

## **Unit VI**

# **Introduction to Animation and Gaming**

**Segment:** Introduction, Segment table, Segment creation, closing, deleting and renaming, Visibility.

**Animation:** Introduction, Conventional and computer based animation, Design of animation sequences, Animation languages, Key-frame, Morphing, Motion specification.

**Gaming:** Introduction, Gaming platform (NVIDIA, i8060), Advances in Gaming.

### **Course Outcomes:**

**CO6: Create** effective programs using concepts of curves, fractals, animation and gaming.

### **Text Books:**

1. S. Harrington, "Computer Graphics"||, 2nd Edition, McGraw-Hill Publications, 1987, ISBN 0 – 07 – 100472 – 6.
2. Donald D. Hearn and Baker, "Computer Graphics with OpenGL", 4th Edition, ISBN-13: 9780136053583.
3. D. Rogers, "Procedural Elements for Computer Graphics", 2nd Edition, Tata McGraw-Hill Publication, 2001, ISBN 0 – 07 – 047371 – 4.

### **Reference Books:**

1. J. Foley, V. Dam, S. Feiner, J. Hughes, "Computer Graphics Principles and Practice"||, 2nd Edition, Pearson Education, 2003, ISBN 81 – 7808 – 038 – 9.
2. D. Rogers, J. Adams, "Mathematical Elements for Computer Graphics"||, 2nd Edition, Tata McGraw Hill Publication, 2002, ISBN 0 – 07 – 048677 – 8.

# Segment & Animation

[Reference - TBI]

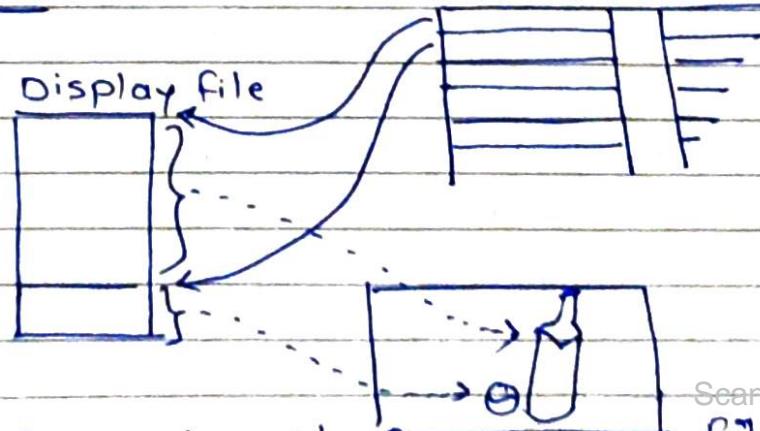
## Introduction -

- image on the display screen is often composed of several pictures or items of information.
- image may contain several views of an object & related information.
- it may also contain close up view of a particular component.  
e.g - if we want to display an internal plan of a living room, it contains various objects such as sofa-set, T.V., show-case, teapot, show-pieces, wall hangings etc.
- each object has a set of attributes like size, color, & its position in the room.
- we want to see all these objects simultaneously or single object at a time.
- To view the entire image or part of the image with various attributes we need to organize the image information in a particular manner.
- image information is stored in the display file.
- existing structure of the display file does not satisfy our requirements of viewing the image. the display structure is modified to reflect the subpicture structure. to achieve this display file is divided into segments.

- each segment corresponds to a component or an object of the overall display & is associated with a set of attributes.
- Along with the attributes the segment is also associated with the image transformation parameters such as scaling along X & Y direction, rotation & shearing.
- The presence of segment allows
  - subdivision of pict.
  - visualization of a particular part of picture.
  - Scaling, rotation & translation of particular part of picture.

### Segment Table :-

- To access a particular segment & the information associated with it we must have a unique name assigned to each segment.
- along with the NAME we must <sup>have</sup> its display file position & its attribute information.
- The structure used to organize all this information related to segments is called segment Table.



- it indicates the portions of display file used to construct the picture.
- The segment table is formed by using arrays.
- 1st array holds the display file starting location for that segment, the second array holds the segment size information, while the 3rd indicates the visibility & so on

Segment no	segment start	segment size	scale x	scale y	color
0					
1					
2					
3					
4					

segment table

- each row in segment table represents info of one segment including its name, position, size, attribute & the image info parameters.
- if we want to make segment 4 visible then this is achieved by setting the corresponding entry in the array to 'ON'.

