Graphics Concepts

CSCU9N6

Introduction

The following material provides a brief introduction to some standard graphics concepts. For more detailed information, see DGJ, Chapter 2, p23.

- Display Modes
- Full Screen Graphics
  - Screens
  - Pixel Colour & Bit Depth
- Image types and formats
- Anti-aliasing

Display Modes

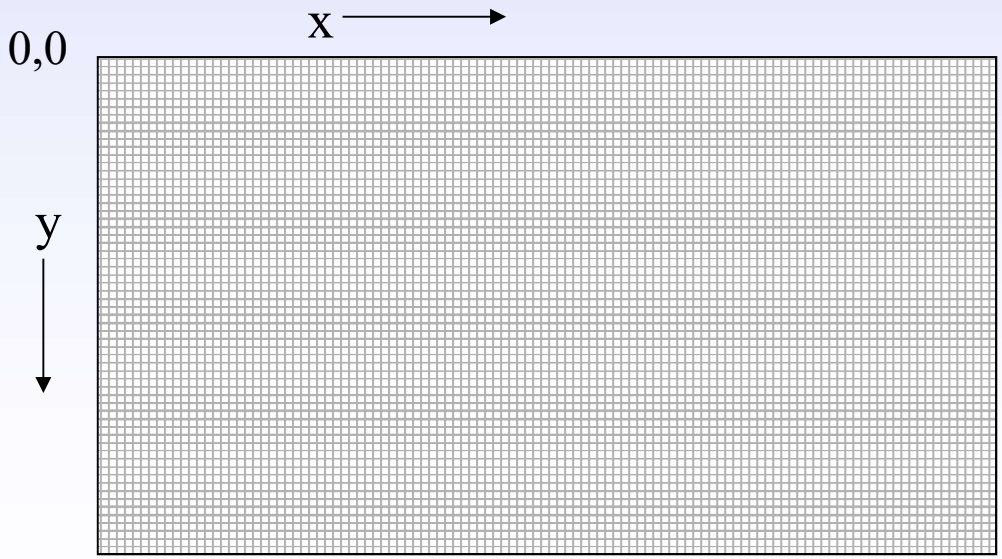
Display Modes

- Applet / Plug In
  - Applications that run within a browser
  - Applet security features impose restrictions
- Windowed
  - A standard windowed application
  - Restrictions set by user privileges
  - The application draws within its own window but does not worry about the 'world outside the window'
- Full Screen
  - Similar to the above but responsible for drawing everything on the screen and handling keyboard/mouse events at a relatively low level.
  - More immersive since the entire display is at your command but requires more work since you have to manage everything.

Screen Resolution & Colours

Screen

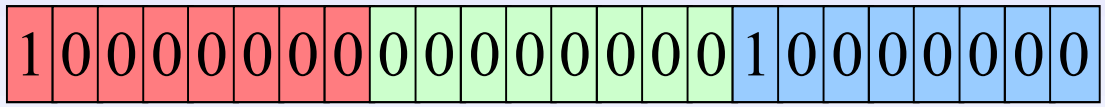
- Defined in horizontal & vertical pixels (e.g. 1024x768)
  - In standard Java graphics, the origin 0,0 is at the top left and co-ordinates are labelled x then y.
  - Some graphics co-ordinate systems allow you to specify the location of the origin and co-ordinate system, e.g. 0,0 could be moved to the centre
  - Windows supports the concept of a virtual screen which is mapped (in many different ways) to the actual screen

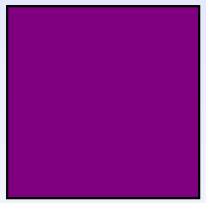


Bit Depth & Colour

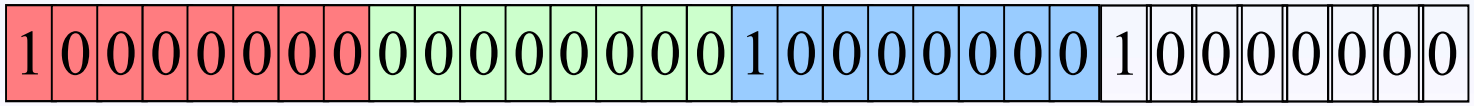
Each pixel has a bit-depth, with each bit-depth setting enabling different levels of colour or grey scale

- Usually $2^{\text{bit-depth}}$ colours or shades of grey
- Colours are built up from a red, green and blue components
- 24 bit colour uses 8 bits for each of the red, green and blue components respectively
- 32 bit colour can add an 8 bit transparency level

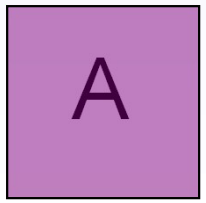




24 bits – 8 Red, 8 Green, 8 Blue (RGB)



32 bits – 8 Red, 8 Green, 8 Blue, 8 Alpha (RGB)

