Computer Graphics
Course Introduction

Konstantin Tretyakov
kt@ut.ee

Organization

Organization

Organization

Lectures (Wednesdays)

Geometry & Linear algebra, Algorithms, Sampling, Modeling

Organization

Lectures (Wednesdays)

Geometry & Linear algebra, Algorithms, Sampling, Modeling

Practice sessions (Mondays)

Allegro, OpenGL, Blender, Unity3D

Organization

Lectures (Wednesdays)

Geometry & Linear algebra, Algorithms, Sampling, Modeling

Practice sessions (Mondays)

Allegro, OpenGL, Blender, Unity3D

Project (Sep 22)

- In teams of 2-3 people
- Related to CG
- Open-source
- Software + write-up + demo

Organization

Lectures (Wednesdays)

Geometry & Linear algebra, Algorithms, Sampling, Modeling

Practice sessions (Mondays)

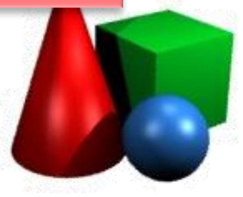
Allegro, OpenGL, Blender, Unity3D

Project (Sep 22)

- In teams of 2-3 people
- Related to CG
- Open-source
- Software + write-up + demo

Exam (January)

"B-spline equation:
p(t) = ________", etc.



Organization

Lectures (Wednesdays)

Geometry & Linear algebra, Algorithms, Sampling, Modeling

Practice s (Mondays)

Allegr ender,

40

Proj 22)

- In tea
- Relate
- Open-s
- Software + p + demo

30

Ex v)

“B on:

p______”, etc.

30

What if I have a question?

- Mailing list:
  - **aine.ati.arvutigraafika@lists.ut.ee**
- Personally:
  - Konstantin (**kt@ut.ee**)
  - Ilya (**ilya.kuzovkin@gmail.com**)

What if I forget all that?

What if I forget all that?

http://courses.cs.ut.ee/2013/cg

Organization

Questions?

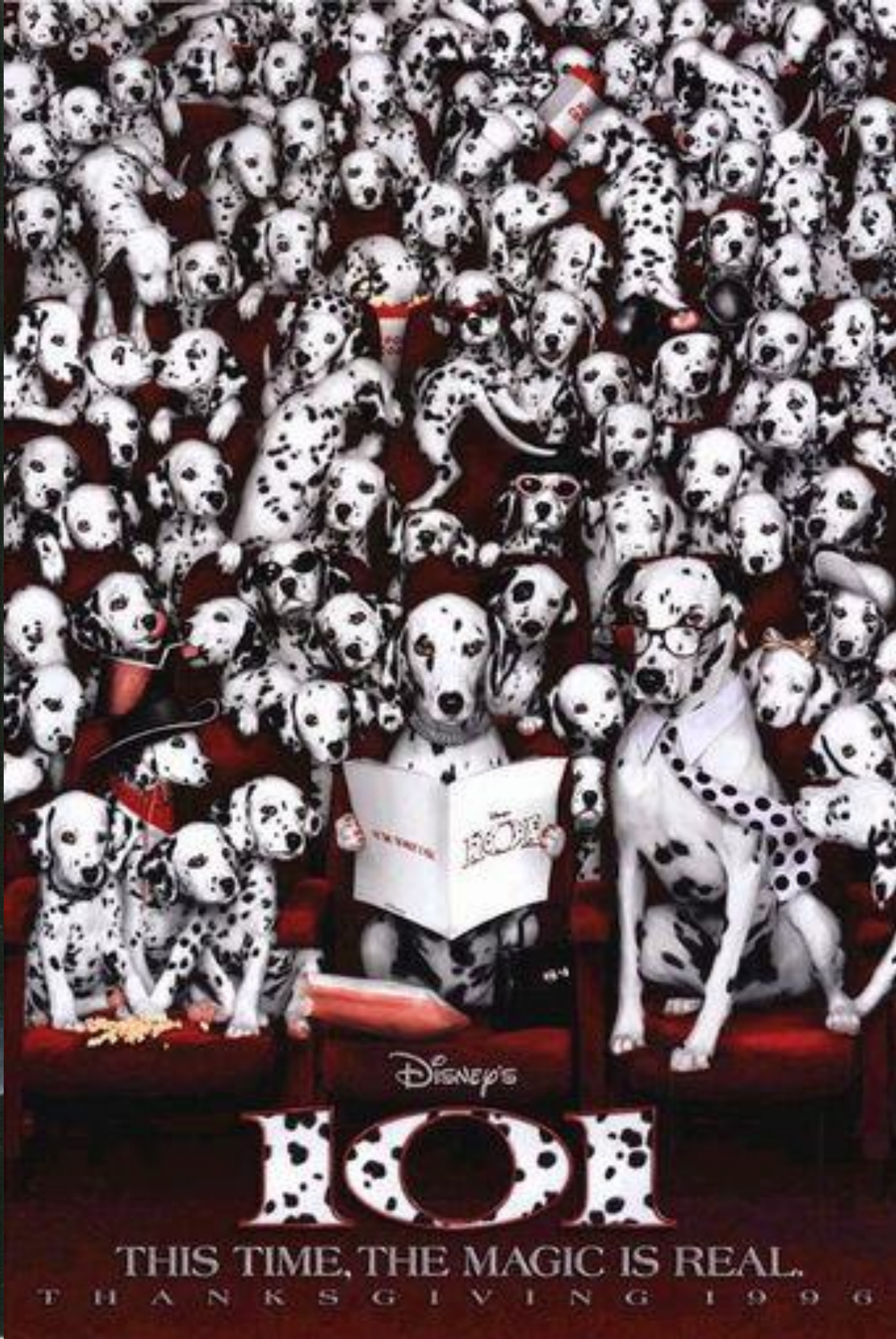
Quiz

- Computer graphics is used for:
  - ______________________________
  - ______________________________
  - ______________________________
  - ______________________________
  - ______________________________

