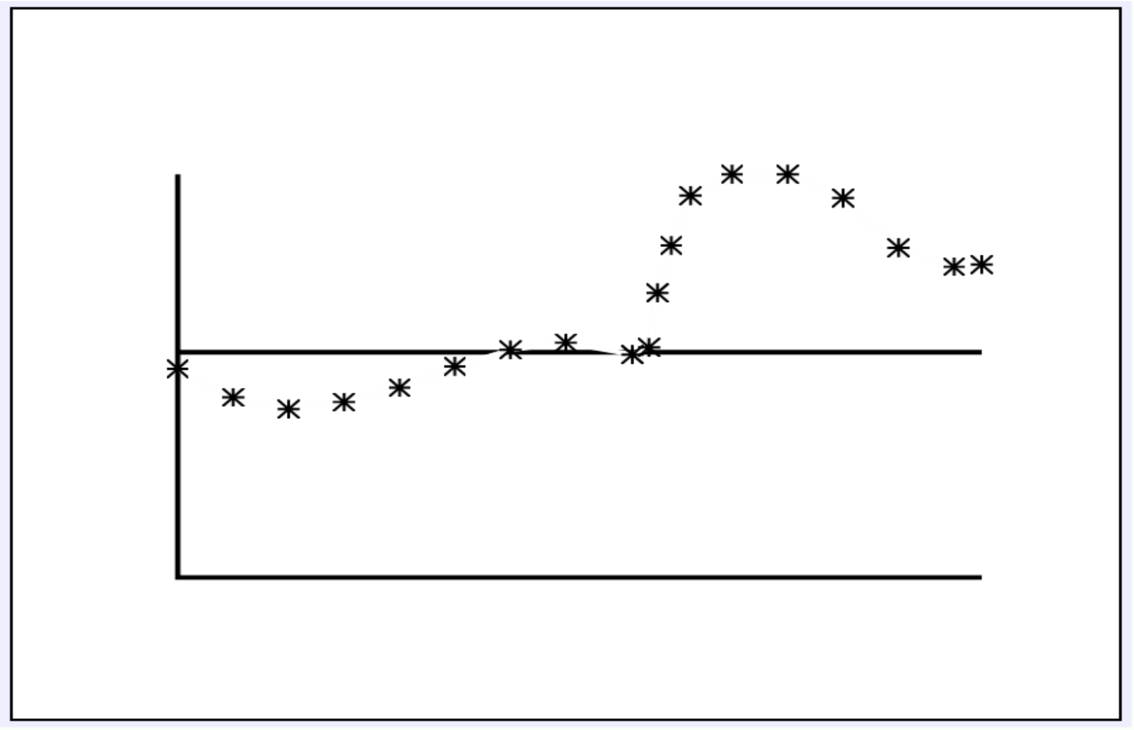


POLYLINE(N, XPTS, YPTS)

- The polyline function takes an array of X-Y coordinates and draws line segments connecting them

* **POLY MARKER**: which marks a sequence of points with the same symbol.

POLYMARKER(N, XPTS, YPTS)



* **FILL AREA:** which displays a specified area.

  FILL AREA(N, XPTS, YPTS)

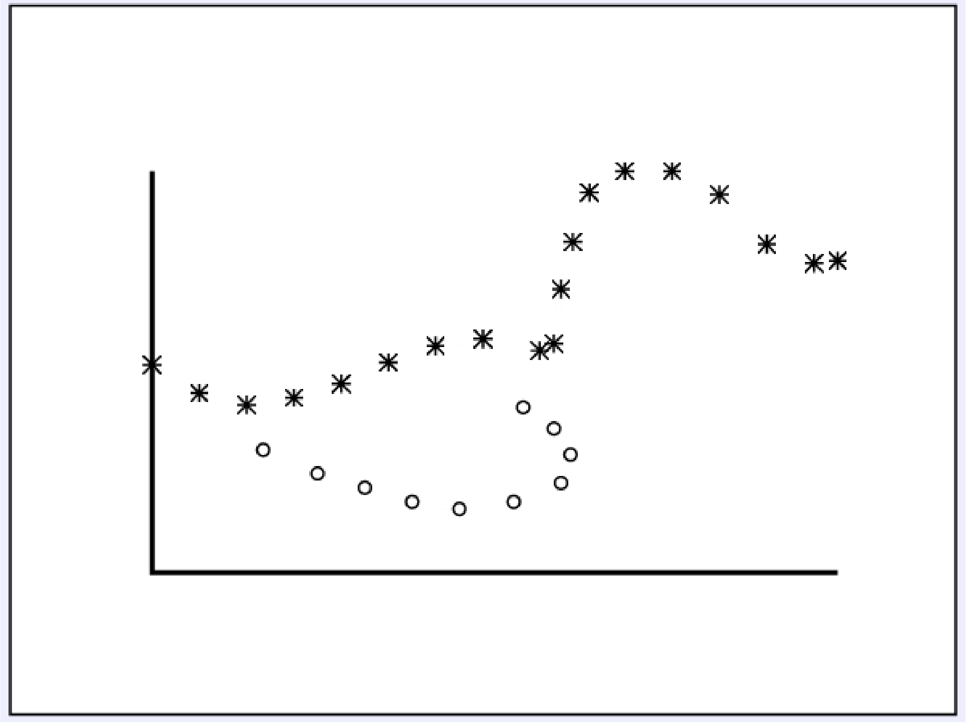
* **TEXT:** which draws a string of characters.