### Functions for Segmenting the Display file:-

- we are familiar with the common methods of creating & modifying Sequential disk files.
- to open a file before we add data to it, & we close the file when we have added the last data item

PAGE NO. [ ] / [ ]  
DATE [ ] / [ ] / [ ]

- it indicates the portions of display file used to construct the picture.
- The segment table is formed by using arrays.
- 1st array holds the display file starting location for that segment, the second array holds the segment size information, while the 3rd indicates the visibility & so on.

Segment no	Segment start	Segment size	scale x	scale y	color	visibility
0						
1						
2						
3						
4						
:	:	:	:	:	:	:

segment table

- each row in segment table represents info of one segment including its name, position, size, attributes & the image info parameters.
- if we want to make segment 4 visible then this is achieved by setting the corresponding entry in the array to 'ON'.

### Functions for Segmenting the Display file:-

- we are familiar with the common methods of creating & modifying Sequential disk files.
- to open a file before we add data to it, & we close the file when we have added the last data item

& the file is complete. To change the contents of a file, we open it again, add the new data to replace the old. & close new file.

To get rid of file, delete it.

- The very same 'open's are used for manipulating display file segments.
- To create new segment, we create it & then call graphic primitives to add to the segment the lines & text to be displayed then we close the segment.
- To modify the contents we rename current segment with the modified segment & to remove segment from the display file, delete it.

Four basic functions -

- Create Segment
- Close Segment
- Delete Segment
- Rename for Segment.

### Segment Creation -

- in this process, we have to check whether some other segment is still open.
- it is not allowed to open two segments at same time because it is then difficult to assign the drawing instructions to particular segment. Segment must be created or opened when no other segment is currently open.

- it must give the segment a name so that we can identify it. while doing this it is important to check whether a given segment name is valid or not & whether there already exists a segment name. if valid so we have to assign other valid name to the Segment.
- Once valid segment name is assigned, we have to initialize the items in the Segment table under our segment name to indicate that this is fresh new segment.
- The 1st instruction of this segment will be located at next free storage area in the display file. we have not entered any instructions into the segment yet, its size is initialized with value zero.
- The attributes of the segment are initialized as default attribute values.

Algo:-

1. Check whether any segment is open, if so display error msg "segment is still open" & go to step 9.
2. Read the name of new segment.
3. Check whether the new segment name is valid, if not display error message : Not a valid segment name & go to step 9.
4. Check whether the new segment name is already existing in the same name list. if so display error message.

it must give the segment a name so that we can identify it while doing this it is important to check whether a given segment name is valid or not & whether there already exists a segment name if ~~valid~~ so ~~not~~ we have to assign other valid name to the Segment.

- Once valid segment name is assigned, we have to initialize the items in the segment table under our segment name to indicate that this is fresh new segment.
- The 1st instruction of this segment will be located at next free storage area in the display file. We have not entered any instructions into the segment yet, its size is initialized with value zero.
- The attributes of the segment are initialized as default attribute values.

Algo:-

1. Check whether any segment is open, if so display error msg "segment is still open" & go to step g.
2. Read the name of new segment.
3. Check whether the new name is valid, if not display error message : Not a valid segment name & go to step g.
4. Check whether the new segment name is already existing in the name list. if so <sup>display</sup> ~~so~~ error message.

- it must give the segment a name so that we can identify it. while doing this it is important to check whether a given segment name is valid or not & whether there already exists a segment name. if valid so we have to assign other valid name to the Segment.
- Once valid segment name is assigned, we have to initialize the items in the segment table under our segment name to indicate that this is fresh new segment.
- The 1st instruction of this segment will be located at next free storage area in the display file. We have not entered any instructions into the segment yet, its size is initialized with value zero.
- The attributes of the segment are initialized as default attribute values.

Algo:-

1. Check whether any segment is open, if so display error msg "segment is still open" & go to step 9.
2. Read the name of new segment.
3. Check whether the new segment name is valid, if not display error message : Not a valid segment name & go to step 9.
4. Check whether the new segment name is already existing in the same namelist. if so display error message.

" Segment name already exists &  
go to Step 9.

