GRAPHICS TECHNOLOGY FOR CHEMISTRY APPLICATIONS

Internal Report C89-43

G. LOMBARDI
P. PALAMIDESE

INTRODUCTION TO MOLECULAR GRAPHICS

With this short introduction I will try to underline a few main points about computer graphics. It seems likely that in the near future the typical working environment for a researcher will be a workstation in a stand-alone configuration or on a network and graphics tools or visualization methodologies will be very effective steps in the computational cycle.

The true role of computer graphics was deeply understood by Sutherland since 1965, when he said:
The screen is a window through which one sees a virtual world. The challange is to make that world look real, act real, sound real, feel real.

All c.g. techniques developed so far have tried to reach these goals. We have developed:

- models to represent physical objects and natural phenomena
- many different algorithms for rendering
- animation techniques
- turn-key packages for specific applications
- basic software with standard functionalities (i.e. PHIGS)

But all that has many limits, from the point of view of a scientist. For example, turn-key packages are not extensible; standard software requires a lot of time to a scientist who wants to develop his own application.

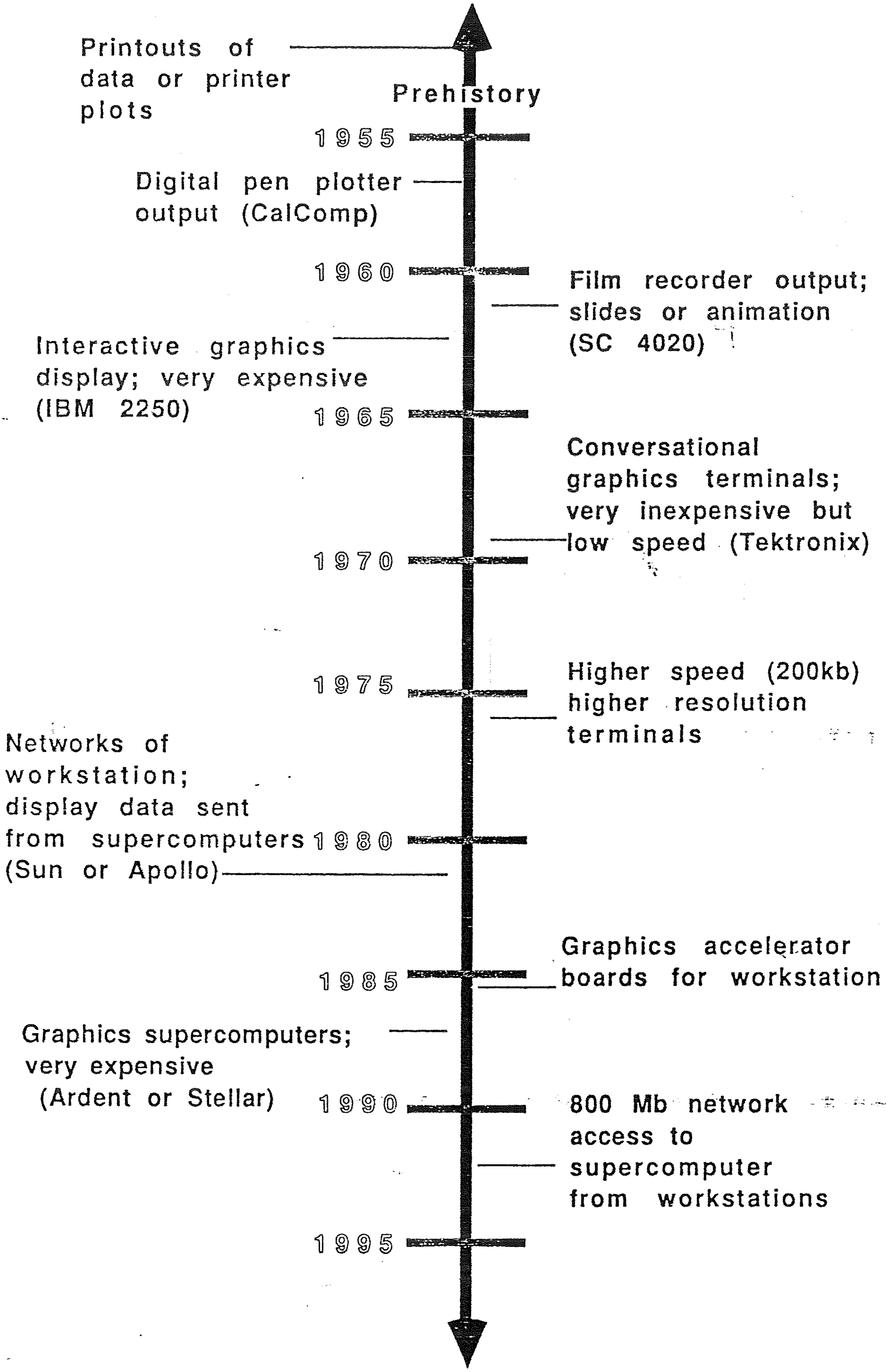
The trend, now, is toward the development of a new generation of graphics software made as a *high level, extensible, graphics toolkits.*