Computer graphics is used for

1. Entertainment

Doom III

FROM THE CREATORS OF INDEPENDENCE DAY
GODZILLA
SIZE DOES MATTER.
MAY 20

DISNEY'S
101
THIS TIME, THE MAGIC IS REAL.
THANKSGIVING 1996

http://www.jazmedia.com.au

Computer graphics is used for

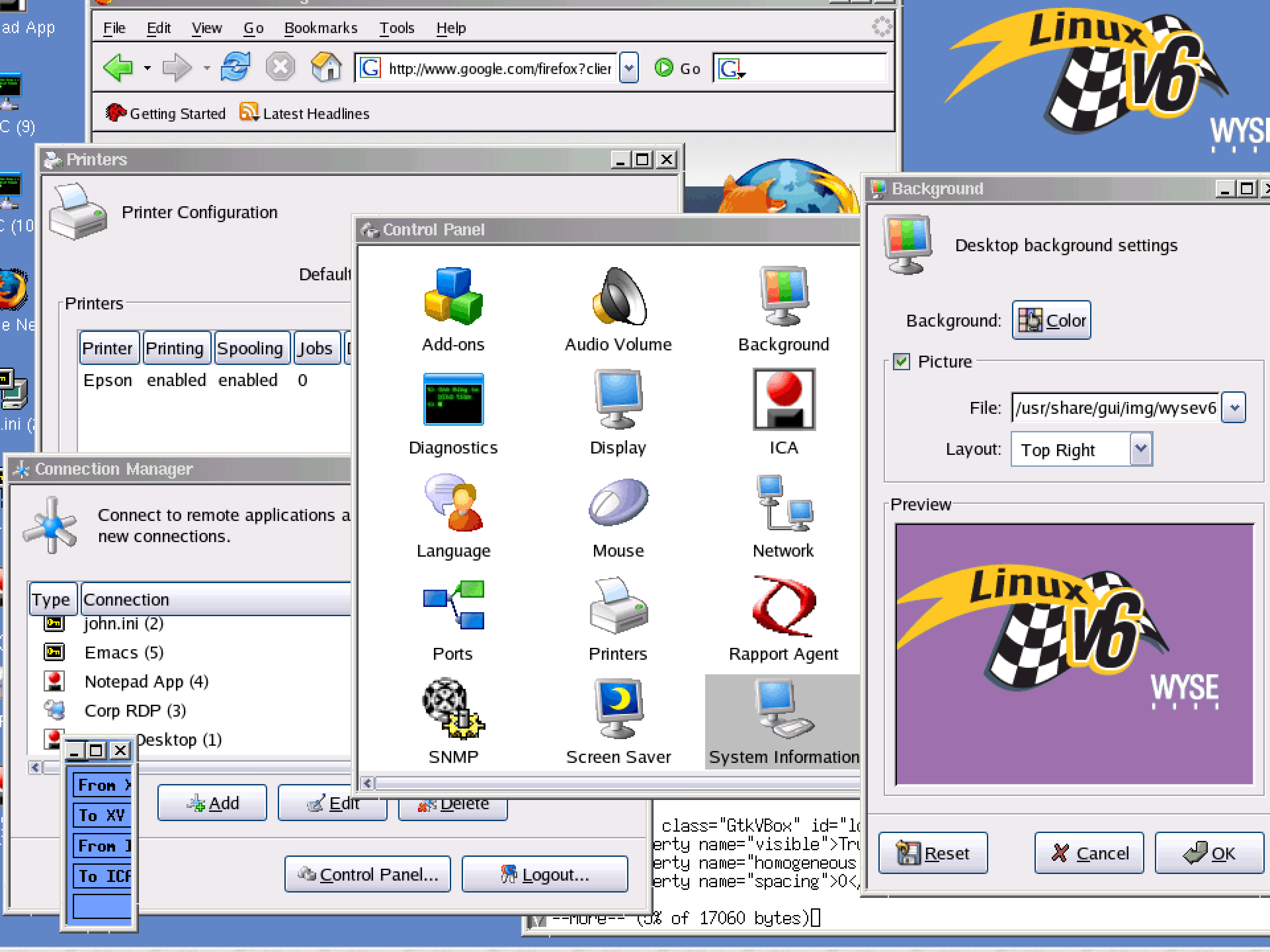
2. Art

http://www.areaticino.com/3d.asp

ART IS HARD

Computer graphics is used for

3. User interfaces



http://www.google.com/firefox?clier

Printer Configuration

Default

Printers

Printer	Printing	Spooling	Jobs
Epson	enabled	enabled	0

Connect to remote applications a
new connections.

Type	Connection
	john.ini (2)
	Emacs (5)
	Notepad App (4)
	Corp RDP (3)
	Desktop (1)

Desktop background settings

Background: Color

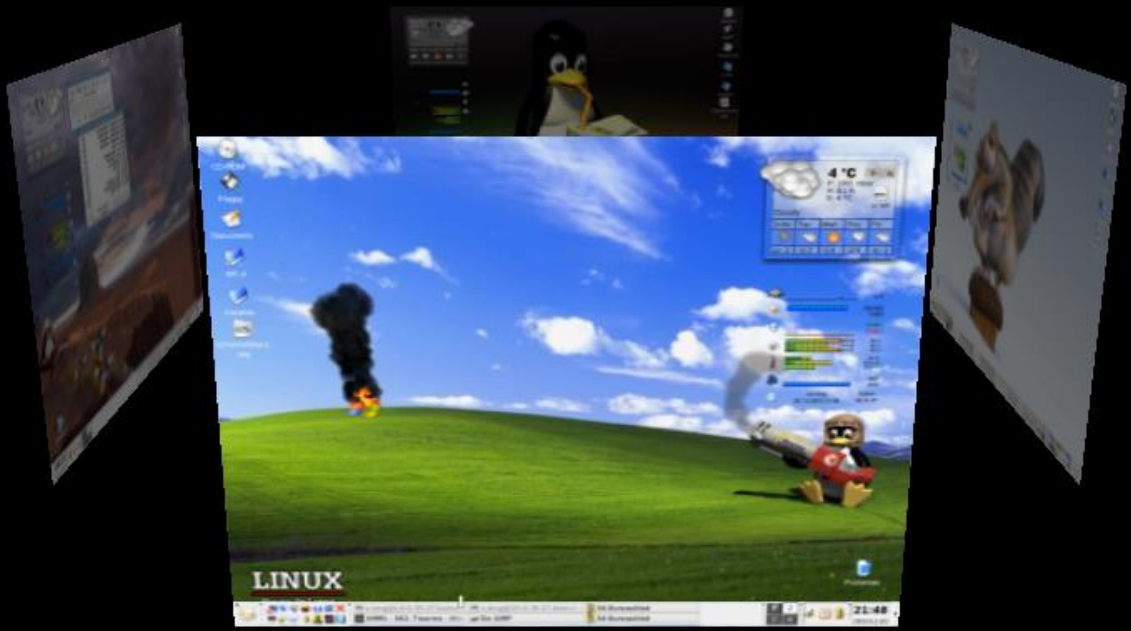
✓ Picture

File: /usr/share/gui/img/wysev6

Layout: Top Right

```
 class="GtkVBox" id="l
erty name="visible">Tr
erty name="homogeneous
erty name="spacing">0<
--More--(3% of 17060 bytes)
```