5. Initialize start of the segment at  
the next free storage area in the  
display file.
6. Initialize size of this segment equal  
to zero.
7. Initialize all attributes of segment  
to their default values.
8. Indicate that the new segment  
is now open.
9. stop.

### Closing a segment :-

- Once a segment is open we can  
enter the display file instructions  
in it. The entered commands are  
then associated with the open  
segment.
- After completion of entering all  
display file instructions, the segment  
must be closed.
- To close a segment it is necessary to  
change the name of the currently  
open segment. It can be achieved  
by changing the name of currently  
open segment as 0. Now the segment  
with name 0 is open i.e unnamed  
segment is open.
- If there are two unnamed segments  
in the display file one has to be  
deleted.

## Algo:- close Segment

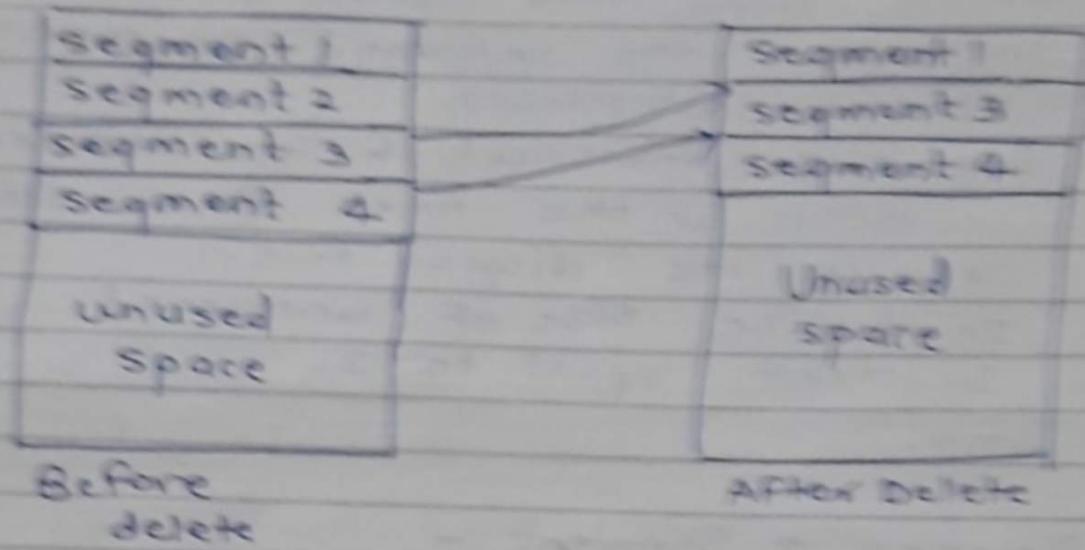
1. check whether any segment is open if no, display error message: "No segment open" & go to step 6.
2. Change the name of currently open segment, i.e 0, unnamed segment.
3. Delete any unnamed Segment instructions which may have been saved. i.e initialize the unnamed segment with no instructions.
4. Initialize the start of the unnamed segment at the next free storage area in the display file.
5. Initialize size of unnamed segment equal to 0.
6. stop.

## Deleting a Segment :-

- when we want to delete a particular segment from the display file then we must recover the storage space occupied by its instructions & make this space free for some other segment.
- To do this we must not destroy & reform the entire display file, but we must delete just one segment while preserving the rest of the display file.
- The method to achieve this depends upon the data structure used to represent the display file.
- we have used arrays to store the display file info.
- use of arrays makes the recovery of storage space occupied by the

Segment very easy & straight forward. using arrays is not as efficient as some other storage techniques such as linked list.

- In case of arrays the gap left by deleted block is filled by shifting up all segments which are following the deleted segment.