The scientist is no longer seen as an "end user" but as one candidate to build his own modules from high level functions. Hence, in the near future, he will need to be familiar with many languages and systems which are de facto standard at any workstation: for example the operating system UNIX, the programming language C or C++ which is becoming a developing language even in scientific area and will eventually substitute FORTRAN, the x-window system which allows the user to control several applications running on different machines.

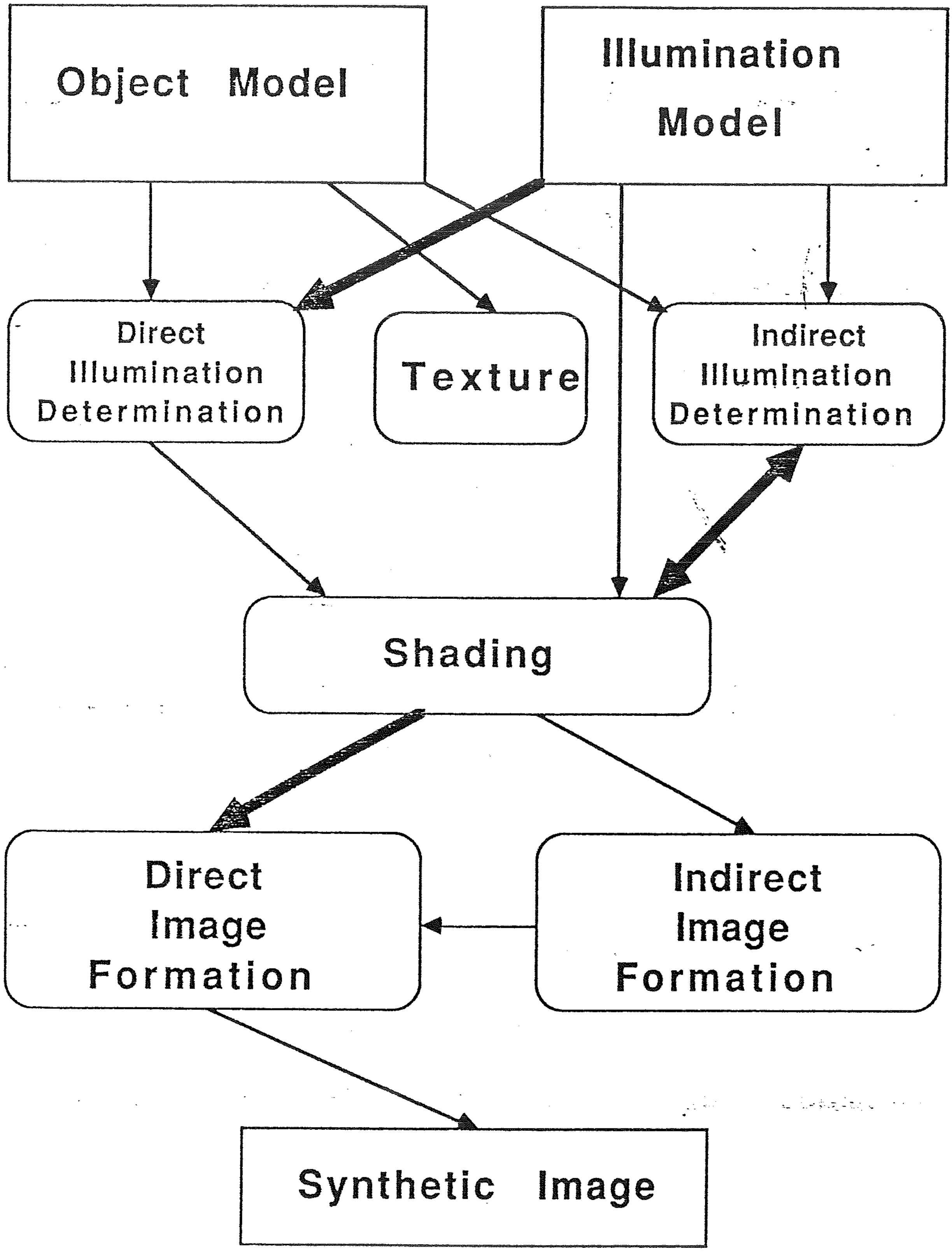
But the main problem a scientist has to face with, is integration of applications. While traditionally a program was a collection of instructions, a "scientific program" nowadays is a collection of applications. From simulation to animation there are many steps which should be integrated in an "easy to use" methodology.

