



# Bài 2

## CÁC HỆ THỐNG ĐỒ HỌA

---

Trịnh Thành Trung  
[trungtt@soict.hust.edu.vn](mailto:trungtt@soict.hust.edu.vn)

# NỘI DUNG

1. Tiến trình xử lý đồ họa
2. Phần mềm hệ đồ họa
3. Phần cứng hệ đồ họa

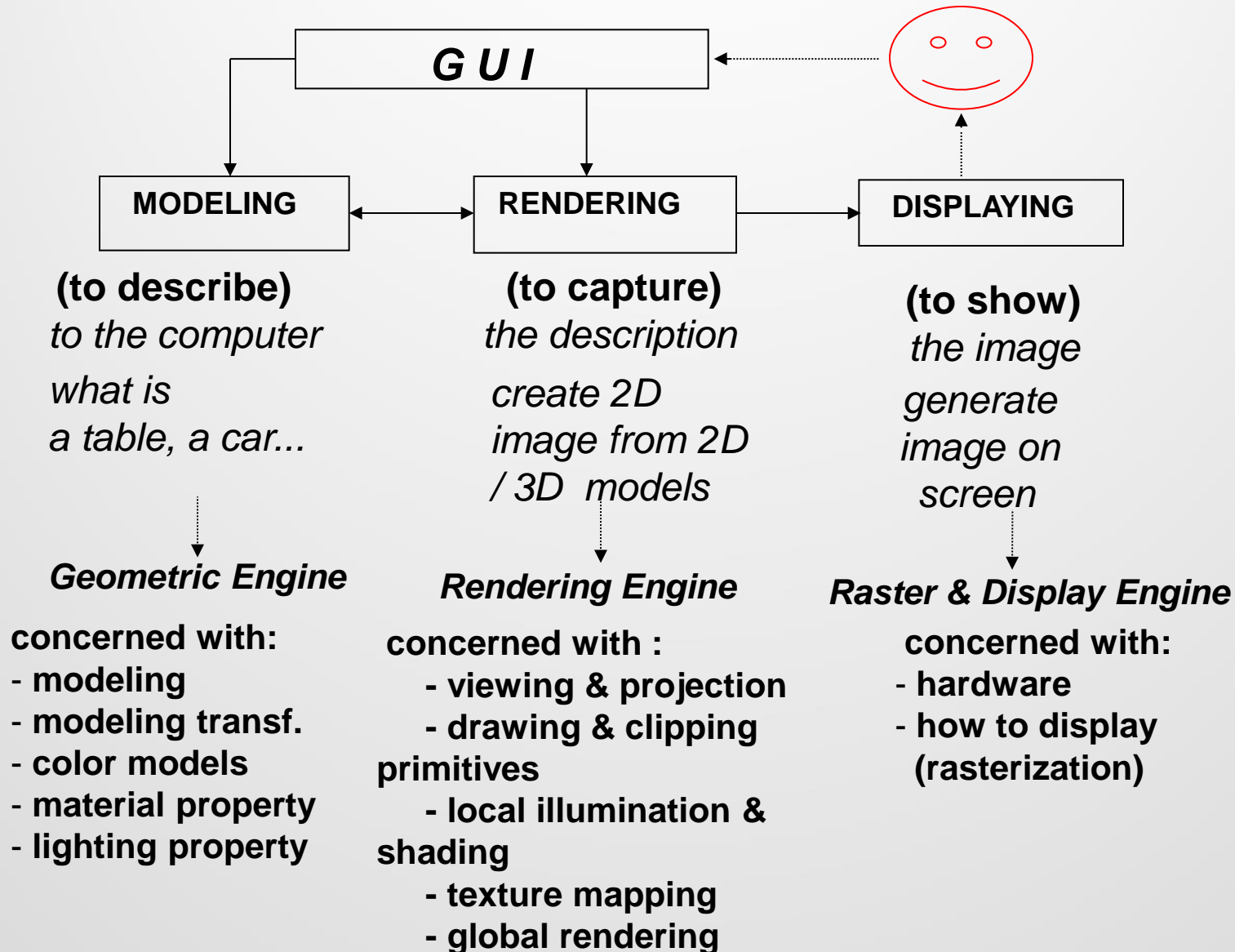


*1*

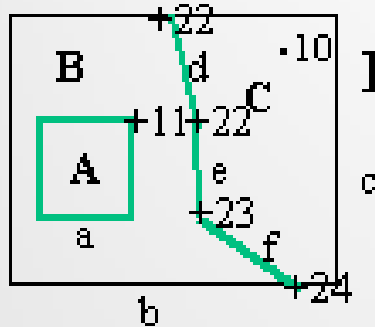
# TIẾN TRÌNH XỬ LÝ ĐỒ HỌA

---

# Tiến trình xử lý đồ họa



# Mô hình hóa



Data model

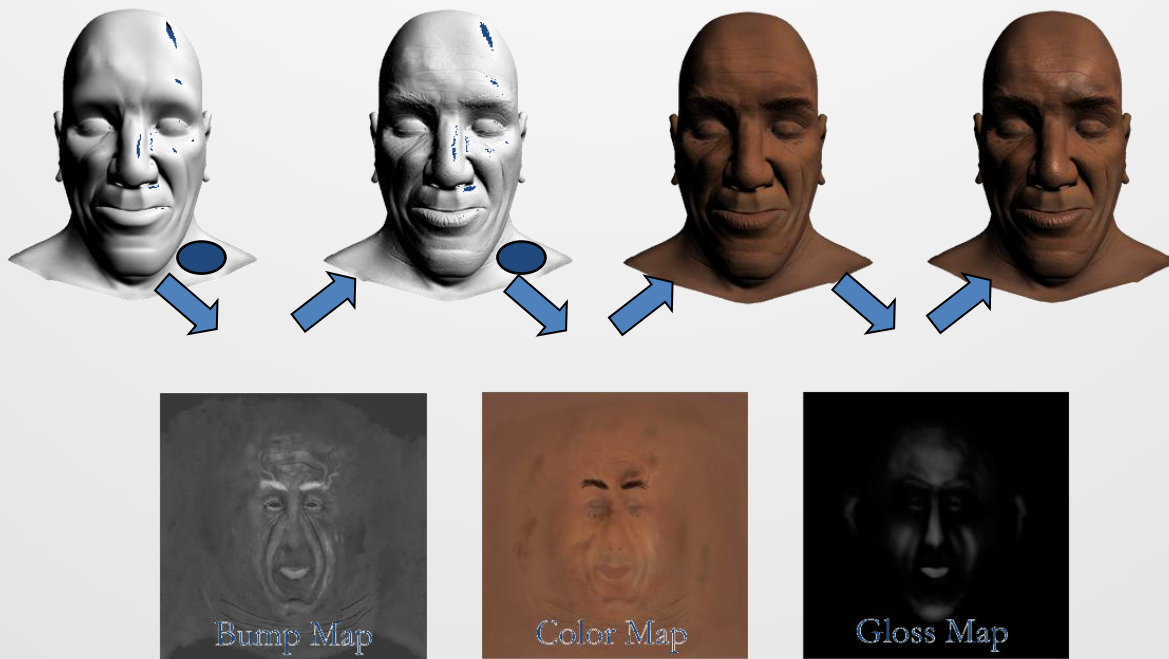
Polygon	Bounding chains
A	a
B	a,b
C	d,e,f,c

Data structure

Chain ID	From node	To node	Left poly.	Right poly.
a	11	11	A	B
b	22	24	B	--
....				
f	23	24	C	B

Node	x	y
10		
11		
...		
24		

# Tô trát



# Hiển thị

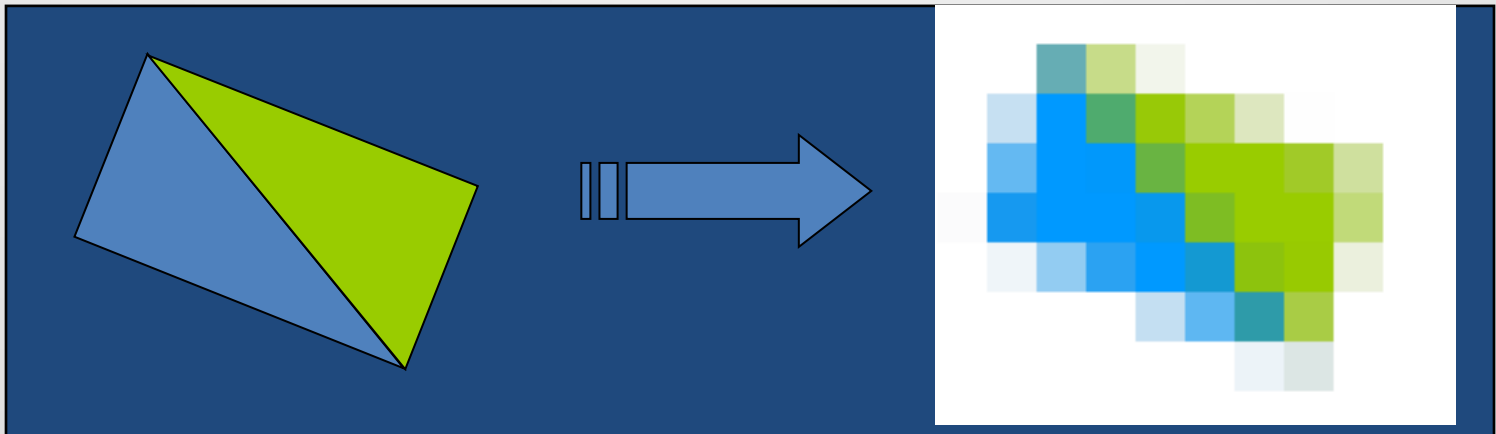
- Rời rạc hóa - Rasterizer

- Chuyển đổi các đối tượng hình học. (vertex) thành các biểu diễn ảnh. (fragment)

- Fragment = image fragment

- Pixel + associated data: color, depth, stencil, etc.

- Chấp nhận nội suy tạo các điểm ảnh



# Hệ thống đồ họa

- Phần cứng đồ họa:
  - Là tập hợp các thiết bị điện tử (CPU, bộ nhớ màn hình) giúp cho việc thực hiện các phần mềm đồ họa.
- Phần mềm đồ họa hệ thống:
  - Là tập hợp các lệnh đồ họa của hệ thống (graphics output commands),
  - Thực hiện công việc hiển thị cái gì (what object) và chúng sẽ được hiển thị như thế nào (how).
  - Phần mềm đồ họa hệ thống là phần mềm xây dựng trên cơ sở một thể loại phần cứng nhất định và phụ thuộc vào phần cứng.