display file contents before & after deleting segment 2

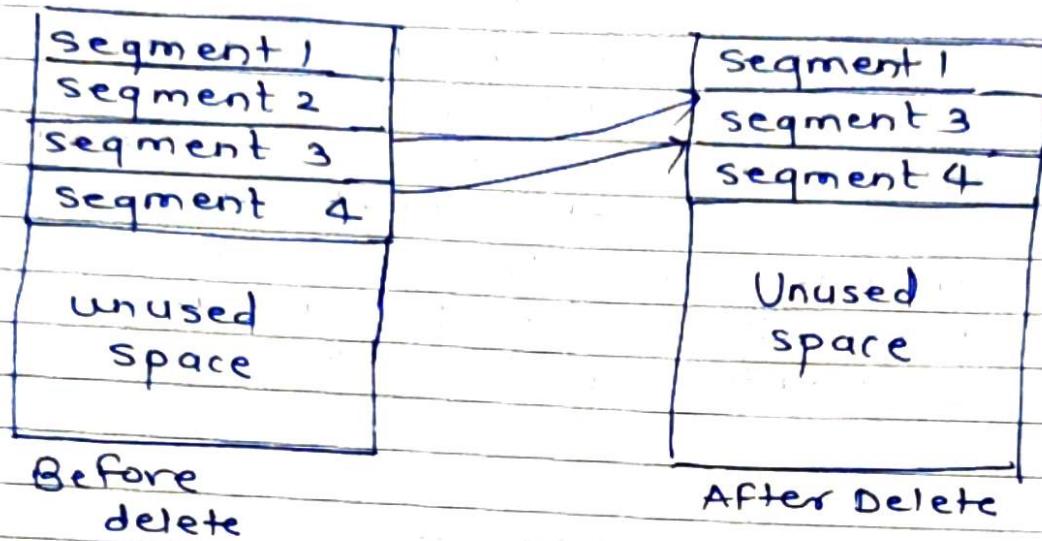
Algo :-

1. Read the name of segment which is to be deleted.
2. Check whether that segment name is valid, if not, display error message "segment not valid" go to step 3.
3. Check whether the segment is open if yes, display error message, can't delete the open segment so go to step 8.
4. Check whether the size of segment is greater than 0, if no no processing is required, as segment contains no instructions. go to step 8.

PAGES: \_\_\_\_\_ DATE: \_\_\_\_\_

Segment very easy & straight forward. Using arrays is not as efficient as some other storage techniques such as link list.

- In case of arrays the gap left by deleted block is filled by shifting up all segments which are following the deleted segment.



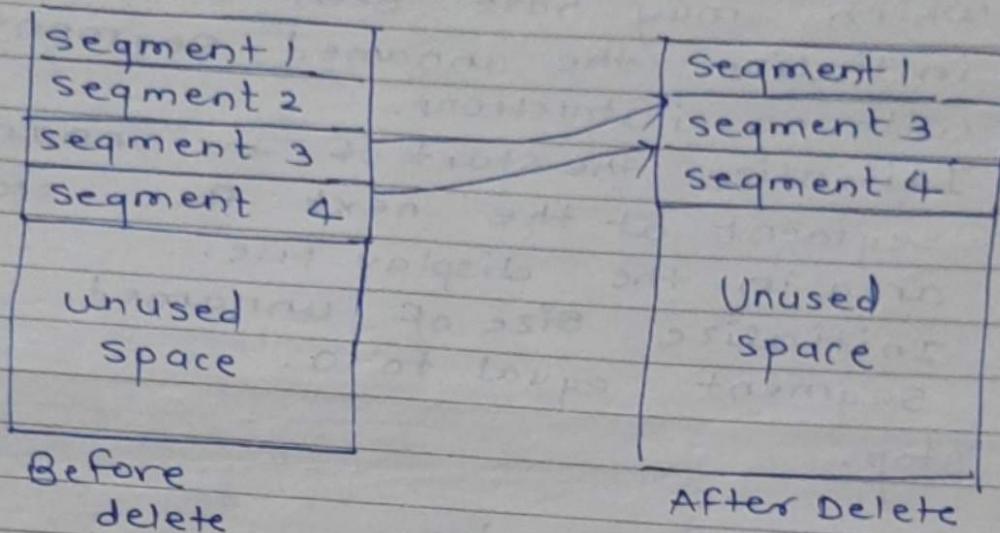
display file contents before & after  
deleting segment 2

Algo :-

1. Read the name of segment which is to be deleted.
2. Check whether that segment name is valid, if not, display error message "segment not valid, go to step 8."
3. Check whether the segment is open if yes, display error message, can't delete the open segment & go to step 8.
4. Check whether the size of segment is greater than 0, if no no processing is required, as segment contains no instructions. go to step 8.

Segment very easy & straight forward. Using arrays is not as efficient as some other storage techniques such as link list.