One goal of this workshop is to discuss the role of c.g. in many chemical areas and find out new requirements for future c.g. developments.

EVOLUTION OF VISUALIZATION CAPABILITIES



PROCESSES TO CREATE SYNTHETIC IMAGES



GLOSSARY

Bump maps: A *texture map* which contains perturbation of the normal vector of the surface.

Diffuse reflections: the component of reflection that represents light scattered equally in all directions.

Direct illumination: the light that reaches objects directly from one or more of the light sources. Those parts of objects that do not receive direct illumination are in *shadow.*

Illumination model: the process of determining how much light reaches the viewer from each point in the scene based on the light sources present, the viewer's position, and the surface properties and locations of the objects in the scene.

Images synthesis: that part of computer graphics that addresses the production of realistic images. Images synthesis includes illumination, image formation, and shading.

Indirect illumination:light that reaches an object by reflecting from or refracting through other objects.

Motion blur: the blur seen on photographs of moving objects which is the result of the camera lens being open for a finite period of time.

Point light source: a mathematical light source consisting of light coming from a single point, which is used in most illuminations models.

Radiosity: the total light flux leaving a surface, including the self-emitted light and the reflected and transmitted incident light.

Radiosity method: a method for approximating the indirect illumination within an environment based on a equilibrium energy balance in an enclosure.

Rendering: the process of converting the description of a scene into the color values of the individual pixels of a display surface.

Shader: the portion of the graphics system, responsible for the determination of the light intensity transmitted to the camera position, taking into consideration direct and indirect illumination and the position of the camera in the scene.

Shadow: those parts of a scene that are visible from the viewer's position but not from the position of a light source. The shadow is made up of two components: the umbra, in which the light source is totally obscured, and the penumbra, in which the light source is partially obscured.

Specular reflection: light reflected directly from the outer surface of an object typically modeled as a highlight on the object.

Texture maps: arrays containing intensity information which are mapped onto primitives during the final stages of rendering.

Texturing: a technique developed to allow the addition of texture (surface features) to a primitive such as a polygon or a surface patch.

Translucency: blurred transparency. Images seen through translucent objects are not as distinct or sharp as objects seen through transparent objects.

Transparency: a property of materials that allows light to be transmitted through objects.

Accepted and emerging graphics standards

Today we have four approved graphics standards:

GKS	Graphical Kernel System
GKS-3D	3D Graphical Kernel System
CGM	Computer Graphic Metafile
PHIGS	Programmer's Hierarchical Interactive Graphics System

Graphics standards committees within both ISO and ANSI are also working on several emerging standards:

GKS-91	Revision to the GKS standard
PHIGS+	Extension to PHIGS
CGI	Computer Graphic Interface
CGM add 1	Extension to CGM for support of GKS
CGM add 2	3D extension to the CGM standard