2

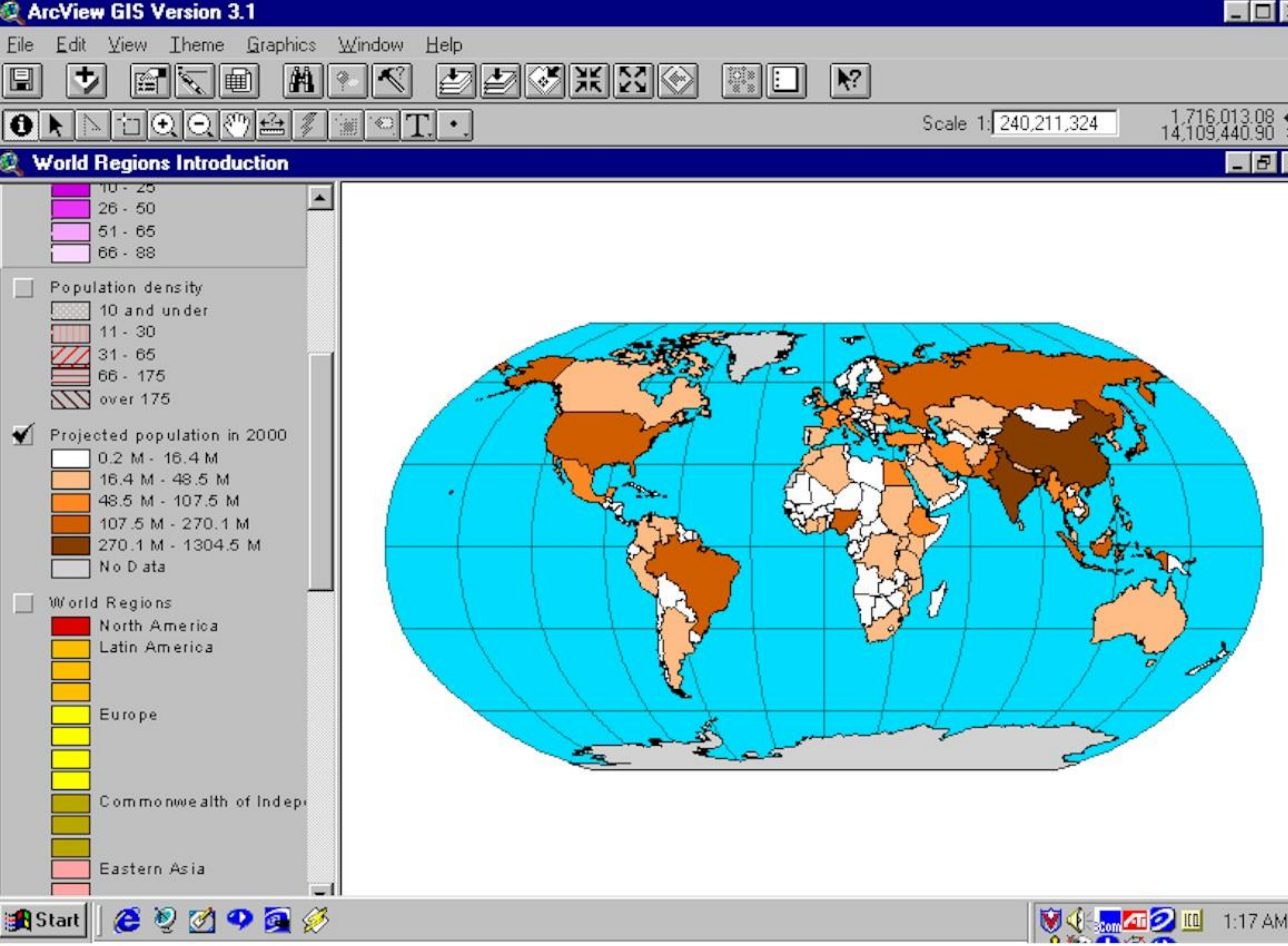


http://www.emezeta.com/articulos/3d-desktop

SynFace Project

Computer graphics is used for

4. Scientific visualization

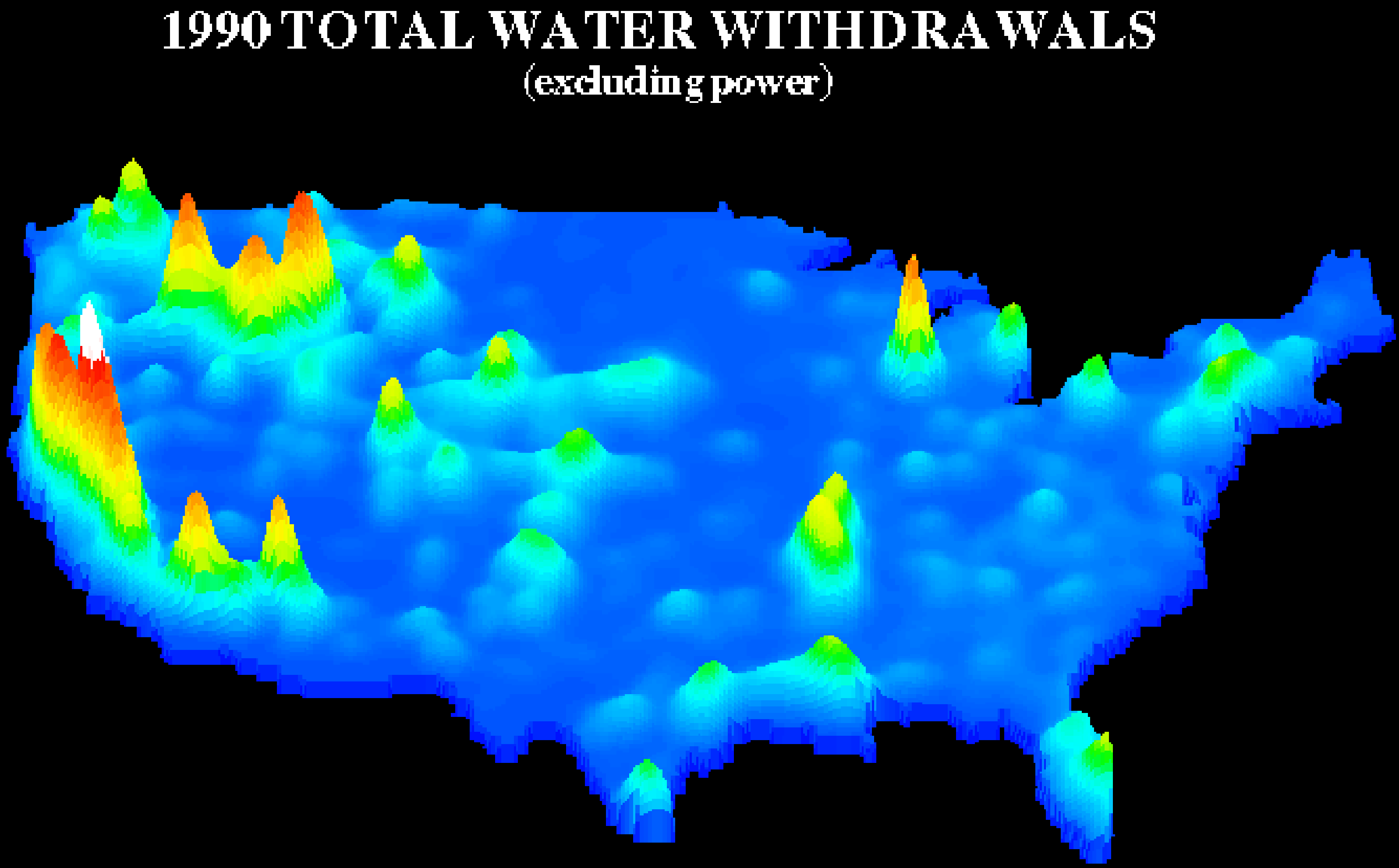


10 - 25
26 - 50
51 - 65
66 - 88

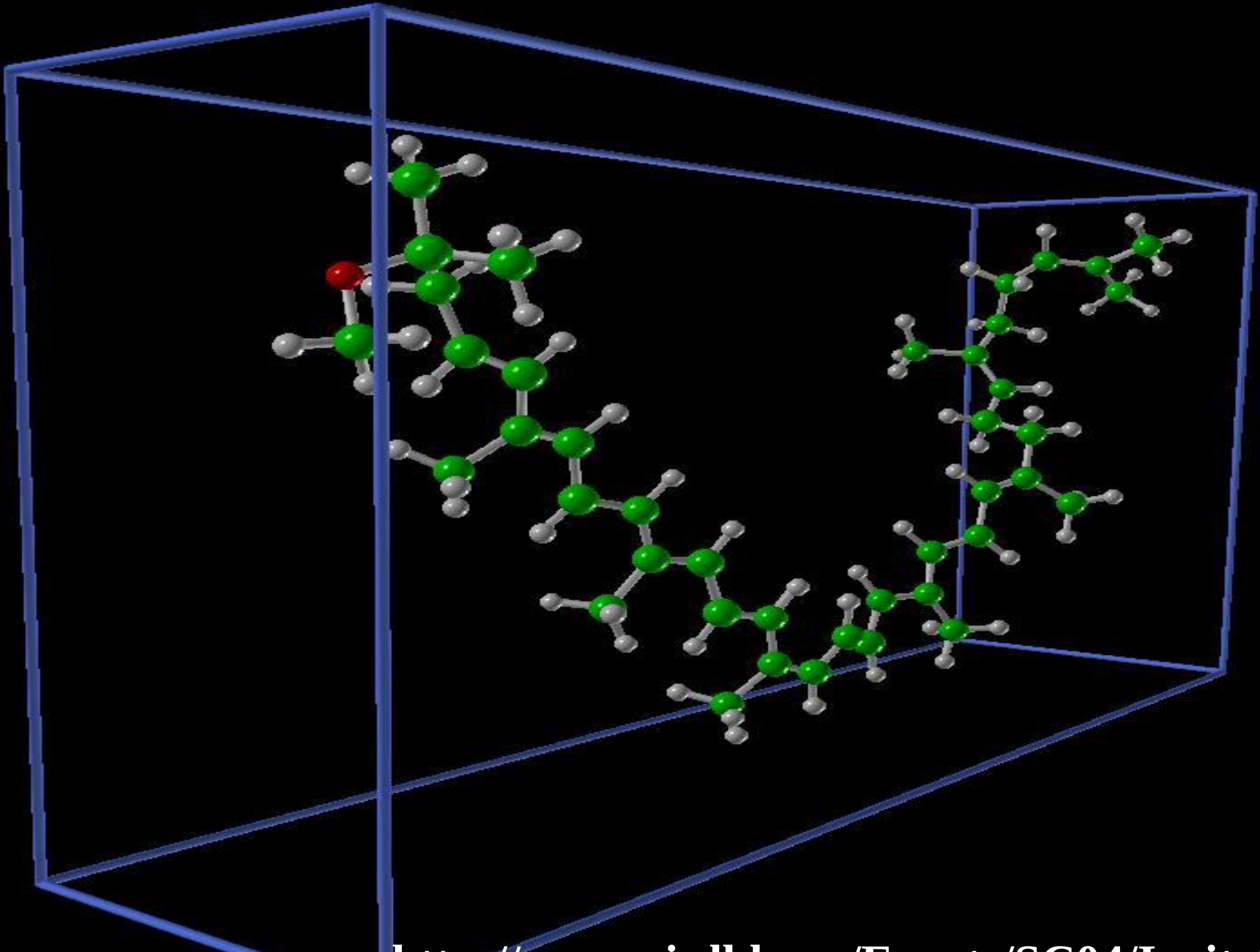
Population density
- 10 and under
- 11 - 30
- 31 - 65
- 66 - 175
- over 175