- In case of arrays the gap left by deleted block is filled by shifting up all segments which are following the deleted segment.



display file contents before & after deleting segment 2

Algo:-

1. Read the name of segment which is to be deleted.
2. Check whether that segment name is valid, if not, display error message
3. If segment not valid, go to step 8.
4. Check whether the segment is open
  - if yes, display error message, can't delete the open segment & go to step 8.
  - if no, no processing is required, as segment contains no instructions. go to step 8.

5. Shift the display file elements which follows the segment which is to be deleted by its size.
6. Recover deleted space by resetting index of next free instruction.
7. Adjust starting positions of the shifted segments by subtracting the size of the deleted segment from it.
8. stop.

### Renaming a Segment -

- we have a segment in display file for animated character. Then to display a new image in the sequence we have to delete the current segment & recreate it with the altered character.
- The problem in this process is that during time after 1st image is deleted & time before the second image is completely entered, only a partially completed character is displayed on the screen.
- we can avoid this problem by keeping the next image ready in the display file before deleting the current segment.
- This means that both segments, the segment which is to be deleted & the segment which is to be replaced with must exist in the display file at the same time.
- This can be achieved by creating a new invisible image under some temporary segment name.

- when the current segment is deleted we can make the new image visible & rename it with the name of deleted segment.
- These steps can be repeated to achieve the animation.
- The idea of storing two images, one to show & other to create or alter, is called double buffering.

Algo:-

1. check whether both old & new segment names are valid, if not display error message "Not valid segment names" & go to step 6.
2. Check whether any of the two segments are open  
if open display error message "Segments still open" & go to step 6.
3. Check whether new name we are going to give to old segment is not already existing in the display file. if yes display error message "Segment already exists" & go to step 6.
4. Copy the old segment table entry into new position.
5. Delete old segment
6. Stop.

### Visibility -

- every segment is provided with a visibility attribute. The visibility of the segment is stored in an array.

which is part of the segment table.

By scanning the array one can determine whether to display the segment or not. The segments visibility can also be changed according to the user's view.

— so that the user can decide to show or not to show the segment.

[Reference — RBI]

### Animation :-

Introduction — Animation covers any change of appearance of any visual effect that is time based.

- it has many faces.
- It includes change of position, transparency, time varying changes in shape & even changes of the rendering techniques.
- The animation classified into 2 types —
  - a) Frame animation
  - b) Sprite animation

- frame animation is an 'internal' animation method.
- It is typically pre-compiled & non-interactive.
- It is rectangular frame & non transparent.
- The frame animation with transparency info is called as 'cell' animation.
- In traditional animation, a cell is sheet of transparent acetate on which a single character or object is drawn.

e.g - EGI is frame animation engine,  
Compiler.

### Sprite Animation

- is 2D graphic object that moves across the display.
- it can have transparent areas when a mask or transparent color is used sprites are not restricted to a rectangular shapes.
- it is well interactive & is referred as external animation.
- All sprite animation libraries & tool kit allow some form of internal Animation.
- e.g - Anisprite is a sprite animation.

### Def" of Animation

- it is defined as time base manipulation of a target element . The animation defines a mapping of time to values for the target attribute.
- Animation is achieved by a series of geometric transformations , Scaling , rotations , translations or any other mathematical technique to produce a sequence of scenes .
- it usually signifies the artificially drawn picture sequence . &
- It does not specify a begin & a simple duration that can be repeated .
- each animation defines an

'animation function' that produces a value for the target attribute, or for any no. of times within the simple duration.

- The user can specify the duration & number of times an animation function should repeat.
- The simple duration combined with any repeating behaviour defines the active duration.

### Design of Animation Sequences :-

#### i) squash & stretch :-

— it is used to indicate the physical properties that are expected to get changed in the process of animation.

e.g. rubber ball when dropped on the floor might have to be shown as getting distorted in its shape due to its elasticity.

— A bouncing rubber ball might be shown as elongating as it approaches the floor, flattening out when it hits & then elongating again as it rises.

— By contrast, a metal sphere hitting the floor might distort very little but might wobble after the impact, exhibiting very small high-frequency distortions.

— These distortions have to be calculated depending on the motion & the environments in

in which the animations are planned.

- They may be derived from complex differential equations to form a simple simultaneous equation to model these distortions.