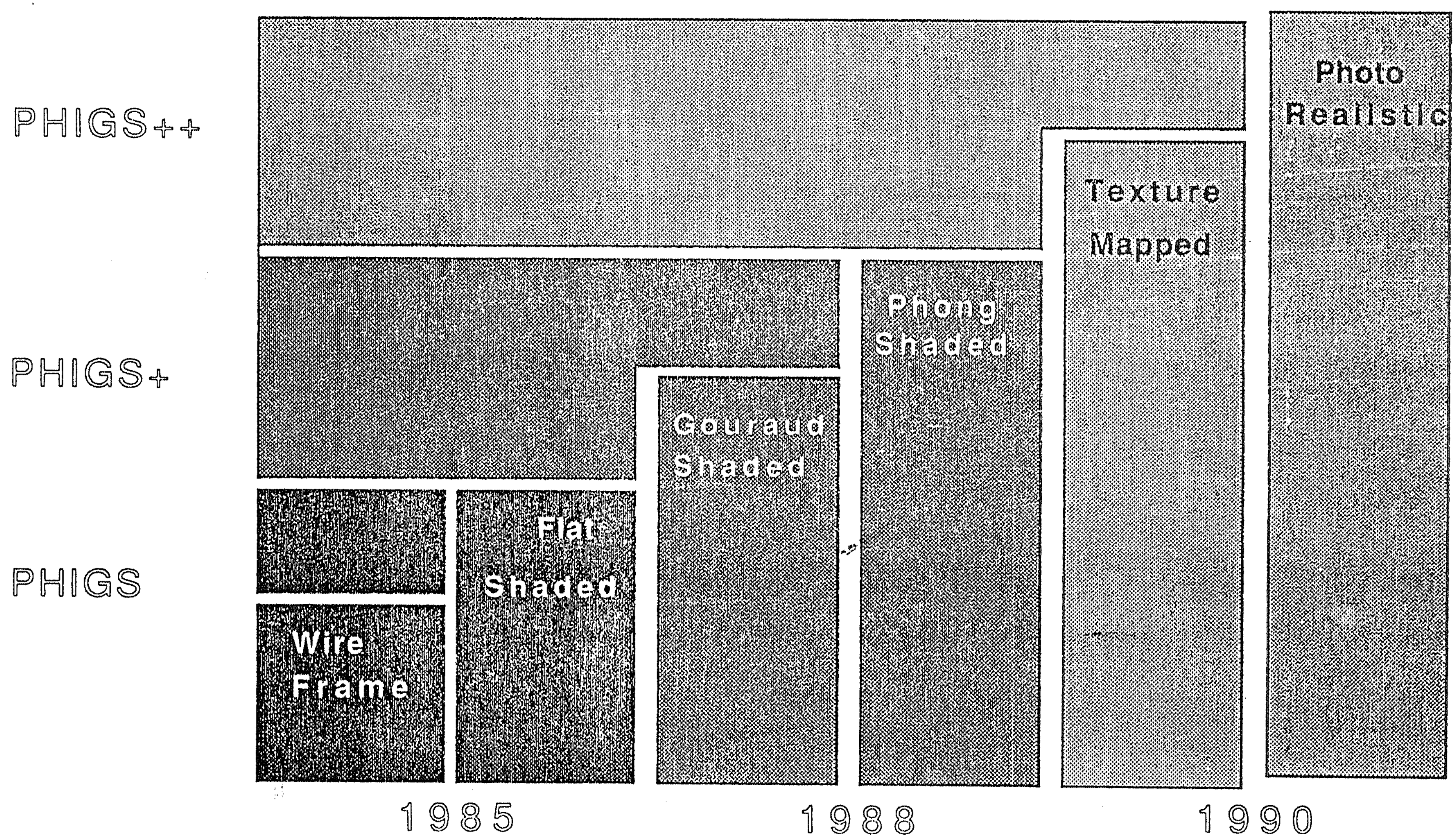
Many packages have also emerged as de-facto standards:

GDDM	IBM's graphics application programming package
PLOT-10	Tektronix' graphics programming package
Calcomp	The de-facto pen plotter standard
PostScript	Documentation graphics language
Regis	Digital Equipment's graphics system
DI-3000	PVI's CORE implementation
DISSPLA	Computer Associates' graphics package
UNIRAS	Device independent raster graphics
Template	Template's CORE implementation

A large number of de-facto standards have emerged for window handling and high-end 3D graphics:

X-Window	A device independent window system
PEX	A 3D extension of X-Window
Display Postscript	Protocol for transferring graphical data
Dore'	Ardent
GL2	Silicon Graphics' 3D Graphics Language
Starbase	HP's 3D graphics language