✓ Projected population in 2000
- 0.2 M - 16.4 M
- 16.4 M - 48.5 M
- 48.5 M - 107.5 M
- 107.5 M - 270.1 M
- 270.1 M - 1304.5 M
- No Data

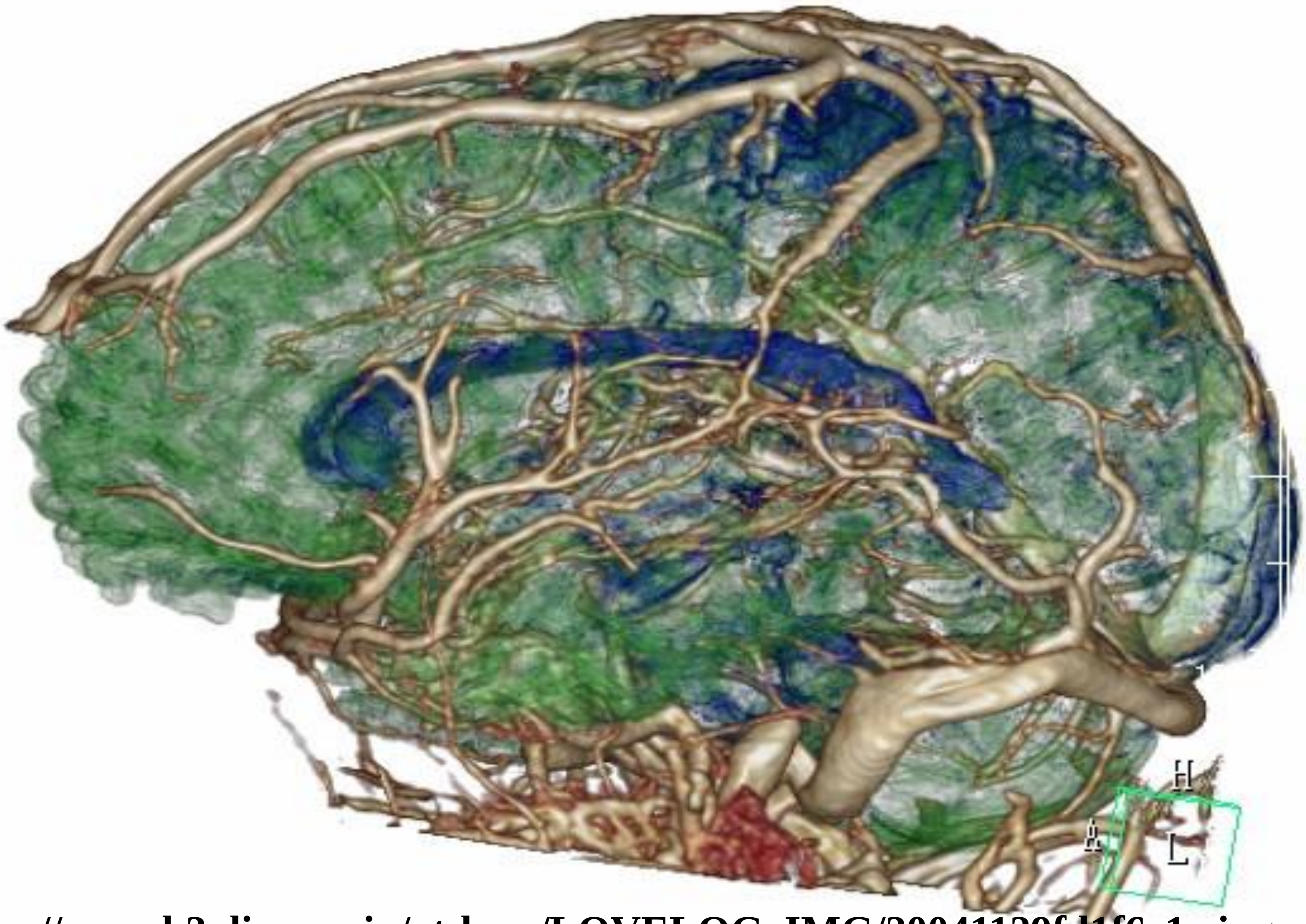
World Regions
- North America
- Latin America
- Europe
- Commonwealth of Indep
- Eastern Asia



http://water.usgs.gov/watuse/graphics/wuto.fact.3d.gif



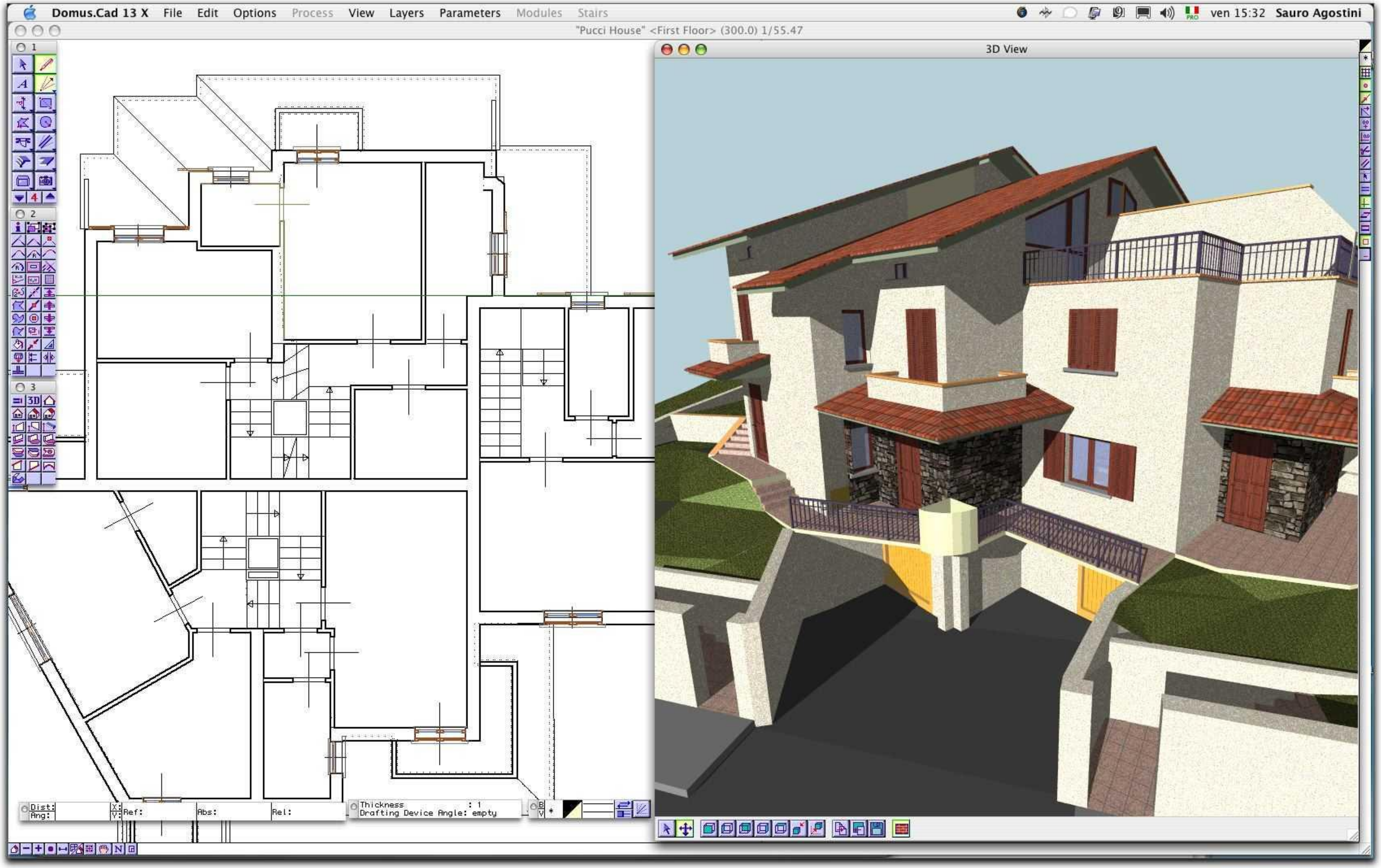
http://www-vis.lbl.gov/Events/SC04/Incite1/



http://www.k2.dion.ne.jp/~t-kmr/LOVELOG_IMG/20041129fd1f6c1e.jpg

Computer graphics is used for

5. CAD


Domus.Cad 13 X File Edit Options Process View Layers Parameters Modules Stairs
ven 15:32 Sauro Agostini
"Pucci House" <First Floor> (300.0) 1/55.47
3D View
Dist:
Ang:
Ref:
Abs:
Rel:
Thickness : 1
Drafting Device Angle: empty