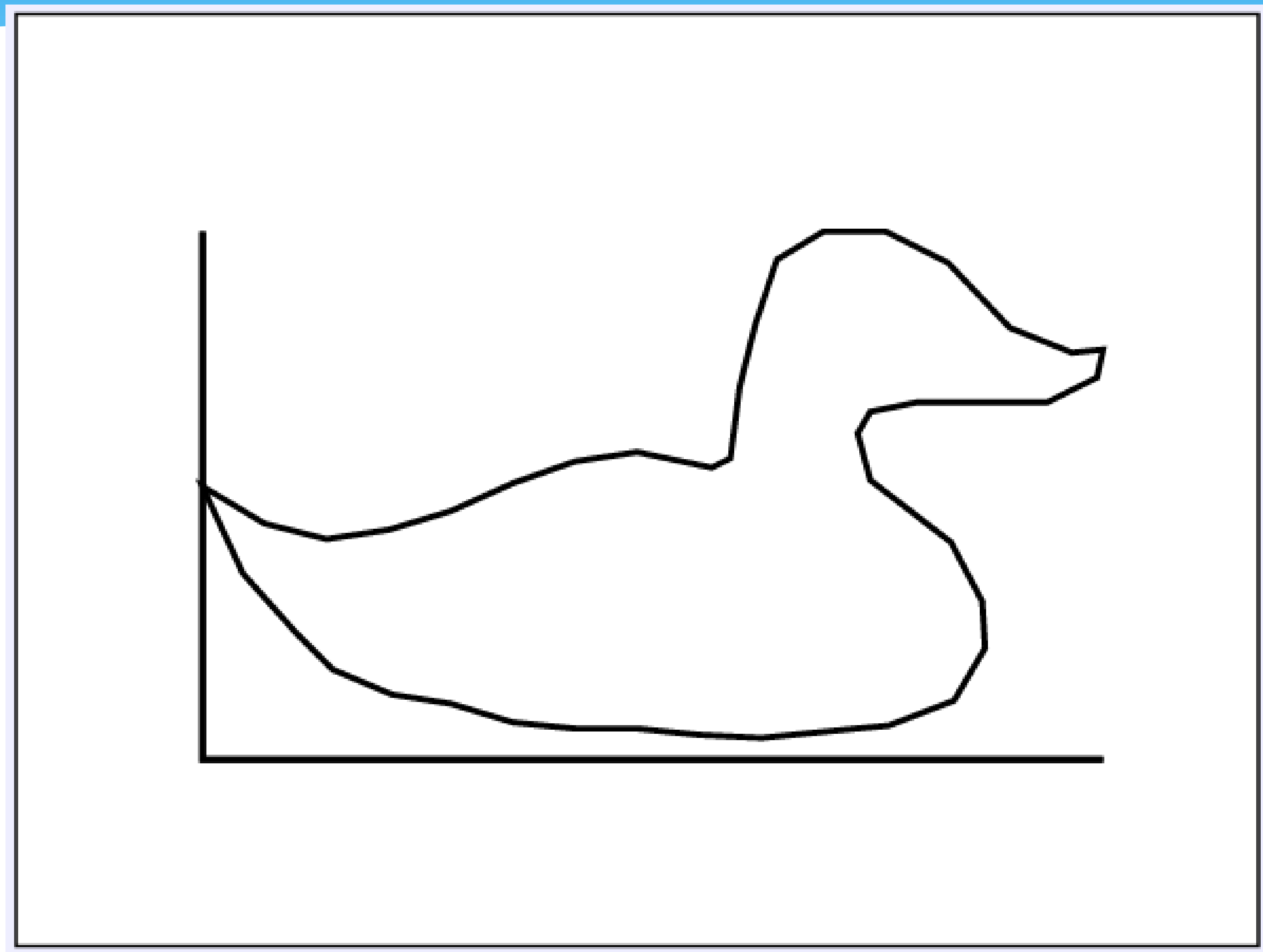
  TEXT(X, Y, STRING)

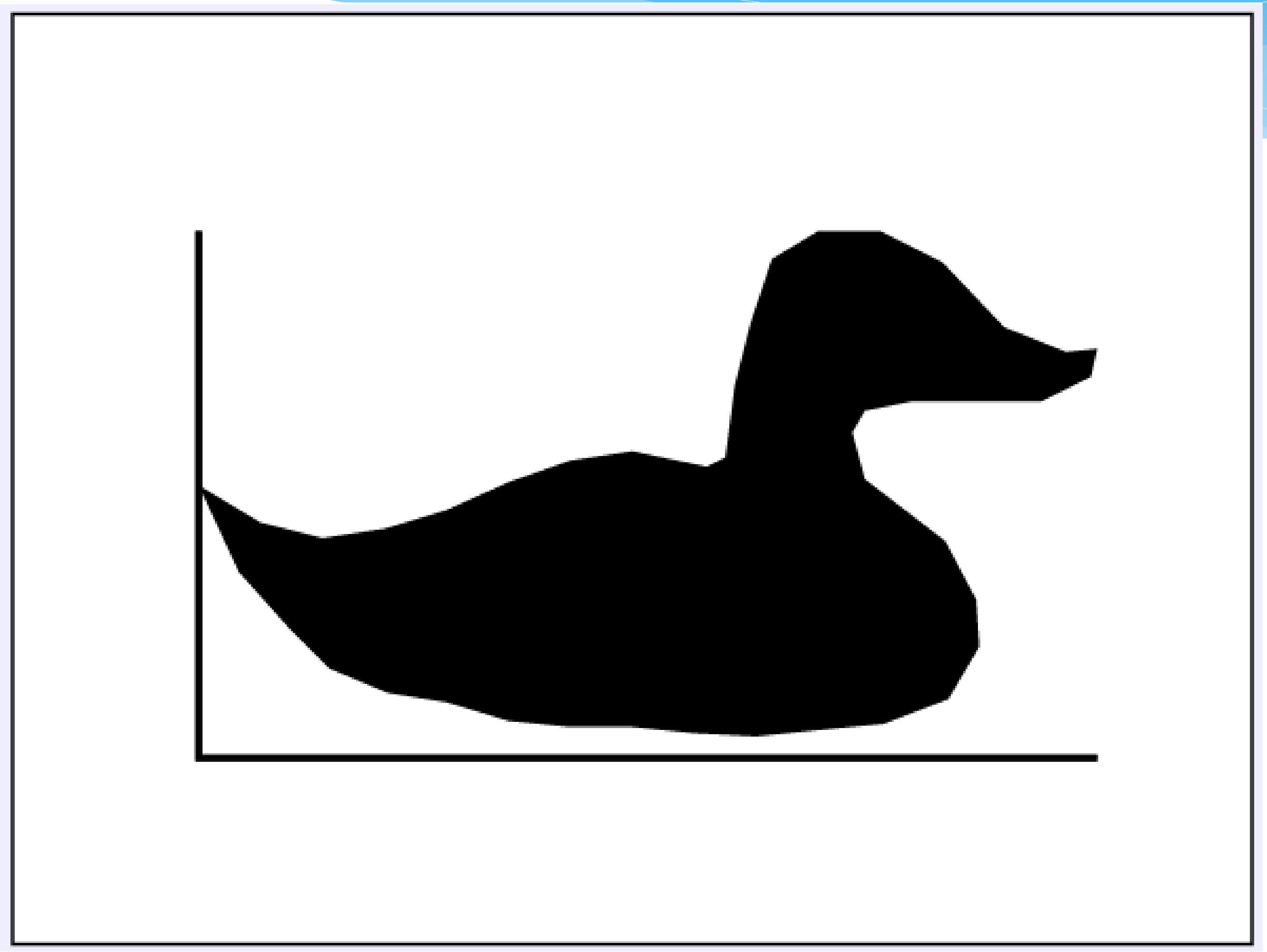
An example of the text primitive is:

TEXT(6, 3, 'A Character String')

Drawing duck using gks primitives







GKS 3D

* the three-dimensional extension of GKS.
* allows the production of 3-D objects.
* handle 3D primitives, 3D input, and 3D viewing.

The Drawing Primitives

* **Polyline**

  3DCALL GPL3(N, PXA, PYA, PZA)

* **Polymarker**

  3DCALL GPM3(N, PXA, PYA, PZA)

* **Fill Area**

  3DCALL GFA3(N, PXA, PYA, PZA)

EXCHANGE OF MODELING DATA

- Necessity to translate drawings created in one drafting package to another often arises.

PRO-E ↔ ANSYS

- One method is to write direct translators from one software to another, which has to be produced by system developer.