The development of rendering technique during the last decade



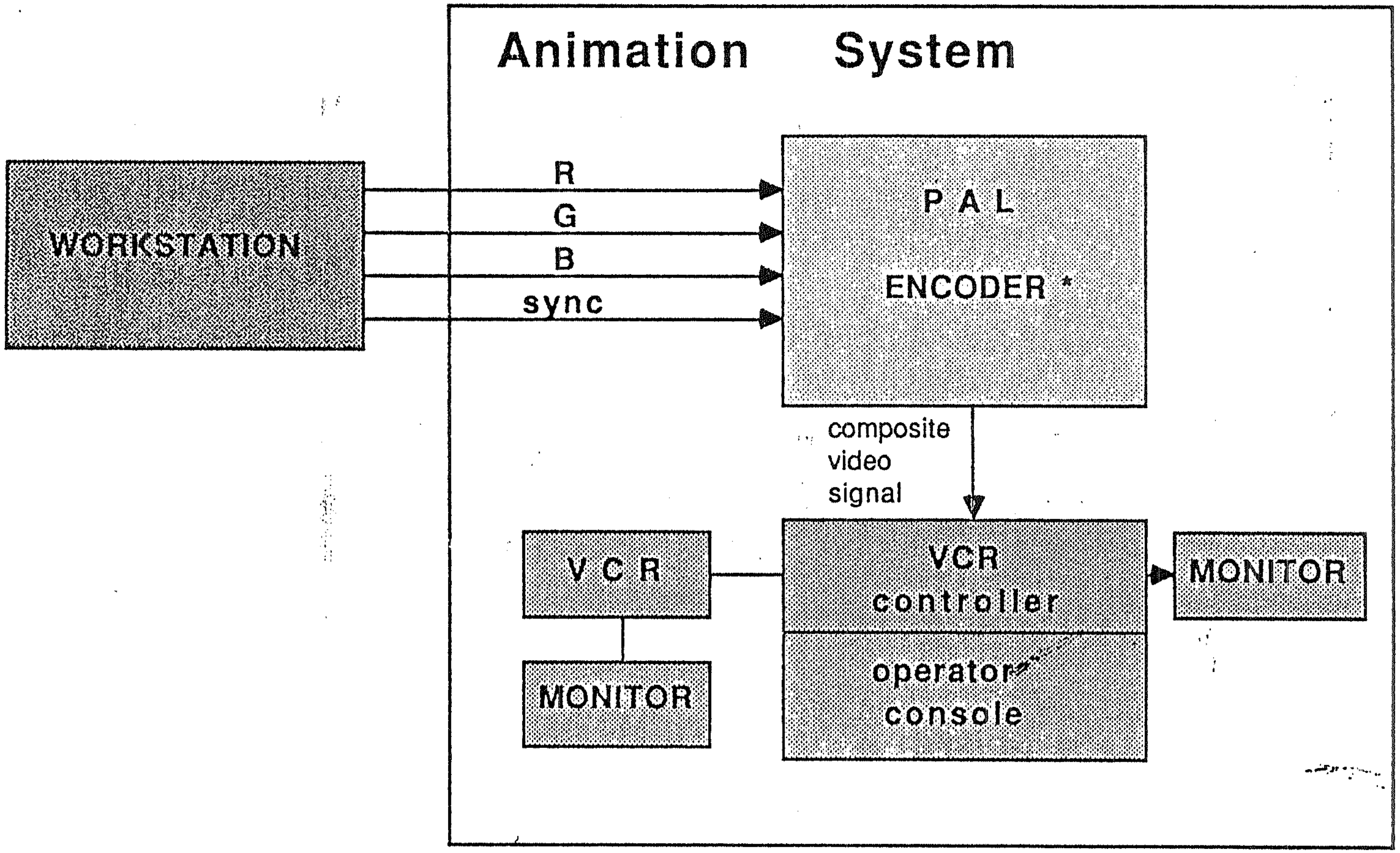
Animation of complex scientific simulations can be accomplished by recording images on a medium and then displaying the images at a rate of about 25 images per second.

Laser video disks require a long and costly mastering process.

WORM disks data transfer time exceeds real time animation requirements.

Most widely used media are Video Cassette Recorders (magnetic technology).

Animation Hardware Configuration for post production



*encoding= create composite signal from primaries

What can be animated ?

The typical animation procedure takes place in the domain of geometric and features representation.

<u>Objects</u>

- locations
- orientation
- size
- shape
- color
- transparency

<u>Cameras</u>

- viewer position
- point of interest
- view angle (zoom)

<u>Light sources</u>

- intensity
- location
- color

<u>Environments</u>

- dust
- fog
- haze
- smog
- steam

STATIC PICTURES versus ANIMATION versus REAL TIME INTERACTION

SIMULATION is a special way to perform animation that involves modeling the physical laws that actually affects objects, as opposed to positioning objects where they are supposed to be.