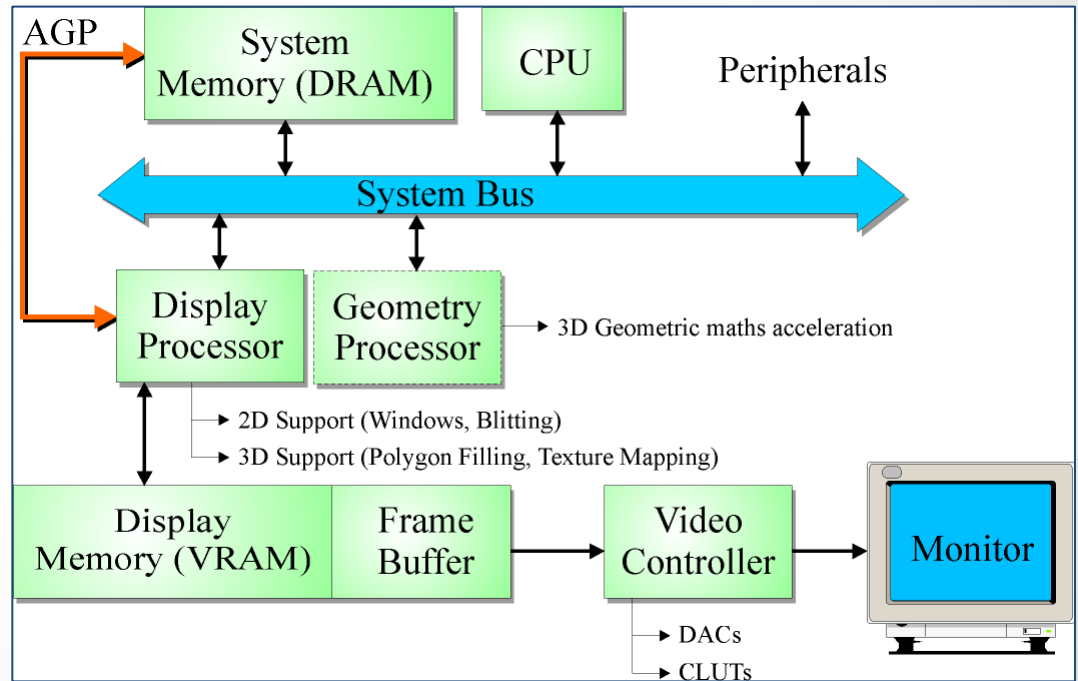


# 2

## PHẦN CỨNG HỆ ĐỒ HỌA

---

# PHẦN CỨNG HỆ ĐỒ HỌA



# Chức năng nhiệm vụ

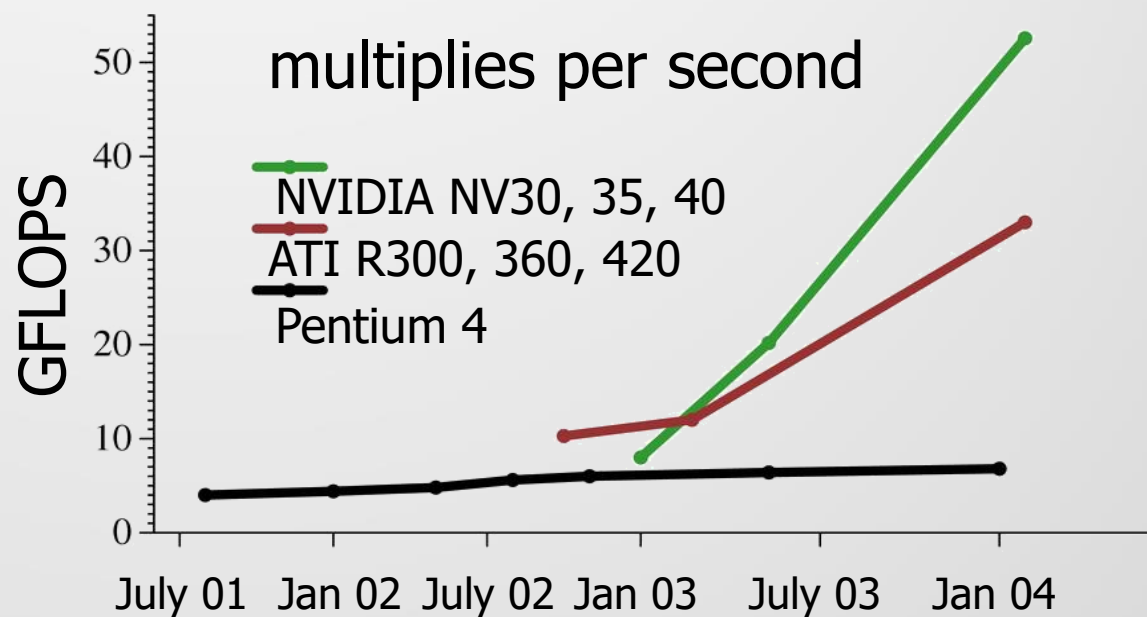
- CPU: thực hiện các chương trình ứng dụng.  
Bộ xử lý hiển thị (Display Processor): thực hiện công việc hiển thị dữ liệu đồ họa.
- Bộ nhớ hệ thống (System Memory): chứa các chương trình và dữ liệu đang thực hiện.

# Chức năng nhiệm vụ

- Bộ đệm (Frame buffer): có nhiệm vụ chứa các hình ảnh hiển thị.
- Bộ điều khiển màn hình (Video Controller): điều khiển màn hình, chuyển dữ liệu dạng số ở frame buffer thành các điểm sáng trên màn hình.

# Vi xử lý đồ họa

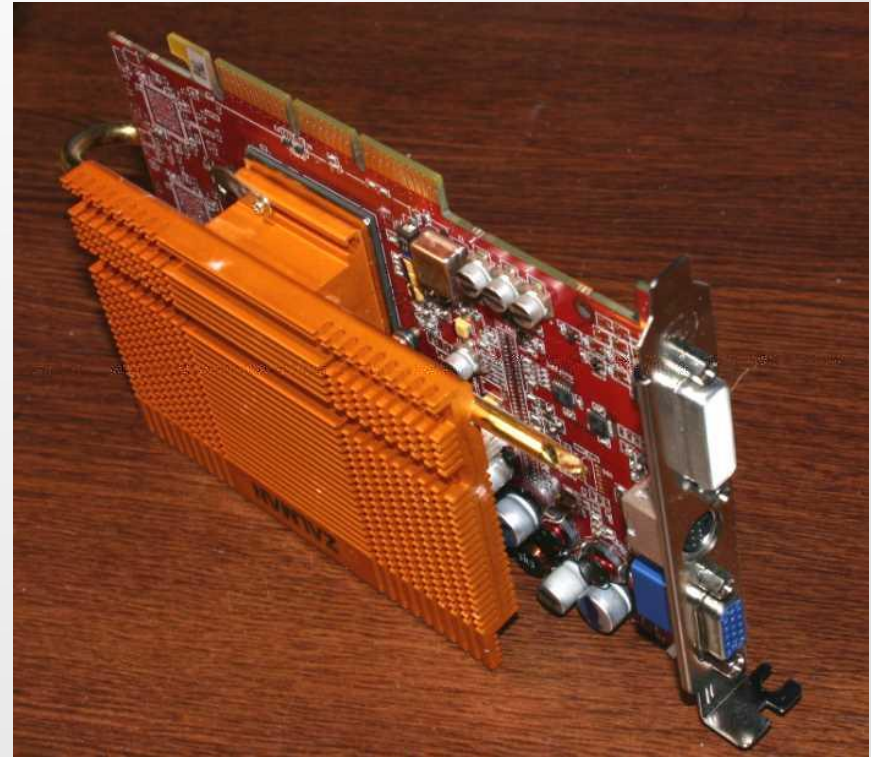
- GPU: thành phần xử lý chính trên bo mạch đồ họa với mục đích tăng tốc và phù hợp với các phần mềm đồ họa.
- Đặc điểm chính
  - Programmability
  - Precision
  - Power