EXCHANGE OF MODELING DATA

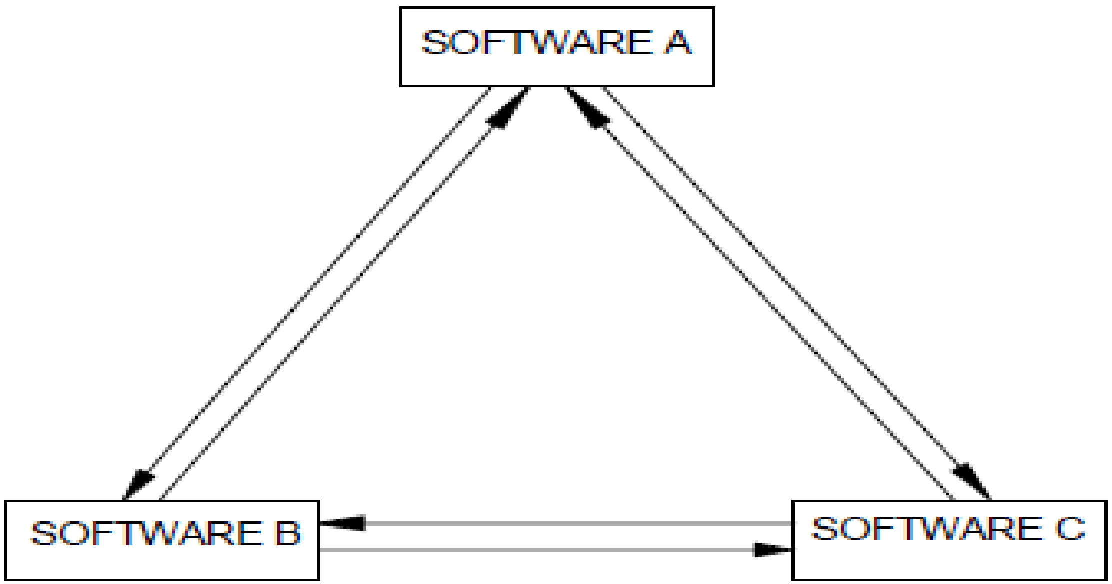


Fig. 17.3 Direct Data Translation

- If we have three software packages we may require six translators among them.
- This will necessitate a large number of translators

IGES

IGES

- **IGES - INITIAL GRAPHICS EXCHANGE SPECIFICATION**
- IGES version 1.0 was released in 1980
- IGES converts the CAD model into neutral file.
- Conversion is done by preprocessors inbuilt in the software.