### 2) Slow-in & slow-out :-

- it helps smooth interpolations.
- Sudden, jerky motions are extremely distracting. This is particularly evident in interpolating the camera position.
- An audience viewing an animation identifies with the camera view, so sudden changes in camera position may make the audience feel motion sickness. Camera changes should be as smooth as possible.

### 3) Maintaining 3D Affects :-

- It carries over naturally from the 2D character animation world to 3D animations is to stage the action properly.
- It includes choosing a view that conveys the most info about the events taking place in animation & isolating events so that only one thing at a time occupies the viewer's attention.
- Staging (setting the scene) - clear & easy to understand presentation

of the idea of animation. Action should be presented in manner easy to understand.

Anticipation - Action of an animated character usually goes in 3 phases - preparation phase, movement & finish phase.

- Anticipation concerns the preparation A jumper usually sags his knees before jumping what is connected connected with physics.
- Secondary Action - is any type of action which results from the main action. This could be a character'summy wobbling after they have jumped from a great height or facial expression of agony Tom has been hit on the toe by Jerry.
- Timing - A tempo of moving object gives a meaning of the object & prompts why this object is moving. e.g. quick wink denotes that the object is worried or stimulated but slow.

### Animation Languages :-

- There are many different languages for describing Comp. animation & new ones are constantly being developed.
- Categorized as
  - linear list notations
  - General purpose lang. with embedded animation directives
  - Graphical lang.

- 1] linear list Notations:- each event in the animation is described by a starting & ending frame no. & an action that is to take place.
- The action is specified by a statement with relative parameters.  
e.g 30,45,C ROTATE "ARM", 1,60
  - This statement say that " bet" frames 30 & 45, rotate the object called table C.
  - statement describe individual actions & have frame values associated with them, their order is for the most part, irrelevant.
  - if two actions are applied to the same object at the same time, their order is important.
  - different order may give different results.

## 2] General-purpose language-

- such as C, C++, pascal or lisp can be used to design & control the animation Sequences.
- The values of variables in these lang can be used as parameters to whatever routines actually generate animations.
- These lang have great potential.
- ASAS is an e.g of this lang. it's built on top of LISP. its primitives includes vectors, colors, polygons, solids, groups, points of view, subworlds & lights.

- it includes a wide range of geometric transformations that operate on objects.
- They take an object as an argument & return a value that is a transformed copy of the object.
- These Transformations include up, down, left & right, zoom-in, zoom out, forward & backward.
- Advantage of ASAS over linear list notations is the ability to generate procedural objects & animations within the lang.

### 3] Graphical Lang:-

- There are several specialized animation lang developed called graphical languages.
- These languages provide various animation fun's which make it easy to design & control the animation.
- animation fun's include a graphics editor, a key frame generator an in bet' generator & standard graphics routines.
- Specialized Animation Language allows us to design & modify object shapes, position of the object & light source.

## Gaming :- [Reference - RBI]

### Introduction :-

- A Gaming platform is also called as videogame platform / videogame system.
- it refers to the specific combination of electronic or comp. H/w which in conjunction with SW, allows a videogame to operate.
- we should select the gaming platform carefully which will be based on mainly area of interest.
- Games could be in domain of entertainment, sport, movies, music, action games, etc.
- All videogames are tested on PC only.
- video game consoles are also used for playing videogames.
- There are many companies in market which deals with gaming platforms.
- These companies are continuously updating predecessors of their products every few year.
- PC's offers the unique ability to upgrade H/w, with some technical experience required. but without having to replace the entire comp.
- either by adding m/m, storage space, graphics cards, or even a new motherboard. we can extend the life of our PC without spending high prices of completely new comp.

### Advantages of PC

- multiple uses other than gaming
- Great multiplayers capability via Internet.
- upgradeable
- Great for strategy, real time strategy & simulation games.

### Disadvantages of PC

- expensive
- Smaller library of games.
- Sports games play very average on this platform.

### Advantages of Console -

- Relatively inexpensive
- Huge variety of games.
- Sports games thrive on consoles.

### Disadvantages of console

- Not upgradeable
- limited use other than playing video games.
- limited multi players capability.

### WORKSTATION :-

- is high end microcomp. designed for technical or scientific appl'ns.
- intended primarily to be used by one person at a time, they are commonly connected to LAN & run multi user oper' sys.
- The term workstation used to refer to a mainframe comp. terminal or PC connected to nw.