AN ANIMATION SYSTEM INCLUDES TWO MAIN "CHARACTERS":

- actors (objects, moving events)
  see next page
- animator (controls actors' motion; can be the scientist, in a simulation system)

MOTION CONTROL METHODS

- scripting systems (more sophisticated motion control requires more complicated motion semantics)
- artificial intelligence systems (under experimentation)

Main methods for broadcasting audio and video signals

SIGNAL	COUNTRY	RESOLUTION	REFRESH FREQUENCY
PAL	Europe India China Australia	625 lines	25 Images/Sec
NTSC	North America Japan	525 lines	30 Images/Sec
SECAM	France U.S.S.R. Eastern Europe		25 Images/Sec

Stand-alone workstations

Company	CPU	CPU memory in Mbytes	Floating-point Performance in Mflops	Operating System
Alliant VFX/40	1-4 64-bit processors	32-128	2.4 (1 processor)	Unix
Alliant VFX/80	1-8 64-bit processors	32-224	2.4 (1 processor)	Unix
Apollo 10000VS	1-4 64-bit processors	8-128	5.1 (1 processor)	Unix
Ardent Titan	1-4 64-bit processors	16-128	6.5 (1 processor)	Unix
DEC VAXstation 3520	2 32-bit processors	8-64	NA	VMS or Unix

for scientific visualization

Monitor Resolution in pixels	Colors Displayed/ Colors Available	Comments
1,280 x 1,024	16.7M/16.7M	Configurable with up to 8 independent Graphics processors; 1-4 users.
1,280 x 1,024	16.7M/16.7M	Configurable with up to 16 independent Graphics processors; 1-8 users.
1,280 x 1,024	16.7M/16.7M	Two-board graphics subsystem; 3D RISC drawing engine; does not require use of proprietary graphics interface.
1,280 x 1,024	16.7M/16.7M	Accepts Cray and VAX source code; bundled with visualization environment and Dore graphics library.
1,280 x 1,024	256/256	Graphics accelerator; accepts PHIGS;

1,280 x 1,024	256/256	16.7 color upgrade available
1,280 x 1,024 1,280 x 1,204 1,280 x 1,024	256/16.7M 256/16.7M 256/16.7M	Graphics subsystem; RISC architecture; lower end field upgradable to heigher end; 16.7M displayable colors option available.
1,280 x 1,024 1,280 x 1,024 1,280 x 1,024	16.7M/16.7M 16.7M/16.7M 16.7M/16.7M	Graphics subsystem; RISC architecture; board-swapping upgrade path; video digitizer board available.
1,280 x 1,024	16.7M/16.7M	Comes with Application Visualization System software.
1,280 x 1,024 1,280 x 1,024	16.7M/16.7M 16.7M/16.7M	Comes with Application Visualization System software; dual-user option available.
1,152 x 900 1,024 x 1,024	16.7M/16.7M	Video from TAAC-1 accelerator board can be displayed in a window or on a separate monitor.

DEC VAXstation 3540	4 32-bit processors	8-48	NA	VMS
Hewlett-Packard 9000				
825SRX	1 32-bit processor	8-96	0.53	Unix
835SRX	1 32-bit processor	8-96	1.8	Unix
835 TurboSRX	1 32-bit processor	8-96	1.8	Unix
Silicon Graphics Power Iris				
4D/120GTX	2 32-bit processors	8-128	NA	Unix
4D/220GTX	2 32-bit processors	8-128	NA	Unix
4D/240GTX	4 32-bit processors	8-128	NA	Unix
Stellar GS1000	4 64-bit processors	16-128	9.8	Unix
Stellar GS2000	4 64-bit processors	16-128	NA	Unix
Stellar GS2500	5 64-bit processors	16-128	NA	Unix
SUN Sparcstation 370 with TAAC-1	1 32-bit processor	8-56	NA	Unix

REFERENCES

"Computer Animation", EUROGRAPHICS '89, Tutorial Notes

"Introduction to Computer Animation", SIGGRAPH '88, Course n. 3

"Scientific Visualization: Bringing data into focus", COMPUTER, August '89

"Visualization of scientific data", EUROGRAPHICS '89, Tutorial

A. Crise, P. Palamidese, "Animazione: applicazioni tradizionali e settori emergenti", PIXEL, Ottobre '89

N. Magnenat-Thalmann, D. Thalmann, "Image Synthesis", Springer Verlag, Tokyo '87