Domus.Cad 13

http://www.leon-arts.tk/

Computer graphics is used for

6. Simulations

http://www.weaverling.org/atc/sim/

© Lance Cpl. Natasha S. Green

Computer graphics is used for

1. Entertainment
2. Art
3. User interfaces
4. Scientific visualization
5. Design
6. Simulation

What is computer graphics

Computer graphics deals with the problem of

Generating images

This course is **not** about

- Image processing & computer vision
- Game development
- 3D modeling & design
- Physics and simulation

Main topics

- **Modeling**
  - How to represent objects?
  - How to construct those representations?
- **Rendering**
  - How to render objects as 2D images?
- **Animation**
  - How to make objects move?

Main topics

- **Modeling**
  - How to represent objects?
  - How to construct those representations?
- **Rendering**
  - How to render objects as 2D images?
- **Animation**
  - How to make objects move?

Modeling

- **How to represent objects?**
  - Geometry (shape of an object)
  - Photometry (color, light effects, reflections, refractions)
- **How to construct those representations?**
  - ________
  - ________
  - ________
  - ________

Modeling

- **How to represent objects?**
  - Geometry (shape of an object)
  - Photometry (color, light effects, reflections, refractions)
- **How to construct those representations?**
  - Describe manually
  - Create interactively
  - Scan
  - Program ("let it grow itself")

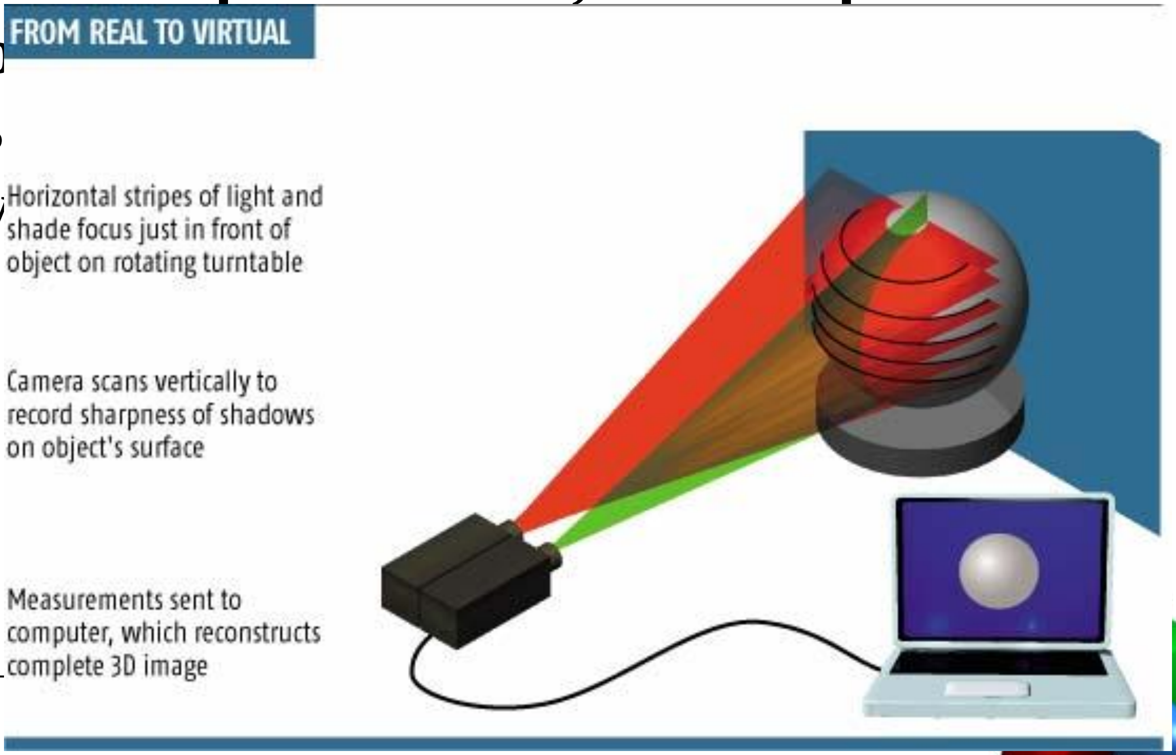
Cheap 3D scanner nearing the desktop

10:01 06 March 2004

Exclusive from New Scientist Print Edition

Will Knight

Ever fancied taking your favourite possessions with you into the virtual world? Spiral Scratch, a start-up company in Liverpoo device that generates representation of any


FROM REAL TO VIRTUAL
Horizontal stripes of light and shade focus just in front of object on rotating turntable
Camera scans vertically to record sharpness of shadows on object's surface
Measurements sent to computer, which reconstructs complete 3D image