IGES

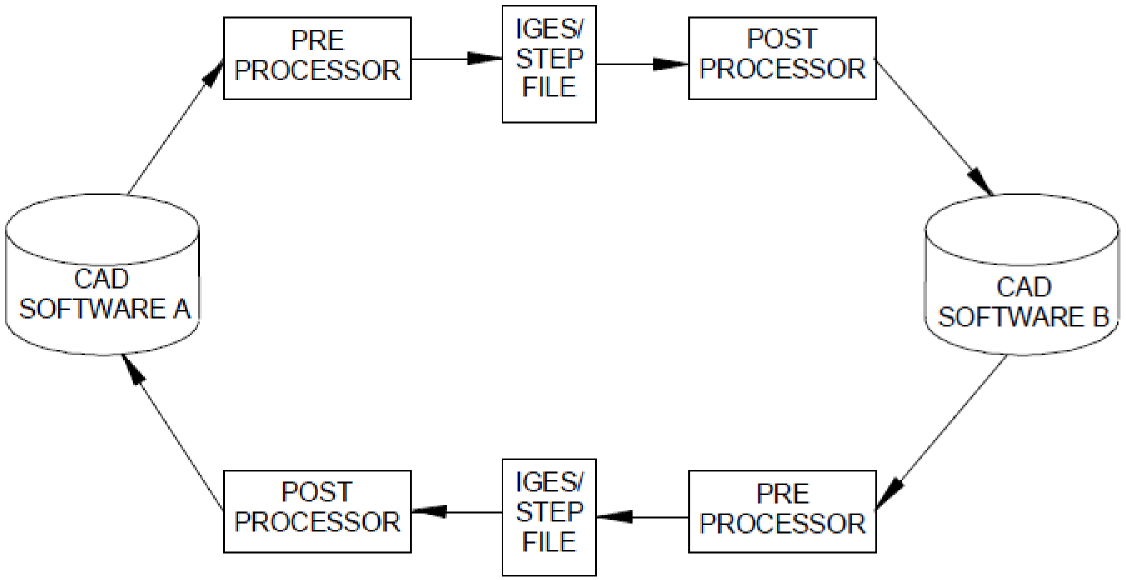


Fig. 17.4 CAD Data Exchange Using Neutral Files

Entities

101: circular arc

108: plane

110: line

116: point

120: tabulated cylinder

134: node

158: sphere

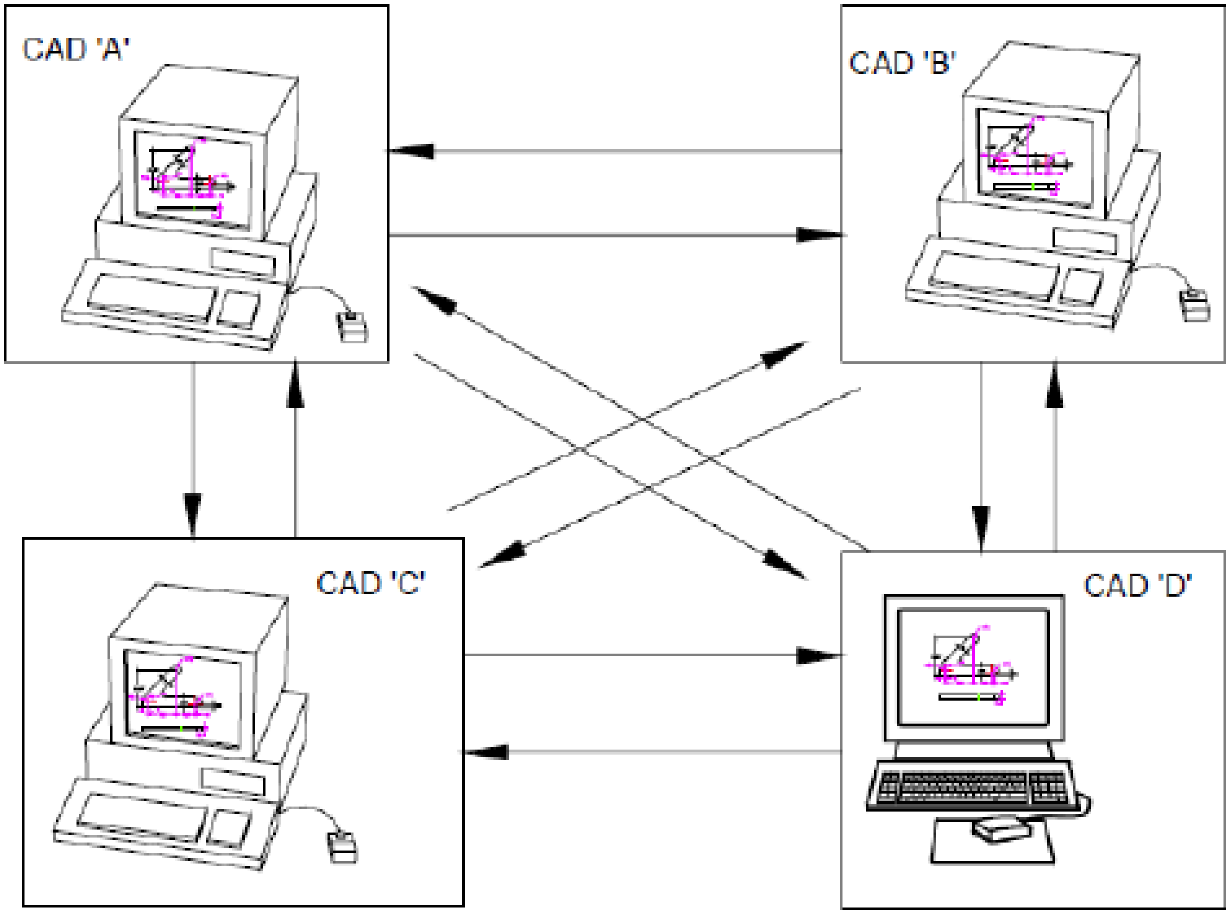
184: solid assembly

190: plane surface

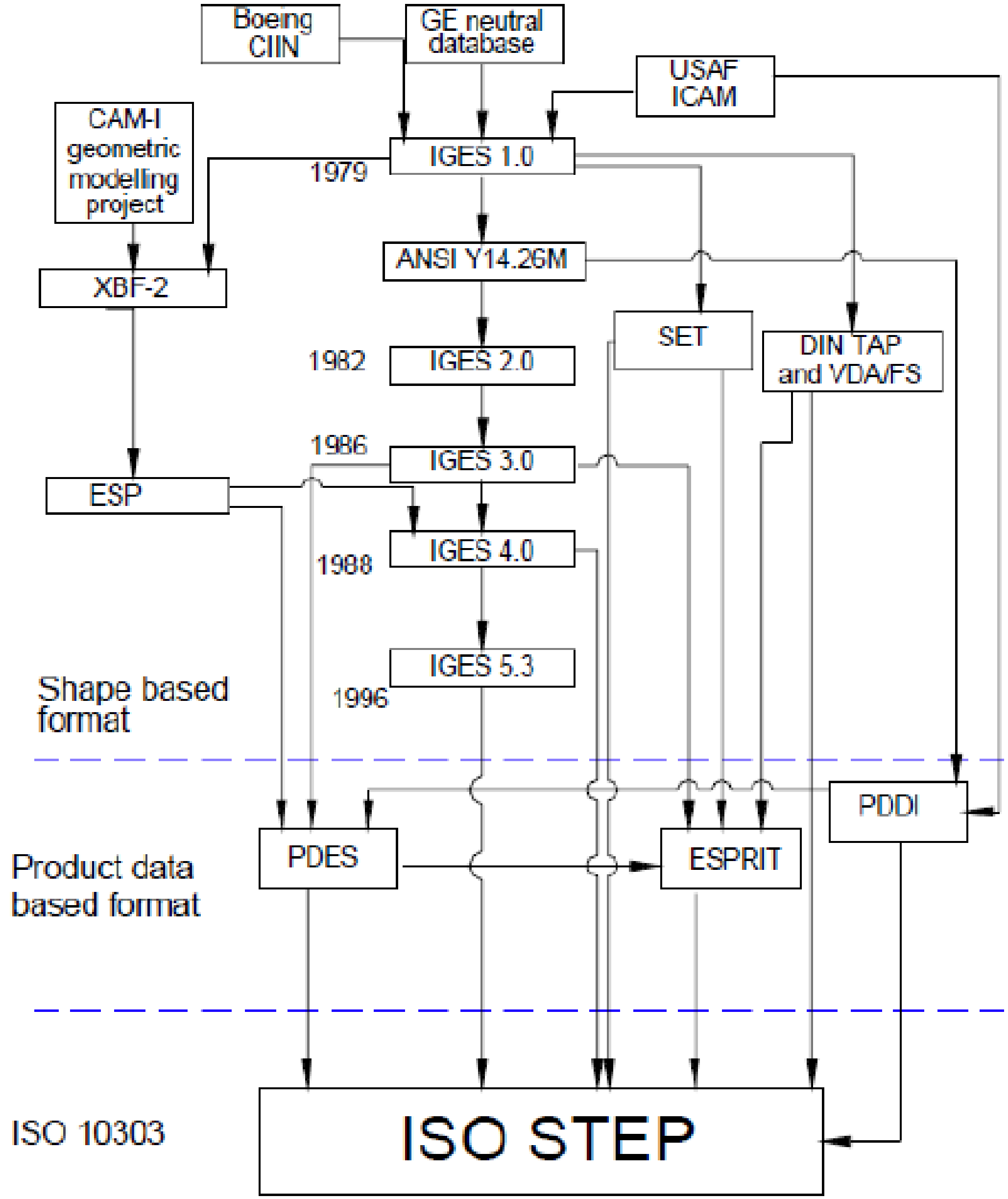
IGES FILE

Contains only two POINT (Type 116), two CIRCULAR ARC (Type 100), and two LINE (Type 110) entities.

Data exchange between various systems



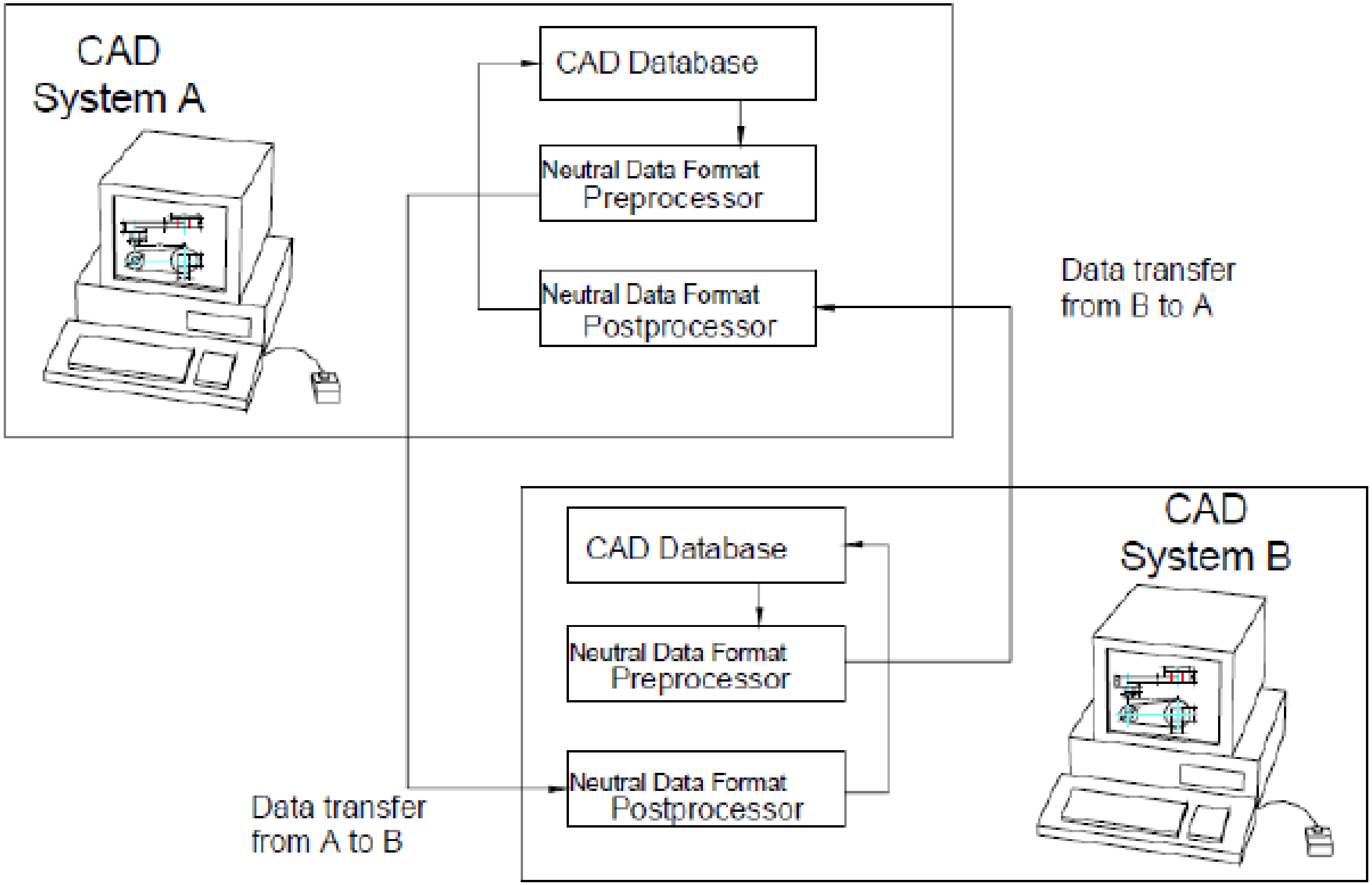
Developments in the drawing data exchange formats