## CUDA :-

- Computer Unified Device Archi., is a parallel computing Archi. developed by NVIDIA.
- CUDA is computing engine in NVIDIA graphics processing units.
- programmers write for CUDA to code algo for execution on GPU.

### Advantages of CUDA:-

- Scattered reads - Code can read from arbitrary addresses in m/m.
- Shared m/m - CUDA exposes a fast shared m/m region (16 KB in size) that can be shared amongst threads. This can be used as a user managed cache, enabling higher BW than is possible using texture lookups.
- faster downloads & readbacks to & from GPU.
- full support for integer & bitwise op's including integer texture lookups.

### NVIDIA workstations -

Graphic processing units

- are based on GPU's with CUDA support.
- NVIDIA has mainly Tesla workstation maximum workstations based on GPU, & CUDA

#### 1] Tesla workstation -

- is desktop comp. that is backed by NVIDIA & built by Dell, Lenovo & other companies.

- It is meant to be a demonstration of the capabilities of NVIDIA's Tesla GPGPU brand.
- it utilizes NVIDIA's CUDA parallel computing Archi & is powered by up to 12688 parallel processing cores per GPGPU which allow it to achieve speeds upto 250 times faster than standard PC's according to NVIDIA.
- NVIDIA Tesla GPU Accelerators turn standard PC's & workstations into personal supercomp.
- Tesla workstation products are supported under windows vista & windows 7 - 32 bit & 64 bit.
- Linux, 32 bit & 64 bit.

## 2] Quadro Workstation -

- Quadro mobile graphics processing units (GPU) are trusted sol<sup>n</sup> for professional 3D appl<sup>n</sup>.
- features -
  - 1) Quadro 5000 comes with more than twice as much memory as the GeForce GTX 465.
  - 2) Quadro sol<sup>n</sup> combine most advanced display technologies & ecosystem interfaces to provides the ultimate visual workspace for maximum productivity.
  - 3) Reliability is core to all Quadro sol<sup>n</sup>. every prod. is designed to deliver peace of mind you need to focus on what you do best.

- IBM system has undergone an evolutionary change in that it supports ECC (Error correction code) making Quadro 5000 the 1st card with this capability.

### 3] Maximus Workstation :-

- Maximus platform, gives workstation users the ability to simultaneously perform complex analysis & visualization on single m/c.
- Supported by Kepler based GPU, Maximus delivers unparalleled performance & efficiency to professionals in fields as varied as manufacturing & visual effects.
- Maximus initially broke new ground as a single system that handles interactive graphics & compute intensive no. crunching required to simulate or render them resulting in dramatically accelerated workflows. Maximus unified technology transparently & automatically assigns visualization & simulation or rendering work to the right processor.

## Gaming platforms

- is also called as videogame platform or video game sys.
- it refers to specific combination of electronic & comp HW which in conjunction with SW, allows a videogame to operate. we should select the gaming platform carefully which will be based on mainly area of interest.
- The games could be in domain of entertainment, sport, movies, music, action games etc.
- all video games are tested on PC only.
- video game consoles are also used for playing video games.
- There are many companies in market which deals with gaming platforms.
- These platforms of companies are continuously updating the predecessors of their products every few years.
- Generally PC offers unique ability to upgrade HW, with some technical experience required, but without having to replace the entire comp. either by adding mem storage space, graphics cards, or by new motherboard. we can extend the life of our PC without spending high prices of a completely new comp.

- another decision that we should make when deciding on a platform is what type of video games we most enjoy and/or intend on playing.
- many action games thrive on console systems, there is huge variety of titles, much greater than what is available for PC, video games consoles also seem to get newer games titles first.
- Action games became popular on a game console before becoming available on pc platform, sports video games such as football, hockey, basketball are generally preferred to games consoles.
- A final consideration that may play into our decision is availability of games that have online game-play capabilities.
- games such as Age of Empires have thrived in multi player universe.
- virtually every pc has internet access allowing gamers to play online anytime.
- nowadays consoles are slowly coming onboard.