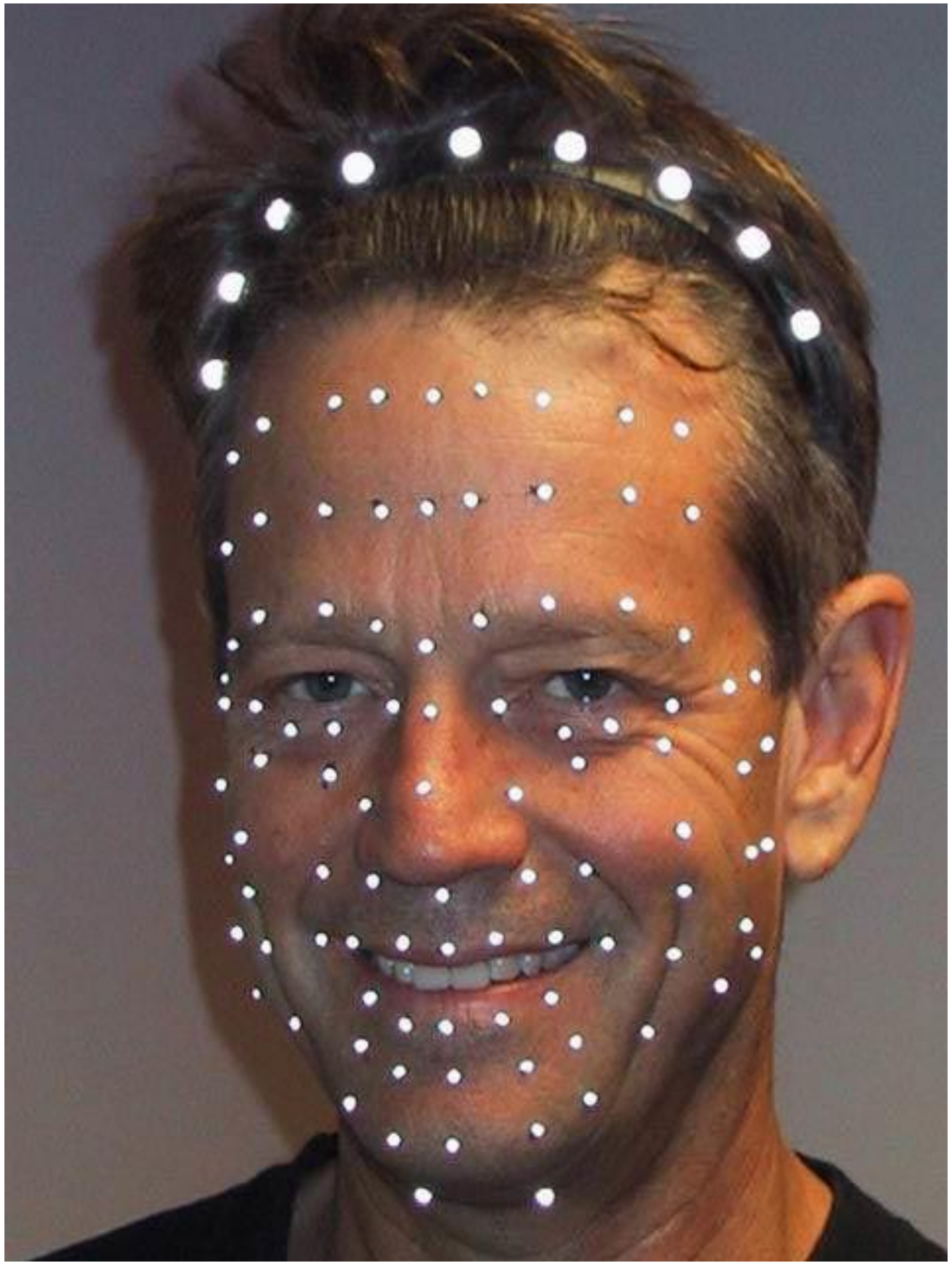
Main topics

- **Modeling**
  - How to represent objects?
  - How to construct those representations?
- **Rendering**
  - How to render objects as 2D images?
- **Animation**
  - How to make objects move?

Animation

- **How to represent movement?**
  - Sequence of frames
  - Trajectories as curves
  - Physical or other laws
- **How to construct representations?**
  - Create manually or interactively
  - Scan (*motion capture*)
  - Program (physic simulations, *A-life*)

01:00:45:29
ND 08:56:18:11
ST.2
SC26A -AA6 ,TK09 A



© Brian Carpenter

Rendering

- **How to represent an image**
  - ________________
  - ________________

Rendering

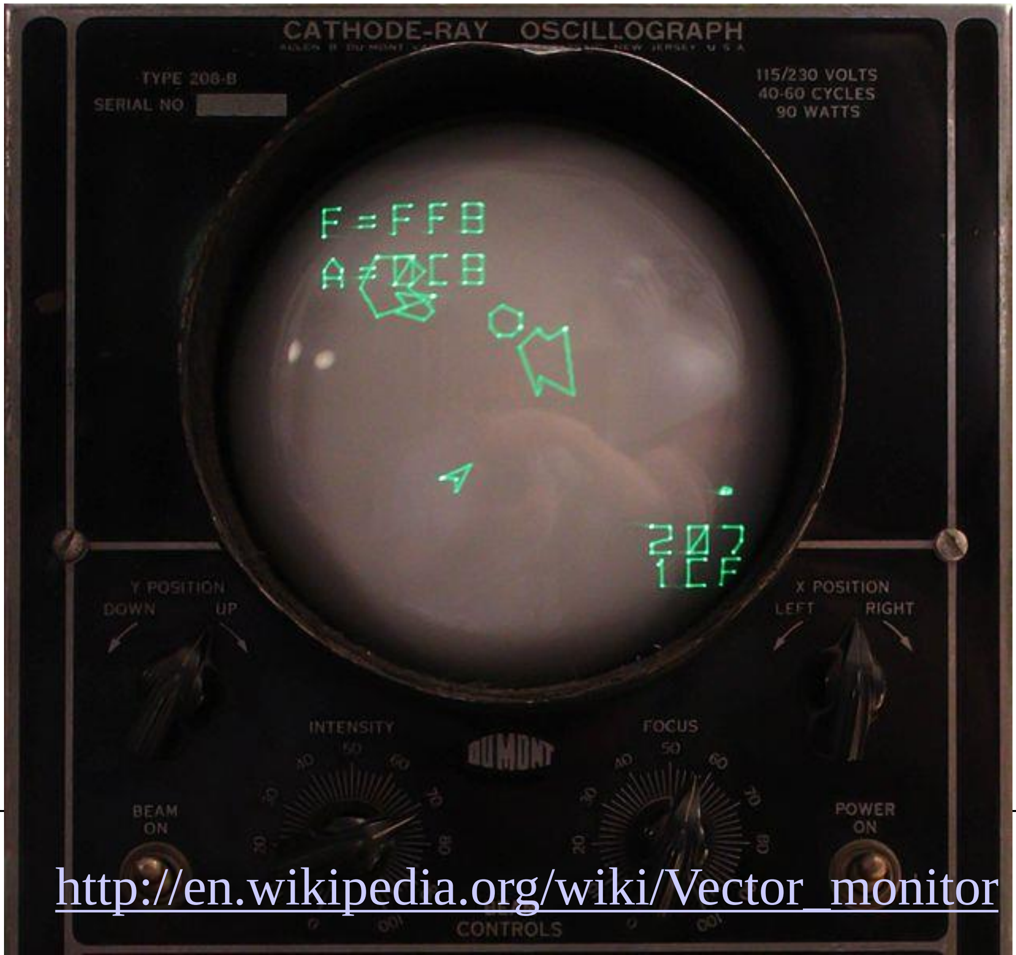
- **How to represent an image**
  - **Raster graphics**: Image is a distribution of light on a plate. Represent it as an array of sampled *pixels* p[x,y].
  - **Vector graphics**: Image is a combination of simple primitives (points, lines, shapes).

Raster vs. Vector

- The first computer displays were inherently ____________________

Raster vs. Vector

- The first computer displays were inherently vector-based.



http://en.wikipedia.org/wiki/Vector_monitor

Raster vs. Vector

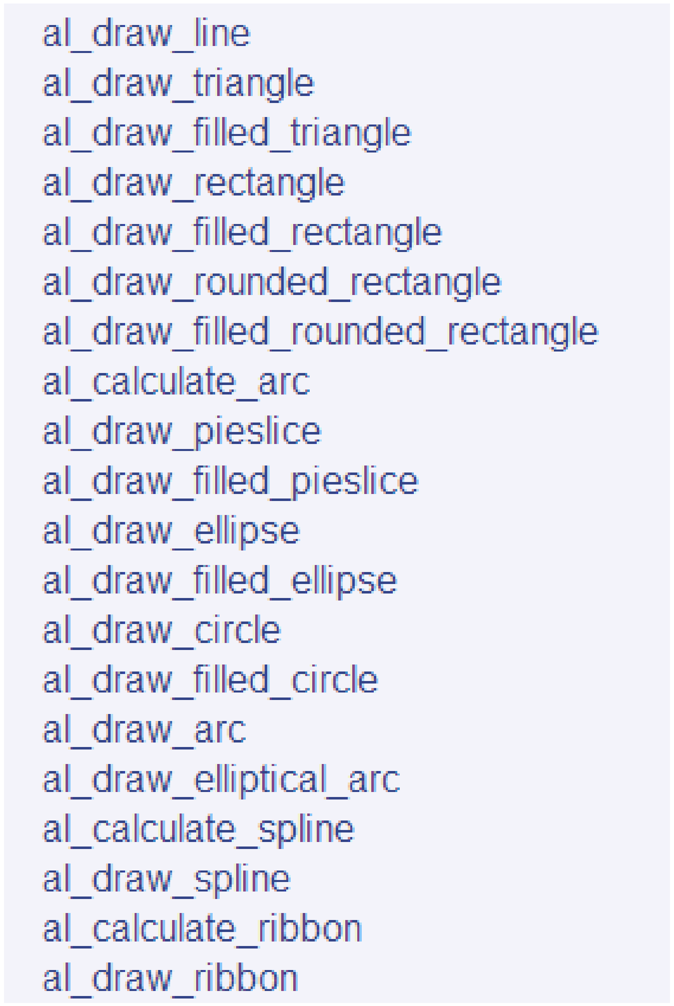
- Nowadays, all monitors are raster-based.
- Are there any vector output devices in use today?