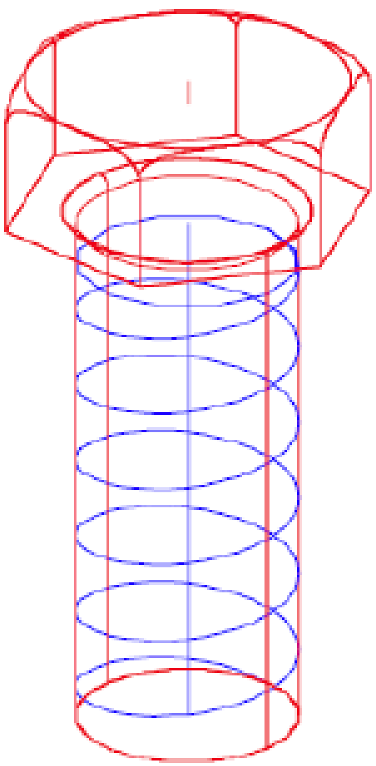
Initial Graphics Exchange Specification (IGES)

- IGES is the most comprehensive standard and is designed to transmit the entire product definition including that of manufacturing and any other associated information

Data interchnage method between two different CAD systems using neutral data format such as IGES or STEP



Component drawing and part of IGES file generated



```
PTC IGES file: inpart.igs                                                 S      1
1H,,1H;,1H1,10Hinpart.igs,                                                G      1
49HPro/ENGINEER by Parametric Technology Corporation,4H9741,32,38,7,38,   G      2
15,1H1,1.,1,4HINCH,32768,0.5,13H970430.104743,0.000396166,3.96182,        G      3
6Hmjiang,7HUnknown,10,0,13H970430.104743;                                 G      4
     124       1       1       1       0       0       0       001000000D      1
     124       0       0       1       0                       XFORM    1D      2
     100       2       1       1       0       0       1       001010000D      3
     100       0       0       1       0                         ARC    1D      4
     110       5       1       1       0       0       0       001010000D      9
     110       0       0       1       0                        LINE    1D     10
     126      23       1       1       0       0       0       001010000D     35
     126       0       0      23       0                    B_SPLINE    1D     36
     128     333       1       1       0       0       0       001010000D    109
     128       0       0       5       0                      SPLSRF    1D    110
     102     339       1       1       0       0       0       001010000D    111
     102       0       0       1       0                      CCURVE    1D    112
     142     392       1       1       0       0       0       001010500D    119
     142       0       0       1       0                      UV_BND    1D    120
     144     393       1       1       0       0       0       000000000D    121
     144       0       0       1       0                     TRM_SRF    1D    122
```

IGES

SUBSECTIONS OF IGES FILE

- **START SECTION:** Contains man readable prologue file.

- **GLOBAL SECTION:** Contains details about the product, organization, software, date etc.

- **DIRECTORY ENTRY SECTION:** Contains attribute information such as color , line type, etc.

* **PARAMETER DATA SECTION:** Contains data associated with entities.

* **TERMINATE SECTION:** This contains sub-totals of records present in each of the earlier sections.

Sub-sections in IGES

1. Flag section – Optional ASCII/binary/compressed ASCII
2. Start section(S) – Man readable prologue
3. Global section(G) – Details of product, person originating the product, name of the company originating it, date, details of system which generated it, drafting standard used etc.
4. Directory section(D) – Index for the file and attribute information like colour, line type etc.

Sub-sections in IGES

5. Parameter Entry section(P)

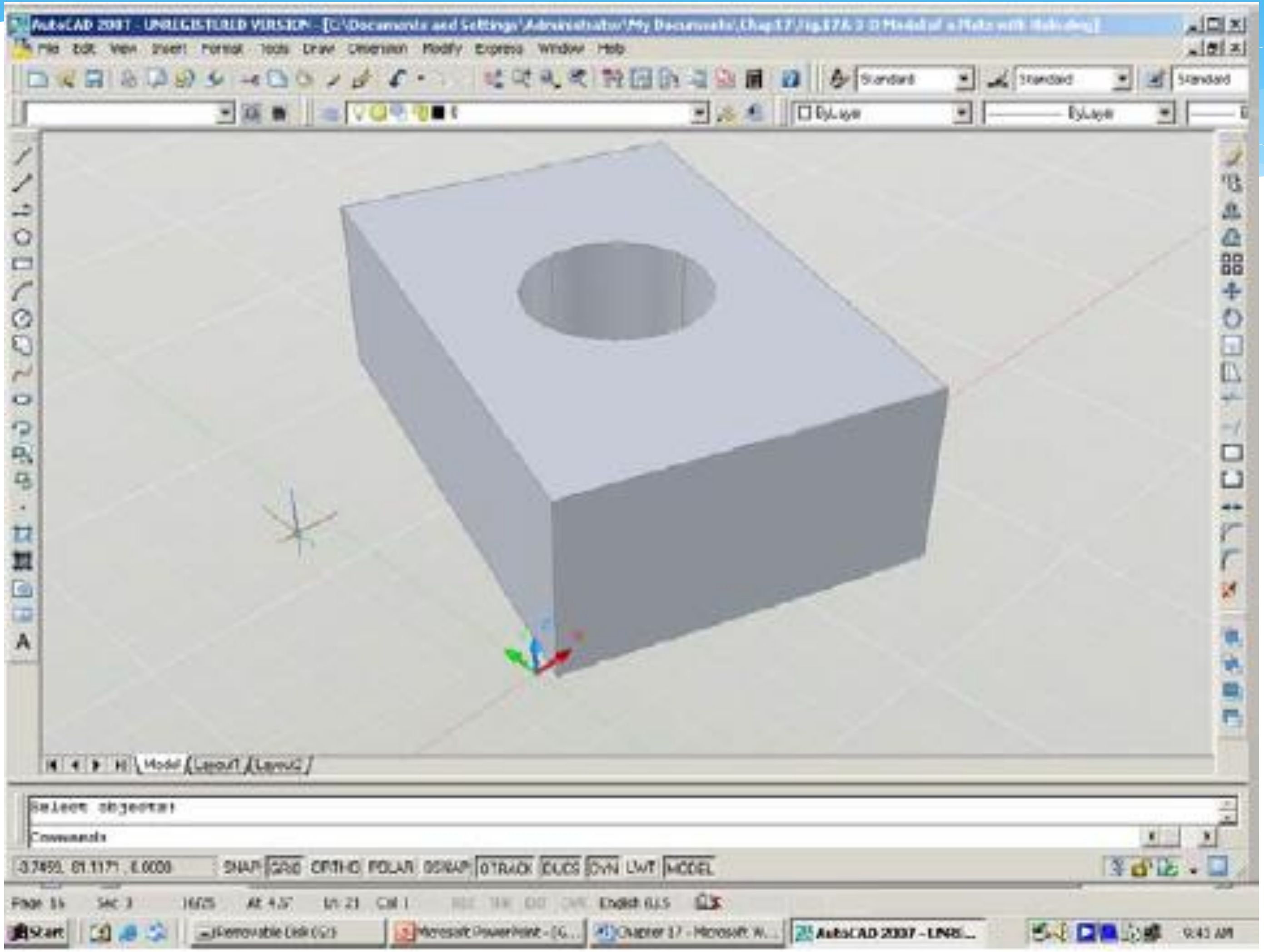
   Geometric entities– Point, line, arc, different surfaces, curves, solid primitives etc.