# Vi xử lý đồ họa

## Ưu điểm

- Modern GPUs có khả năng lập trình
  - Lập trình trên pixel, vertex, video engines
  - Hỗ trợ lập trình với các ngôn ngữ bậc cao
- Modern GPUs hỗ trợ độ chính xác cao
  - Hỗ trợ 32 bit floating point trên pipeline
  - Bộ nhớ



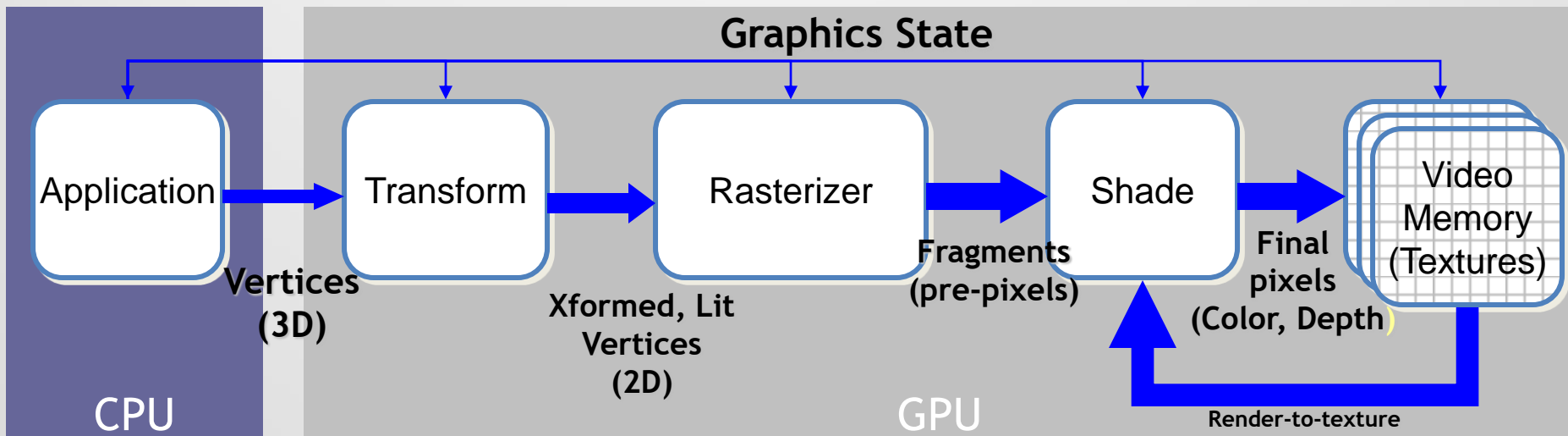
# Vi xử lý đồ họa

## Nhược điểm

- GPU được thiết kế chuyên cho video games
  - Mô hình lập trình đặc biệt theo hướng computer graphics
  - Môi trường lập trình chặt chẽ về cú pháp và câu lệnh
- Kiến trúc mức thấp:
  - Thường song song
  - Phát triển liên tục
  - Công nghệ bí mật của các hãng
- Không đơn giản như viết code cho CPU

# Trong tiến trình xử lý đồ họa

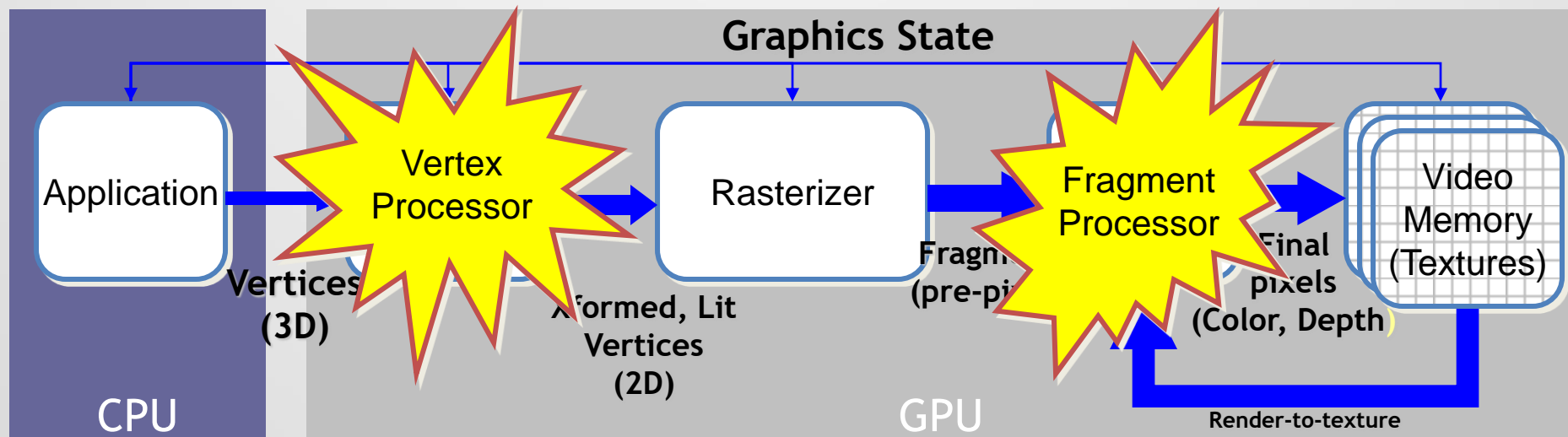
- GPU trong tiến trình xử lý đồ họa
  - Luồng luôn thay đổi độ và chuyển đổi dữ liệu giữa các phần nhiều
  - Nhiều caches, FIFOs, và nhiều vấn đề khác