Raster vs. Vector

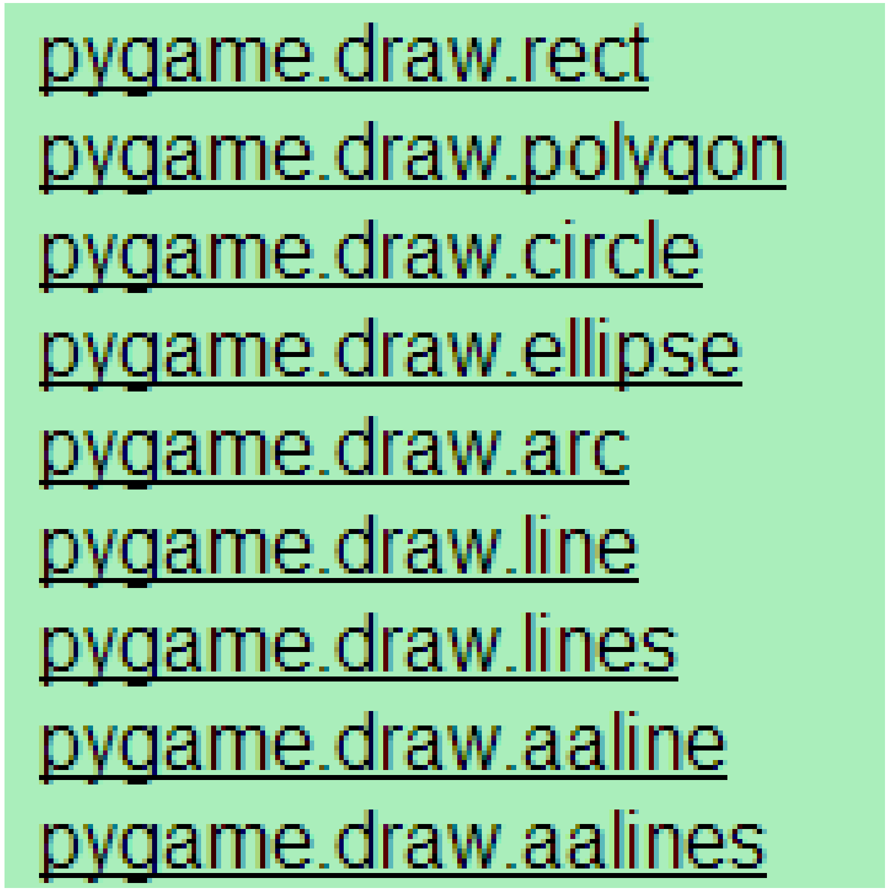
- Nowadays, all monitors are raster-based.
- The process of rendering an image to a raster-based device is called **rasterization**.

Raster vs. Vector

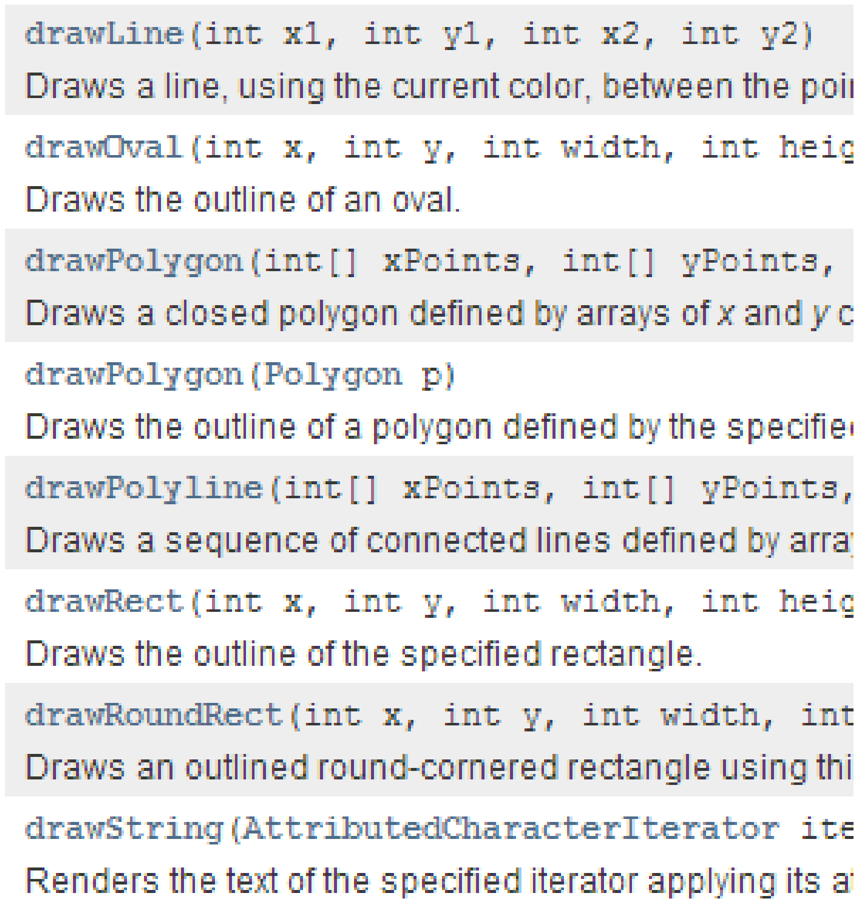
- Nowadays, all monitors are raster-based.
- The process of rendering an image to a raster-based device is called **rasterization**.
- A set of algorithms for rasterization of simple 2D primitives (e.g. lines, polygons, curves) forms the essence of **2D graphics**.



al_draw_line
al_draw_triangle
al_draw_filled_triangle
al_draw_rectangle
al_draw_filled_rectangle
al_draw_rounded_rectangle
al_draw_filled_rounded_rectangle
al_calculate_arc
al_draw_pieslice
al_draw_filled_pieslice
al_draw_ellipse
al_draw_filled_ellipse
al_draw_circle
al_draw_filled_circle
al_draw_arc
al_draw_elliptical_arc
al_calculate_spline
al_draw_spline
al_calculate_ribbon
al_draw_ribbon



pygame.draw.rect
pygame.draw.polygon
pygame.draw.circle
pygame.draw.ellipse
pygame.draw.arc
pygame.draw.line
pygame.draw.lines
pygame.draw.aaline
pygame.draw.aalines



`drawLine(int x1, int y1, int x2, int y2)`
Draws a line, using the current color, between the poi

`drawOval(int x, int y, int width, int heig`
Draws the outline of an oval.

`drawPolygon(int[] xPoints, int[] yPoints,`
Draws a closed polygon defined by arrays of *x* and *y* c

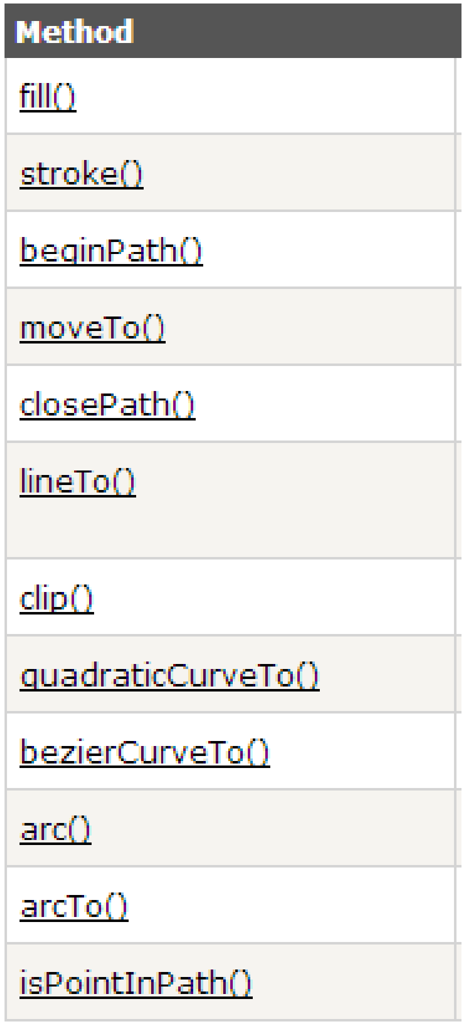
`drawPolygon(Polygon p)`
Draws the outline of a polygon defined by the specifie

`drawPolyline(int[] xPoints, int[] yPoints,`
Draws a sequence of connected lines defined by arra