   Annotation entities – Angular dimension, diameter dimension, label, note, etc.

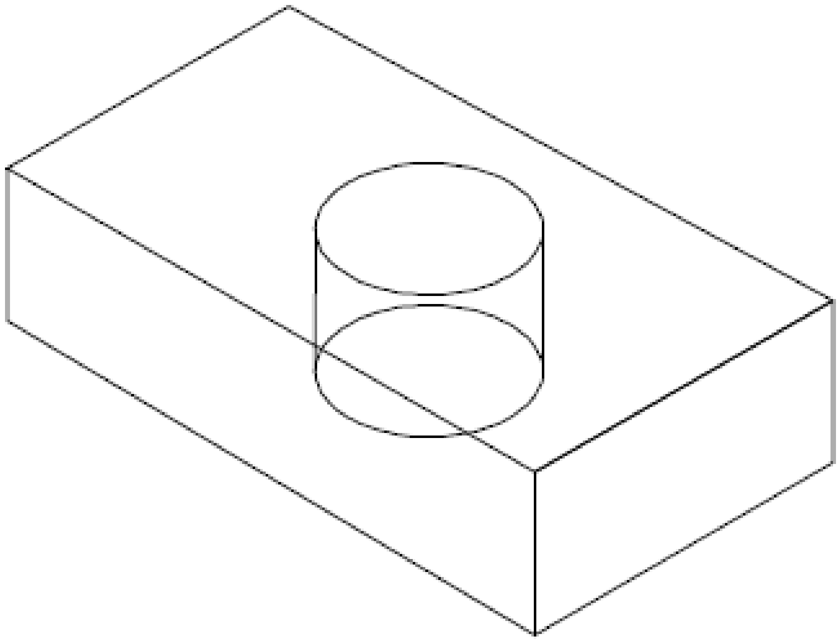
   Structure entities – Associativity definition, text font definition, color definition, units data etc.

6. Terminate section – Sub-totals of record in each of earlier sections

IGES part file case study



IGES OUTPUT OF WIRE FRAME MODEL



ENTITIES

POINTS-8

LINES-12

CIRCLES-2

IGES

Table 17.2 IGES-Wire frame edges

PTC IGES file: Wireframe_Edges.igs							S	1
1H,,1H;,4HSTEP,6Hi1.igs,							G	1
49HPro/ENGINEER by Parametric Technology Corporation,7H1999390,32,38,7,							G	2
38,15,4HSTEP,1.,2,2HMM,32768,0.5,13H000104.164438,0.00187075,18.708,							G	3
5Hstaff,7HUnknown,10,0,13H000104.164438;							G	4
110	1	1	1	0	0	0	000000000D	1
110	0	0	1	0	LINE	1D		2
110	2	1	1	0	0	0	000000000D	3
110	0	0	1	0	LINE	2D		4
110	3	1	1	0	0	0	000000000D	5
110	0	0	1	0	LINE	3D		6
110	4	1	1	0	0	0	000000000D	7
110	0	0	1	0	LINE	4D		8
124	5	1	1	0	0	0	001000000D	9
124	0	0	1	0	XFORM	1D		10
100	6	1	1	0	0	9	000000000D	11
100	0	0	1	0	ARC	1D		12
110	7	1	1	0	0	0	000000000D	13
110	0	0	1	0	LINE	5D		14
110	8	1	1	0	0	0	000000000D	15
110	0	0	1	0	LINE	6D		16
110	9	1	1	0	0	0	000000000D	17
110	0	0	1	0	LINE	7D		18
110	10	1	1	0	0	0	000000000D	19
110	0	0	1	0	LINE	8D		20
110	11	1	1	0	0	0	000000000D	21
110	0	0	1	0	LINE	9D		22
110	12	1	1	0	0	0	000000000D	23
110	0	0	1	0	LINE	10D		24
110	13	1	1	0	0	0	000000000D	25

IGES

```
110     0      0     1     0       LINE      11D                        26
110     14     1     1     0        0         0       000000000D        27
110     0      0     1     0       LINE      12D                        28
124     15     1     1     0        0         0       001000000D        29
124     0      0     1     0      XFORM      2D                         30
100     16     1     1     0        0         29      000000000D        31
100     0      0     1     0       ARC       2D                         32
110     17     1     1     0        0         0       000000000D        33
110     0      0     1     0       LINE      13D                        34
110     18     1     1     0        0         0       000000000D        35
110     0      0     1     0       LINE      14D                        36

110,0D0,0D0,0D0,0D0,-1D1,0D0;                                   1P      1
110,0D0,-1D1,0D0,1.5D1,-1D1,0D0;                                3P      2
110,1.5D1,-1D1,0D0,1.5D1,0D0,0D0;                               5P      3
110,1.5D1,0D0,0D0,0D0,0D0,0D0;                                  7P      4
124,-1D0,0D0,0D0,7.5D0,0D0,1D0,0D0,-5D0,0D0,0D0,-1D0,0D0;       9P      5
100,0D0,0D0,0D0,2.5D0,0D0,2.5D0,0D0;                            11P     6
110,0D0,0D0,0D0,0D0,0D0,5D0;                                    13P     7
110,1.5D1,0D0,0D0,1.5D1,0D0,5D0;                                15P     8
110,1.5D1,-1D1,0D0,1.5D1,-1D1,5D0;                              17P     9
110,0D0,-1D1,0D0,0D0,-1D1,5D0;                                  19P     10
110,0D0,0D0,5D0,0D0,-1D1,5D0;                                   21P     11
110,1.5D1,0D0,5D0,0D0,0D0,5D0;                                  23P     12
110,1.5D1,-1D1,5D0,1.5D1,0D0,5D0;                               25P     13
110,0D0,-1D1,5D0,1.5D1,-1D1,5D0;                                27P     14
124,-1D0,0D0,0D0,7.5D0,0D0,1D0,0D0,-5D0,0D0,0D0,-1D0,5D0;       29P     15
100,0D0,0D0,0D0,2.5D0,0D0,2.5D0,0D0;                            31P     16
110,5D0,-5D0,5D0,5D0,-5D0,0D0;                                  33P     17
110,1D1,-5D0,5D0,1D1,-5D0,0D0;                                  35P     18
S       1G     4D     36P     18                                T       1
```