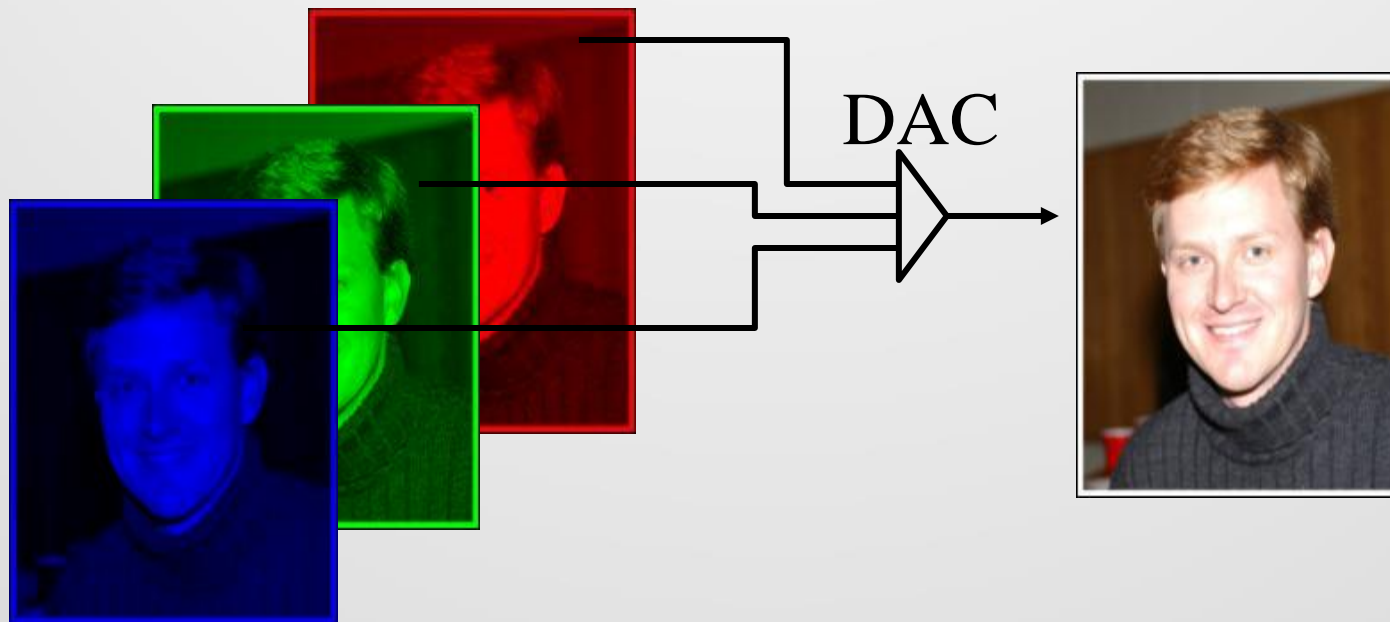
# Trong tiến trình xử lý đồ họa

- GPU hiện đại trong tiến trình xử lý đồ họa
  - bổ sung thêm vertex processor và fragment processor



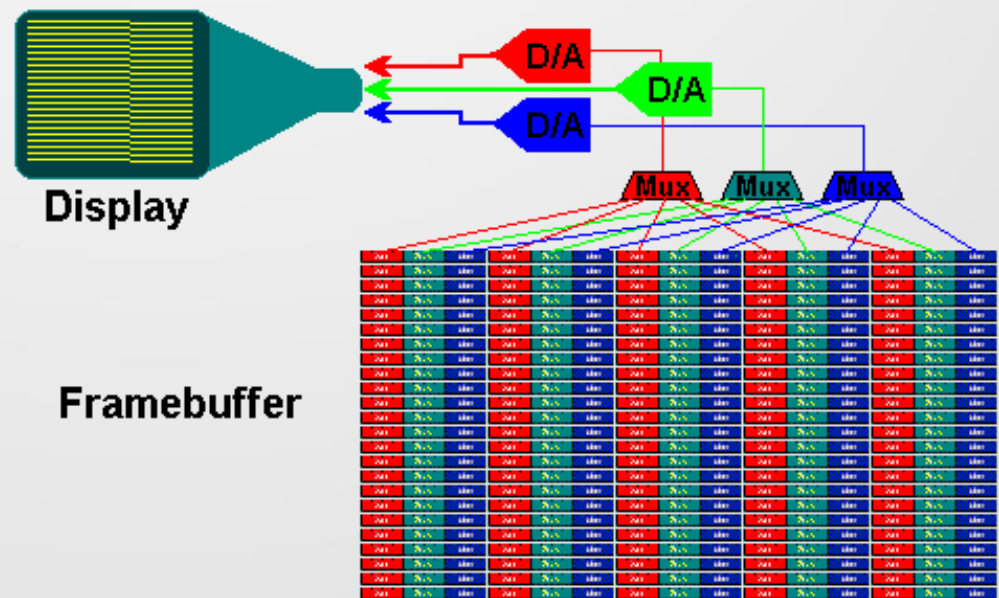
# Bộ đệm màn hình (Framebuffer)