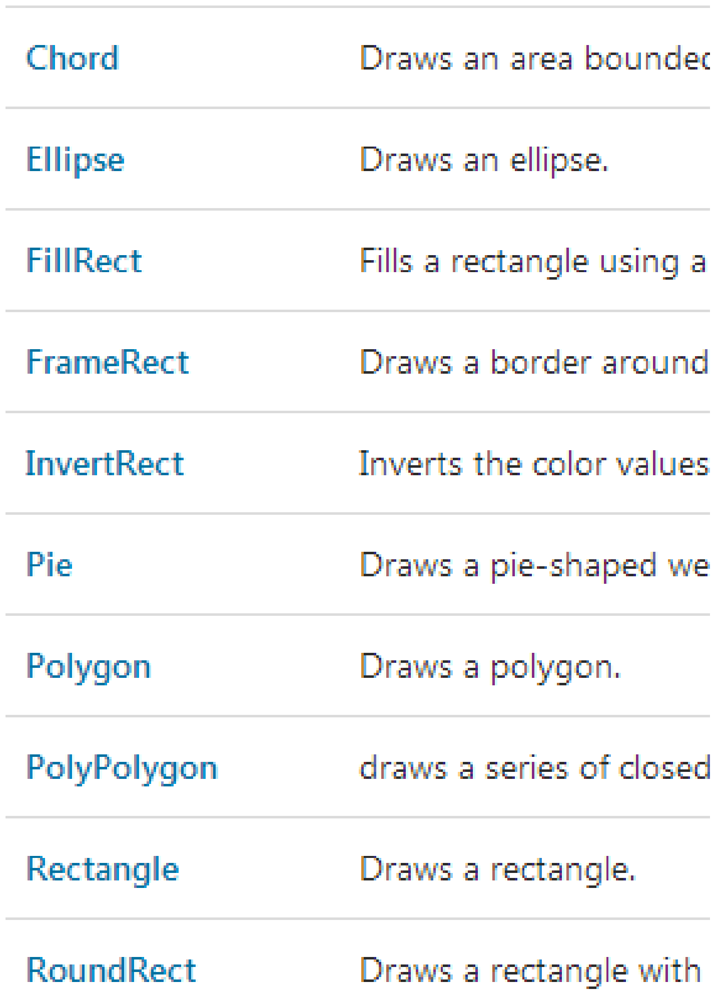
`drawRect(int x, int y, int width, int heig`
Draws the outline of the specified rectangle.

`drawRoundRect(int x, int y, int width, int`
Draws an outlined round-cornered rectangle using thi

`drawString(AttributedCharacterIterator ite`
Renders the text of the specified iterator applying its a



Method
fill()
stroke()
beginPath()
moveTo()
closePath()
lineTo()
clip()
quadraticCurveTo()
bezierCurveTo()
arc()
arcTo()
isPointInPath()



Chord	Draws an area boundec
Ellipse	Draws an ellipse.
FillRect	Fills a rectangle using a
FrameRect	Draws a border around
InvertRect	Inverts the color values
Pie	Draws a pie-shaped we
Polygon	Draws a polygon.
PolyPolygon	draws a series of closed
Rectangle	Draws a rectangle.
RoundRect	Draws a rectangle with

3D graphics

- How to rasterize 3D objects?
  - Project to the “camera plane”, come up with a reasonable coloring, and reduce the task to 2D graphics.
    - Classical 3D rasterization pipeline
  - Simulate light
    - *Raytracing*, *Radiosity*, *MC-lighting*, …

3D graphics

- How to rasterize 3D objects?
  - Project to the "camera plane", come up with a reasonable coloring, and reduce the task to 2D graphics.
    - Classical 3D rasterization pipeline
  - Simulate light
    - *Raytracing, Radiosity, MC-lighting,…*

3D graphics

```
...
draw_monster(bad_guy1);
...
```

my_code.cpp

3D graphics

```
...
draw_monster(bad_guy1);
...
```

```
...
draw_monster_head(..);
...
```

```
...
draw_triangle(..);
...
```

3D graphics

```
draw_triangle(..) {
  compute_position_in_3d(..);
  is_it_visible?();
  compute_lighting_and_color(..);
  project_to_screen(..);
  draw_2d_triangle(..);
}
```

```
...
set_pixel(..)
...
```

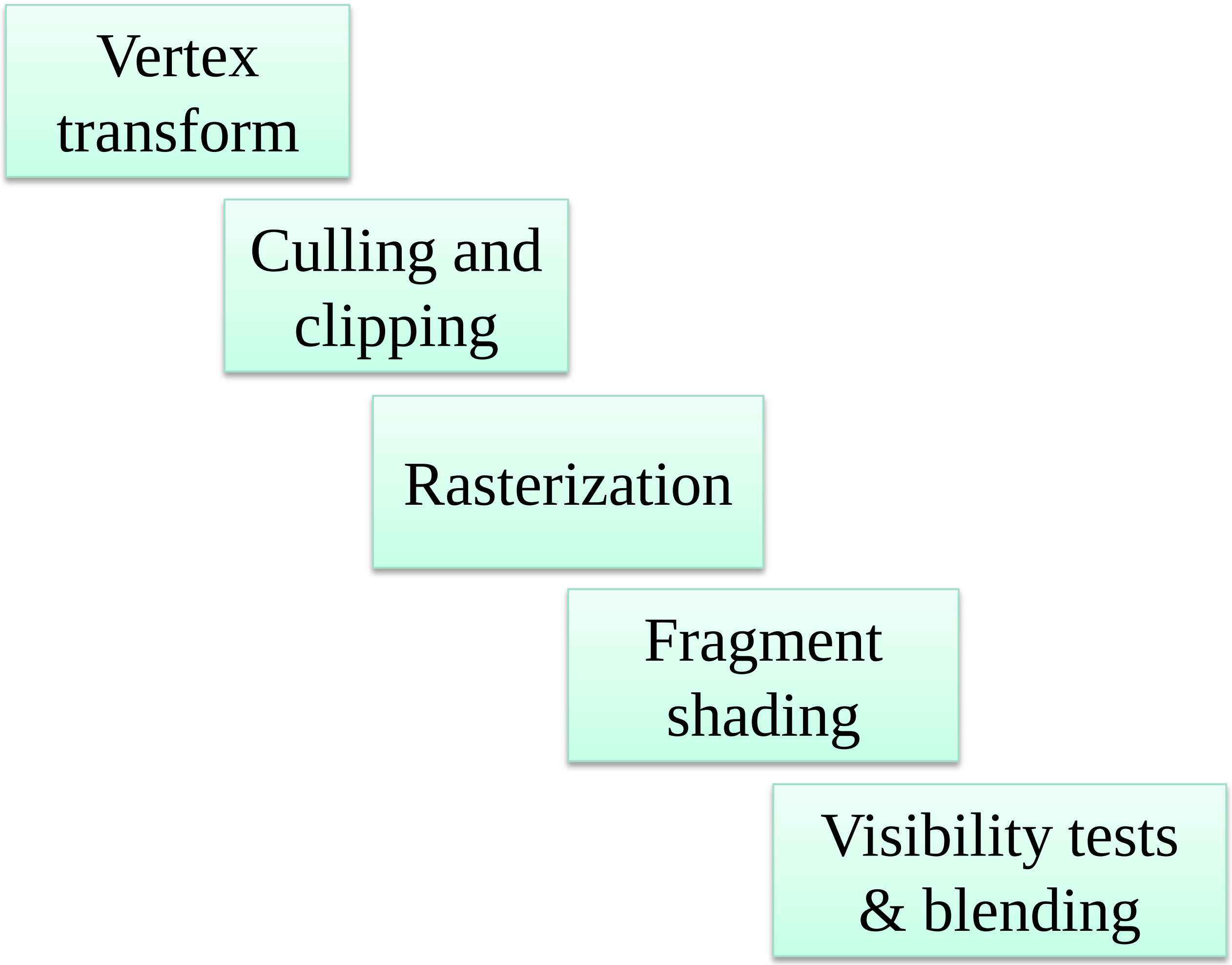
3D graphics

```
draw_triangle(..) {
    compute_position_in_3d(..);
    is_it_visible?();
    compute_lighting_and_color(..);
    project_to_screen(..);
    draw_2d_triangle(..);
}
```

```
...
set_pixel(..)
...
```

Hardware (GPU)

Standard Graphics Pipeline



Questions?