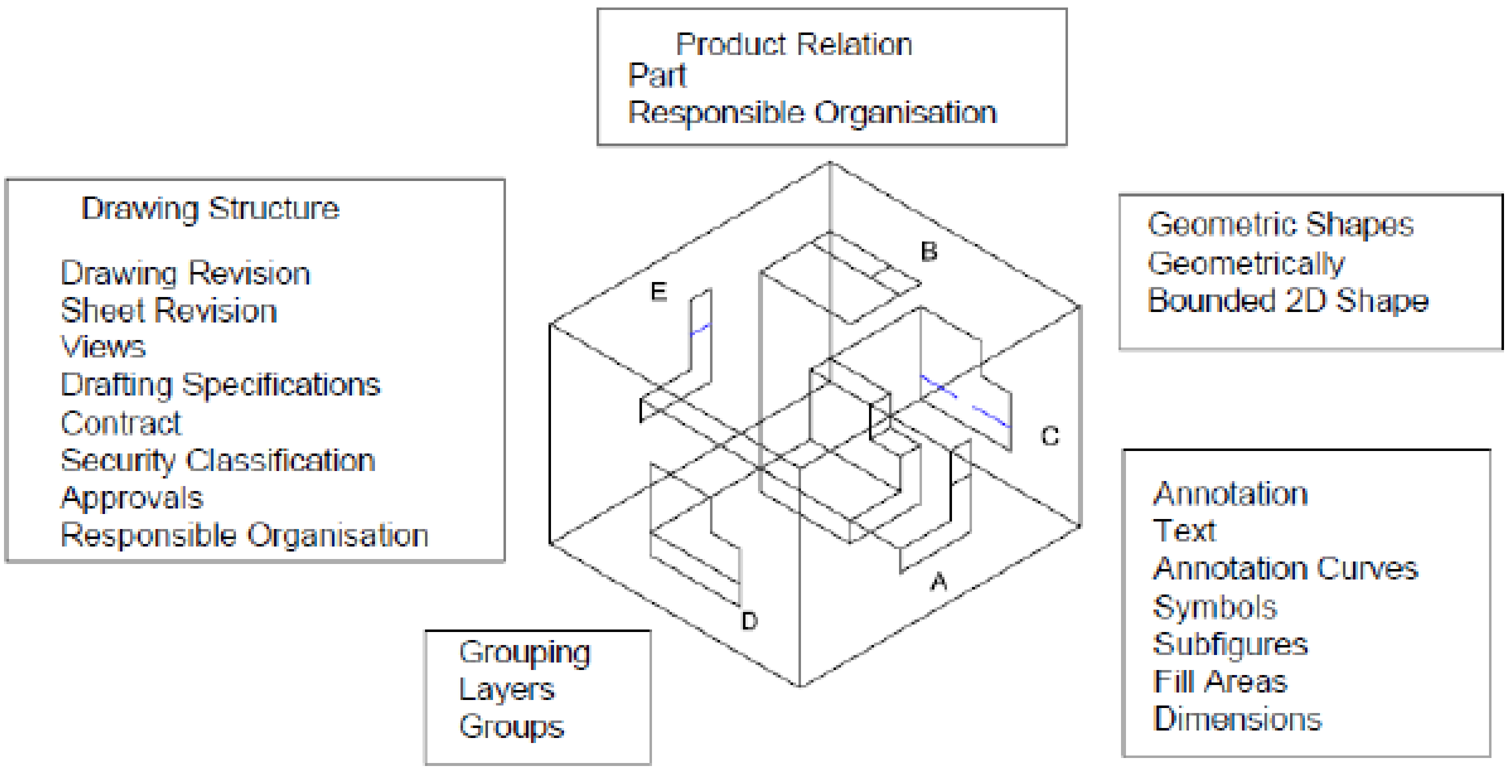
Standard for the Exchange of Product model Data (STEP)

- The broad scope of STEP is as follows:

– The standard method of representing the information necessary to completely define a product throughout its entire life, i.e., from the product conception to the end of useful life.

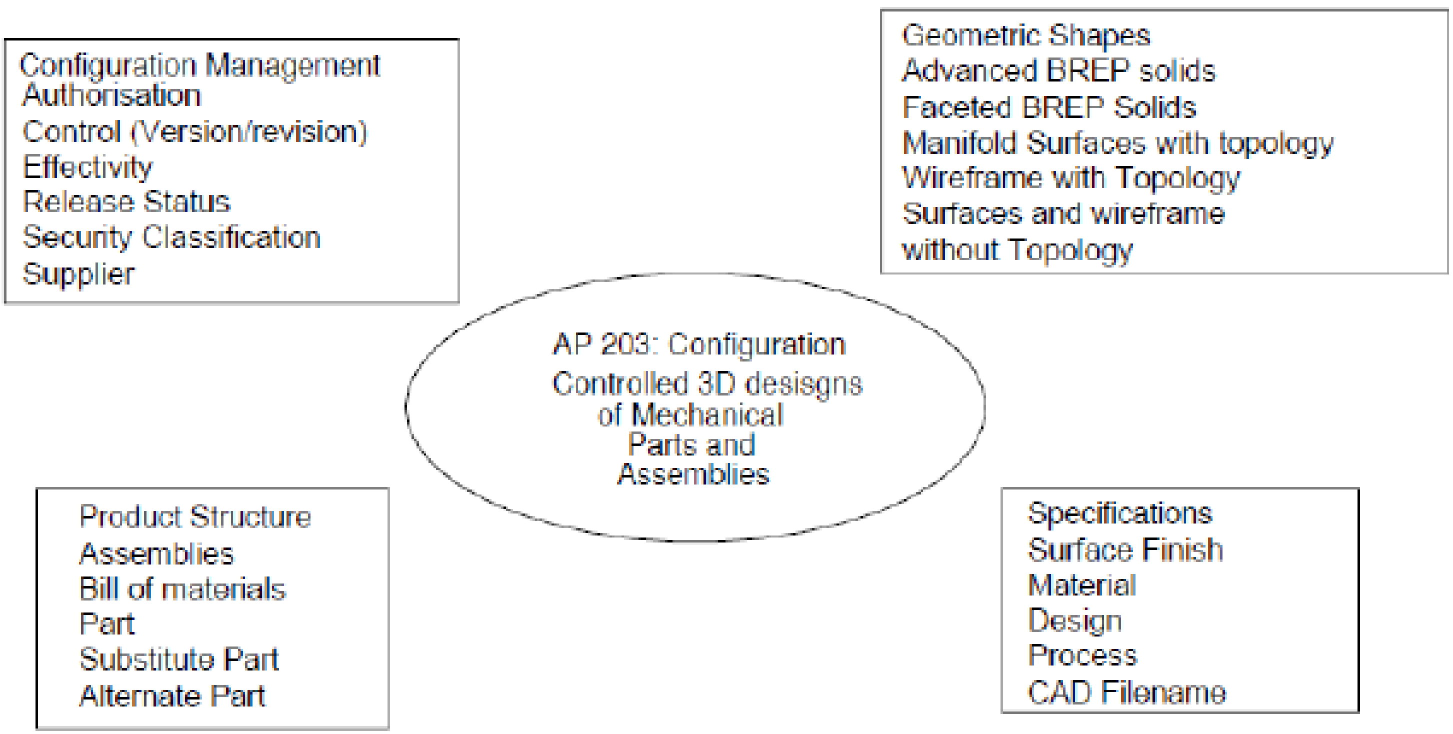
– Standard methods for exchanging the data electronically between two different systems.