- Vùng bộ nhớ chứa dữ liệu cho hình ảnh hiển thị
- Các màu sắc được lưu trữ độc lập trong bộ đệm - framebuffer
- 24 bits per pixel = 8 bits red, 8 bits green, 8 bits blue
  - 16 bits per pixel = ? bits red, ? bits green, ? bits blue



# Bộ đệm màn hình (Framebuffer)

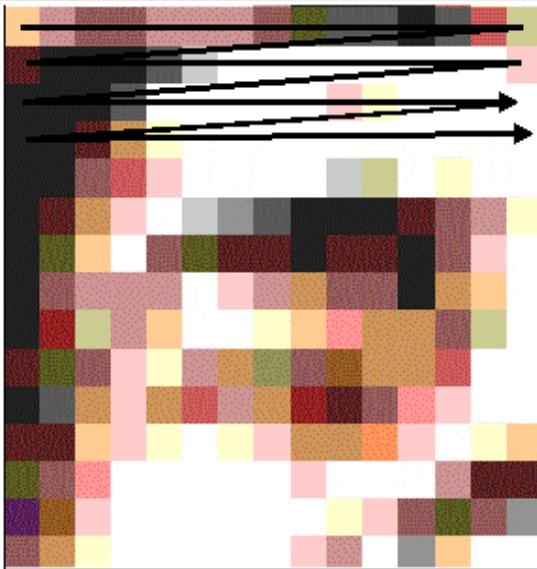
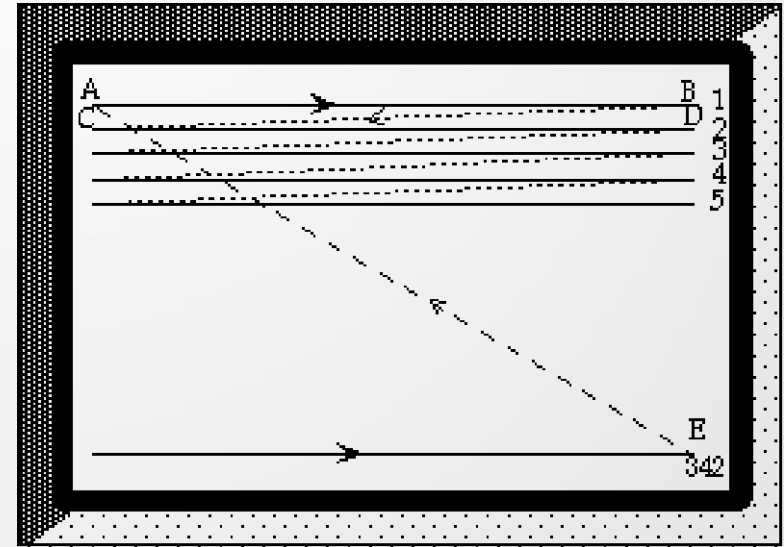
- *True-color* ( *24-bit* hoặc *32-bit* ) *framebuffer* lưu trữ một byte cho red, green hoặc blue
- Do đó, mỗi pixel có thể có được  $2^{24}$  màu



# Bộ đệm màn hình (Framebuffer)

- $X : 0 \div X_{\max}$  2 màu/ 1 bit
- $Y : 0 \div Y_{\max}$  16 màu/ 4 bit
- 256 màu/ 8bit
- 216 màu/ 16 bit
- 224 màu/ 24 bit
- $640 \times 480 \times 16 \rightarrow$  Video RAM = 2MB
- $1024 \times 1024 \times 24 \rightarrow$  Video RAM = 24MB

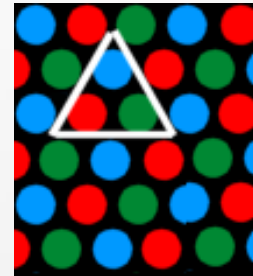
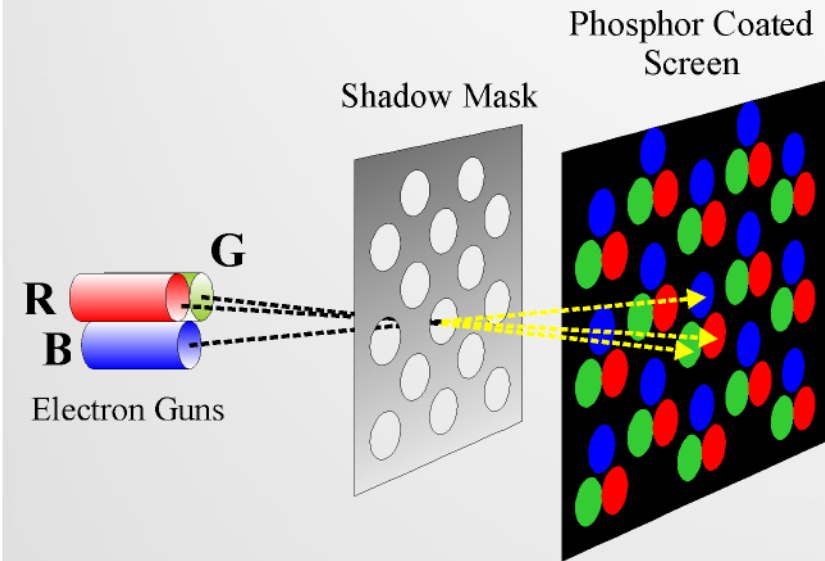
# Các thiết bị hiển thị



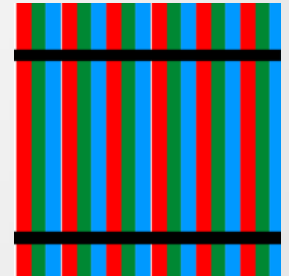
*tất cả các thiết bị  
hiển thị đồ họa  
đều là dạng điểm*