## Advance in Gaming

- 3ds MAX - Originally called 3D studio MAX.  
3ds max used in many industries that utilize 3D graphics. used in video game industry for developing models & creating cinema cut scenes.
- used in architectural visualizations because it is highly compatible with AUTOCAD - also developed by Autodesk. available for windows.
- AC3D - 3D modeling application: it is used in number of industries, math works actively recommends it in many of their aerospace related articles due to price & compatibility. it is first commercial 3D modeler to integrate full support for exporting models to the metaverse platform second life. available for MAC OSX, windows, Linux.
- Cinema 4D:- it is slightly lighter package than others in its basic configuration. s/w is claimed to be artist-friendly, & is designed with less technical user in mind.
- Maya - currently used in the film & TV industry, maya has high learning curve but has developed over years into an appl' platform in & of itself through extensibility via its MEL programming language.  
two versions
  1. maya complete
  2. maya unlimited

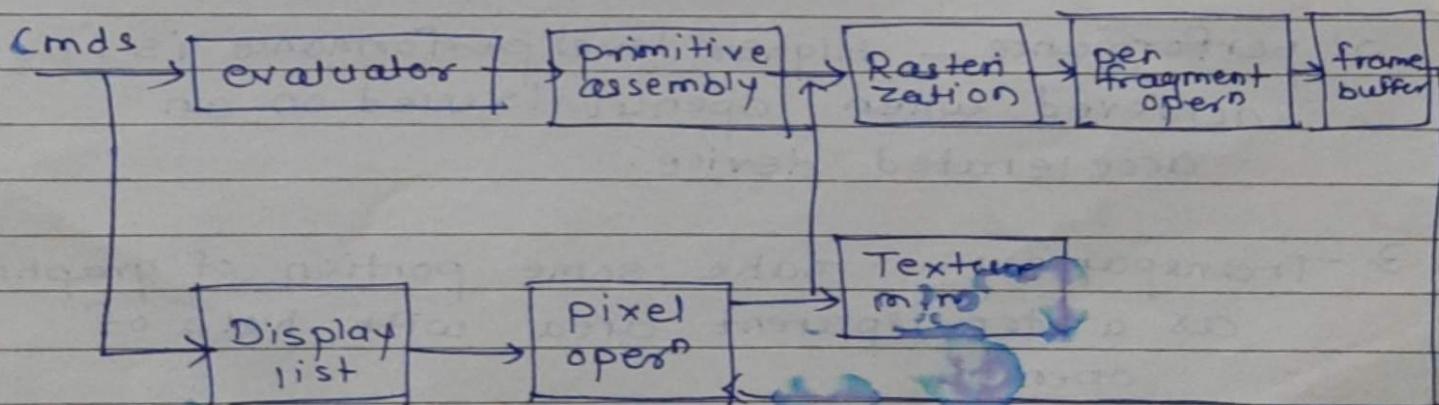
Graphics Tool :- [Reference - RGB]

dpu Interactive graphic Tool : OpenGL

- OpenGL is standard specification defining a cross lang, cross platform API for writing appl' that produce 2D & 3D comp. graphics.
- interface consists of over 250 diff. fun' calls which can be used to draw complex 3D scenes from simple primitives.
- main purpose is to render two & 3D object into frame buffer.
- OpenGL is low level, procedural API requiring the programmer to dictate the exact steps required to render a scene. In its basic oper, OpenGL accepts primitives such as points, lines & polygons & converts them into pixels via a graphics pipeline known as OpenGL state m/c.

### OpenGL operation:-

- block diag. of OpenGL data processing method.



- command enters from left & proceed through a processing pipeline. some commands specify geometric objects to be drawn & others control how the objects are handled during various stages.

### Functions & features of OpenGL :-

1. Display list :- contents of display file are preprocessed & executed more efficiently than same set of OpenGL commands executed in immediate mode.
2. feedback - is mode where OpenGL will return the processed geometric info to the appln as compared to rendering them into frame buffer.
3. Alpha blending - used to create transparent objects.
4. pixel opern :- diff. opern carried on pixels such as storing, mapping, zooming.

### features :-

1. scaling & Rotation - transformation can be achieved very easily without any problem.
2. performance - Higher level performance is achieved when OpenGL is used on an accelerated device.
3. Transparency - make some portion of graphics as a transparent area with help of OpenGL.
4. primitive opern - provides basic drawing primitive opern as point, line, polygon.