STEP Application Protocol AP 203 Explicit Draughting



AP 201: Explicit Draughting

STEP Application Protocol AP 207 Configuration Controlled Design



AP 203: Configuration Controlled 3D designs

Example for STEP file generation Sheet Metal Die Planning and Design

Shape Definition Representation
- Advanced BREP Solids
- Elementary BREP Solids
- Facetted BREP Solids
- Manifold Surfaces with Topology
- Surfaces and Wireframe without Topology
- Constructive Solid Geometry
- Part Shape Relationship to Die Shape
- Physical Model Identification
- Tolerances

AP 207: Sheet Metal Die Planning and Design

Items
- Parts, Dies or Materials
- Versions
- Classifications
- Standard or designed

Work
- Internal or External
- Start or Change

Part Process Plans
- Template
- Operations
- Constraints

AP 203: Configuration Controlled 3D designs

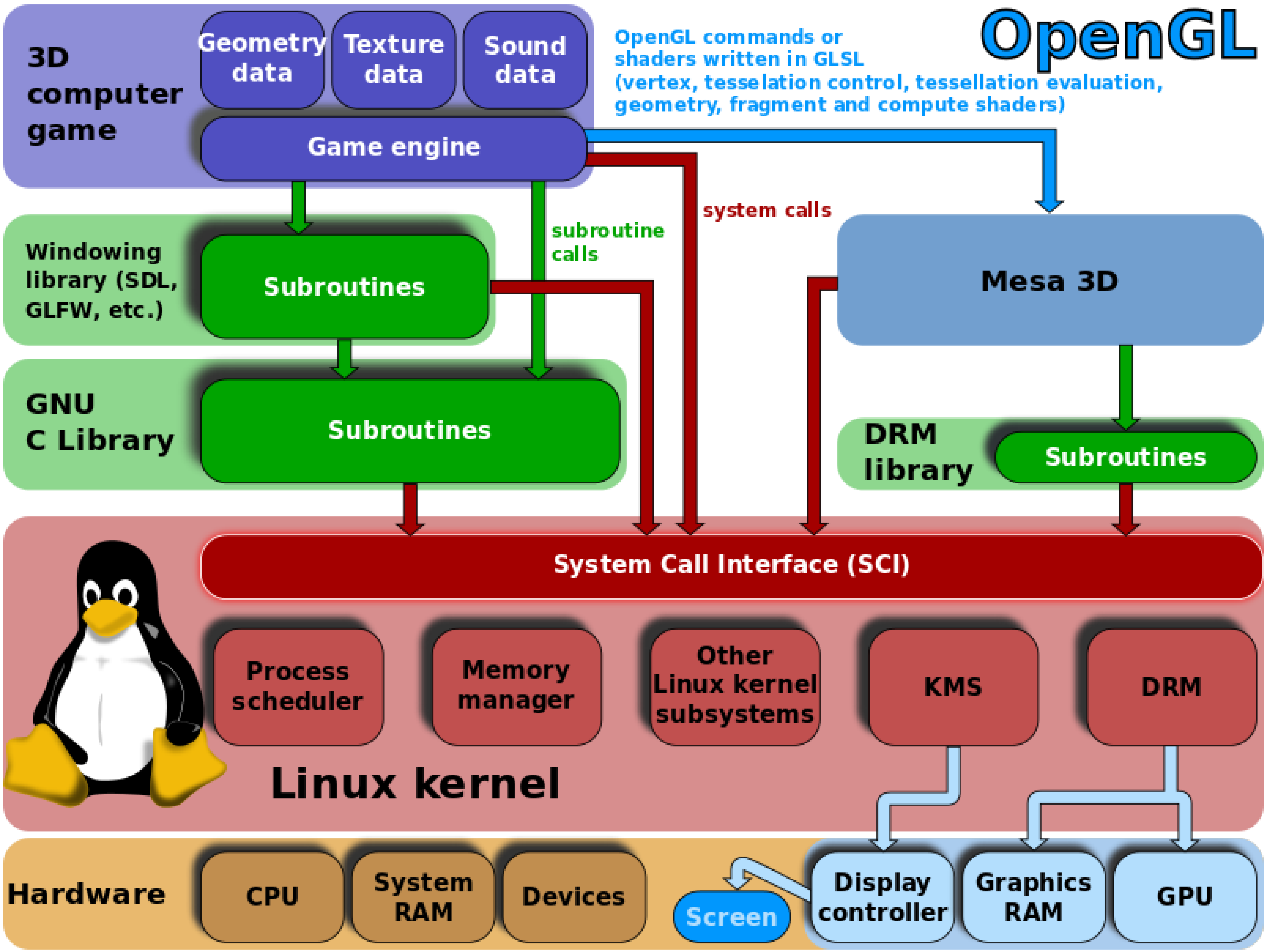
Open GL

Why do we need Data Exchange?

- Why do we need Data Exchange?
- Design projects require data to be shared between suppliers
- Different companies often used different CAD systems
- All CAD systems have their own database formats
- They are mostly proprietary and often confidential
- Data is stored in different ways e.g. 1.0,2.0,3.0 or X1.0,Y2.0,Z3.0, etc.
- Data conversion between systems becomes necessary

Introduction

- Open Graphics Library (OpenGL) is a cross-language, cross-platform application programming interface (API) for rendering2D and 3D vector graphics. The API is typically used to interact with a graphics processing unit (GPU), to achieve hardware-accelerated rendering.
- Silicon Graphics Inc., (SGI) started developing OpenGL in 1991 and released it in January 1992; applications use it extensively in the fields of computer-aided design (CAD), virtual reality, scientific visualization, information visualization, flight simulation, and video games. OpenGL is managed by the non-profit technology consortium Khronos Group.



OpenGL
3D computer game
Geometry data
Texture data
Sound data
Game engine
OpenGL commands or shaders written in GLSL (vertex, tesselation control, tessellation evaluation, geometry, fragment and compute shaders)
system calls
subroutine calls
Windowing library (SDL, GLFW, etc.)
Subroutines
Mesa 3D
GNU C Library
Subroutines
DRM library
Subroutines
System Call Interface (SCI)
Process scheduler
Memory manager
Other Linux kernel subsystems
KMS
DRM
Linux kernel
Hardware
CPU
System RAM
Devices
Screen
Display controller
Graphics RAM
GPU