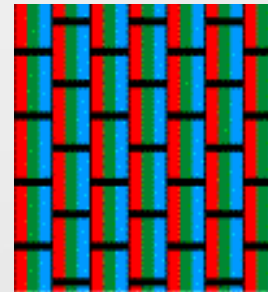
# Hiển thị



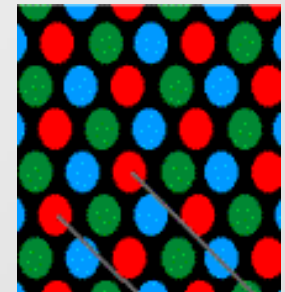
Standard Dot-trio



SONY Trinitron CRT



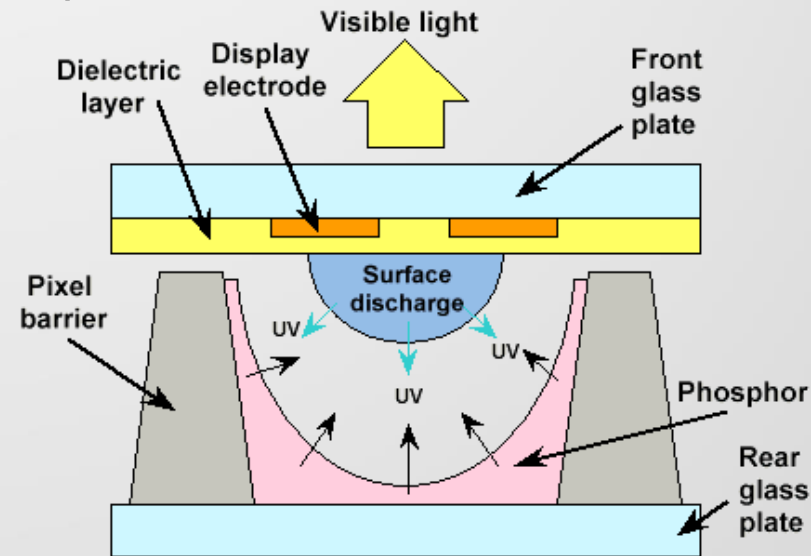
NEC Hybrid Mask



Hitachi EDP

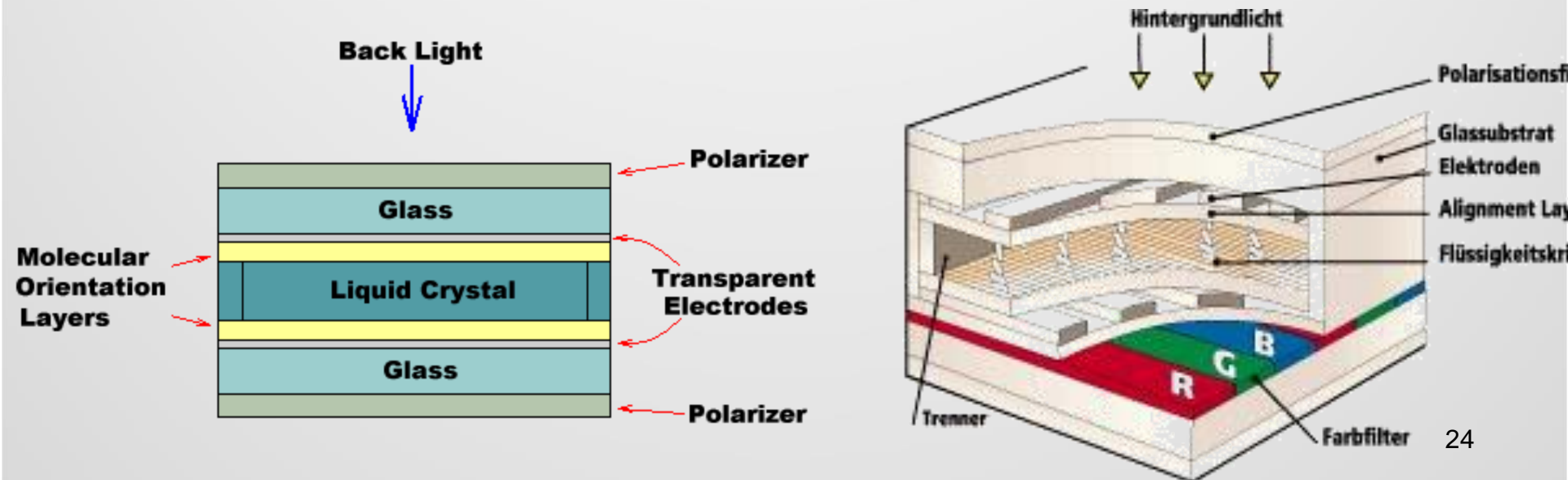
# Màn hình Plasma

- Plasma display panels
  - Similar in principle to fluorescent light tubes
  - Small gas-filled capsules are excited by electric field, emits UV light
  - UV excites phosphor
  - Phosphor relaxes, emits some other color



# Display Technology: LCDs

- Transmissive & reflective LCDs:
  - LCDs act as light valves, not light emitters, and thus rely on an external light source.
  - Laptop screen: backlit, transmissive display
  - Palm Pilot/Game Boy: reflective display





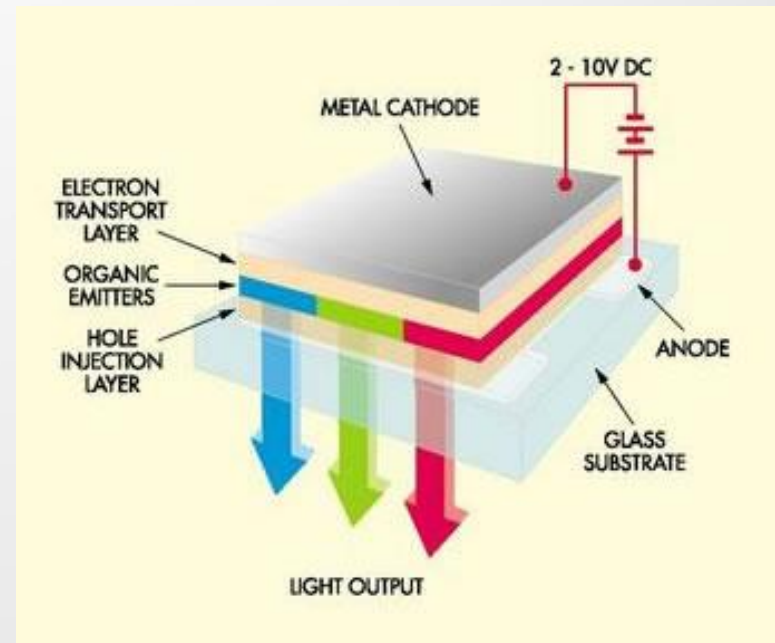
# Màn hình hữu cơ (OLED)

## •Organic Light-Emitting Diode (OLED) Arrays

- OLEDs hoạt động tương tự cơ chế của LEDs bán dẫn

## •Cấu trúc là màng chất dẻo mỏng:

- Màng film cấu tạo bởi các phần tử hữu cơ, các phân tử phát sáng bởi sự thăng hoa khí trong môi trường chân không
- Màu sắc được tạo thành từ các lớp màu gồm các phân tử huỳnh quang được kích thích.
- Mịn, không to như các hạt tinh thể và không phát nhiệt
- Có thể tạo màn hình rộng loại OLEDs

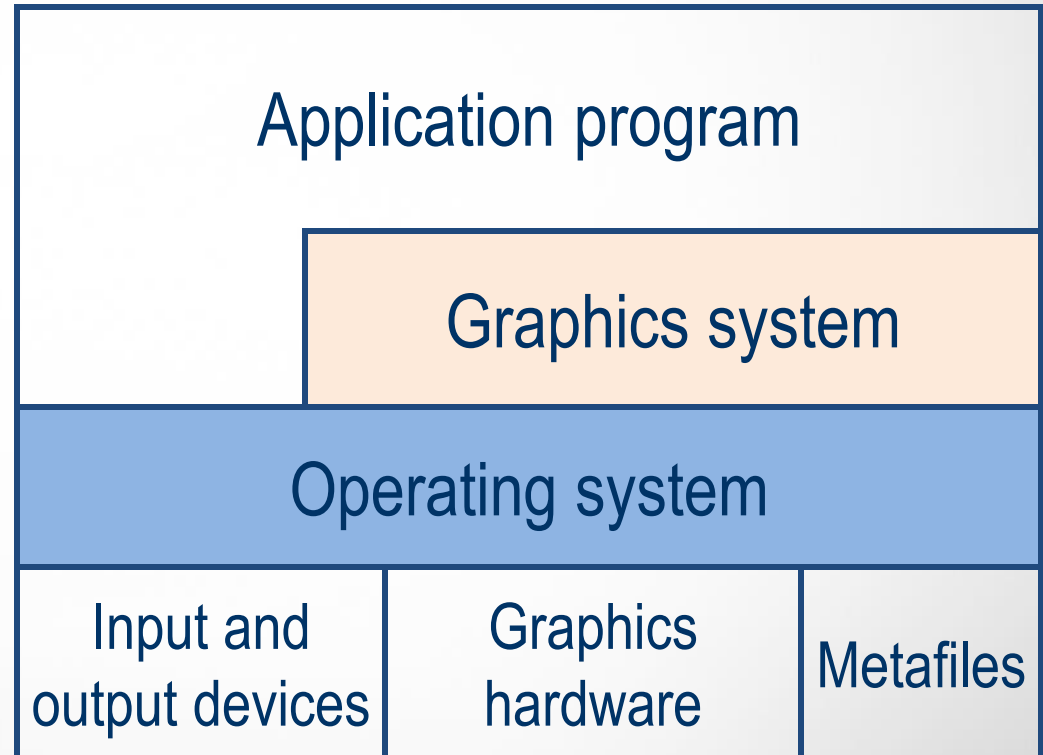


# 3

## PHẦN MỀM HỆ ĐỒ HỌA

---

# PHẦN MỀM HỆ ĐỒ HỌA



# Các chuẩn giao diện

- GKS (Graphics Kernel System): chuẩn xác định các hàm đồ họa chuẩn, được thiết kế như một tập hợp các công cụ đồ họa hai chiều và ba chiều.
  - GKS Functional Description, ANSI X3.124 - 1985.
  - GKS - 3D Functional Description, ISO Doc #8805:1988.
- CGI (Computer Graphics Interface System): hệ chuẩn cho các phương pháp giao tiếp với các thiết bị ngoại vi.
- CGM (Computer Graphics Metafile): xác định các chuẩn cho việc lưu trữ và chuyển đổi hình ảnh.

# Các chuẩn giao diện

- VRML (Virtual Reality Modeling Language): ngôn ngữ thực tại ảo, một hướng phát triển trong công nghệ hiển thị được đề xuất bởi hãng Silicon Graphics, sau đó đã được chuẩn hóa như một chuẩn công nghiệp.
- PHIGS (Programmers Hierarchical Interactive Graphics Standard): xác định các phương pháp chuẩn cho các mô hình thời gian thực và lập trình hướng đối tượng.
  - PHIGS Functional Description, ANSI X3.144 - 1985.
  - PHIGS+ Functional Description, 1988, 1992.

# Các chuẩn giao diện

## Các chuẩn không chính thức

- **OPENGL thư viện đồ họa của hãng Silicon Graphics, được xây dựng theo đúng chuẩn của một hệ đồ họa.**
  - SGI's OpenGL 1993
- **DIRECTX thư viện đồ họa của hãng Microsoft**
  - Direct X/Direct3D 1997