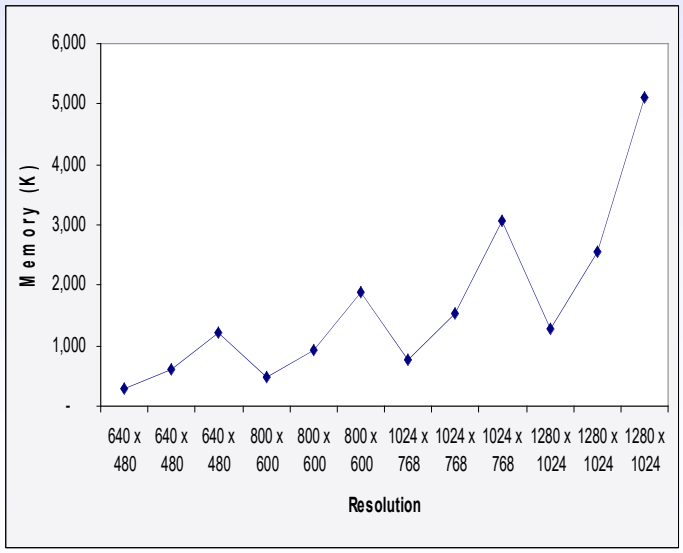


Screen Resolution & Bit Depth

Games will frequently let you define the screen resolution and bit depth you would like to play at.

- This is often limited by the processing power of your computer or more usually, the graphics card
- The data processing throughput rises quite dramatically
- Your games programs should run at the most common resolutions

Horizontal	Vertical	Bit-Depth	Bytes	K
640	480	8	307200	300
640	480	16	614400	600
640	480	32	1228800	1,200
800	600	8	480000	469
800	600	16	960000	938
800	600	32	1920000	1,875
1024	768	8	786432	768
1024	768	16	1572864	1,536
1024	768	32	3145728	3,072
1280	1024	8	1310720	1,280
1280	1024	16	2621440	2,560
1280	1024	32	5242880	5,120



24 & 8 Bit Colour / Grey Scale

24

8

4 & 1 Bit Colour / Grey Scale

4

1

Displays

- Cathode Ray Tubes (CRT) use an electron beam to paint the screen and can therefore paint at whatever resolution is required
- Liquid Crystal Display (LCD) screens are lit using individual transistors per pixel
  - This results in a native 'resolution' for the screen e.g. 1280x1024. An LCD cannot display higher than the native resolution and lower resolutions will look blurred.
  - Make sure your game works at common resolutions since a user will not be happy if it looks blurred at their native resolution
  - LCDs are affected by update latency, e.g. 3,5,10 ms, this causes 'ghosting' when screen shows rapid motion
- A graphics adapter will update a display with a given refresh rate e.g. 75Hz or 75 times a second. Low rates (<=60Hz) appear to shimmer or flicker as we begin to perceive the updates

Image Types

Image Types

- Opaque
  - The image does not store any form of transparency or translucency information
  - It is not 'See Through'
- Transparent
  - One or more pixel colours are defined as 'See Through' and will not appear when displayed.
- Translucent
  - Each pixel colour has a defined level at which a background image will show through it - it becomes blended with the background pixel

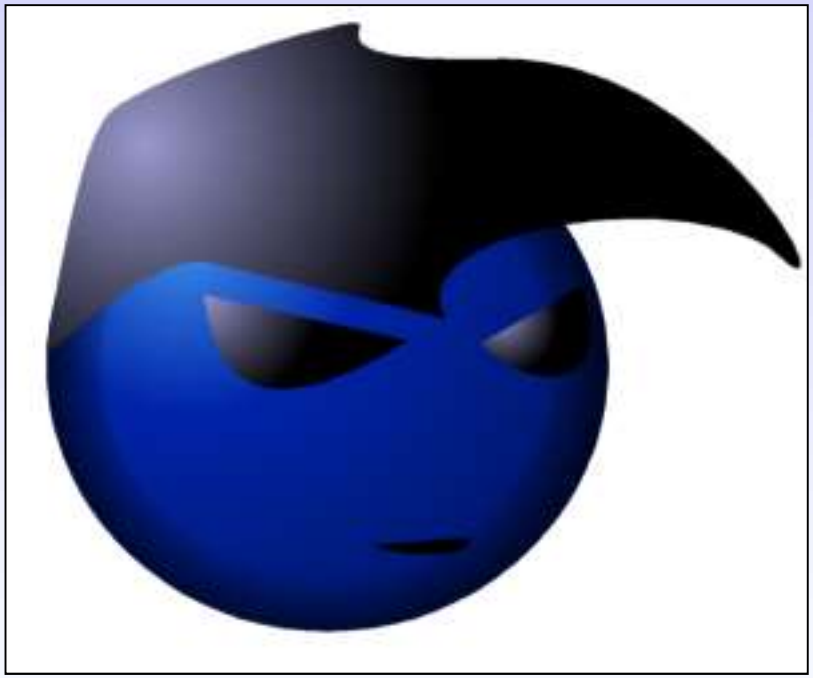
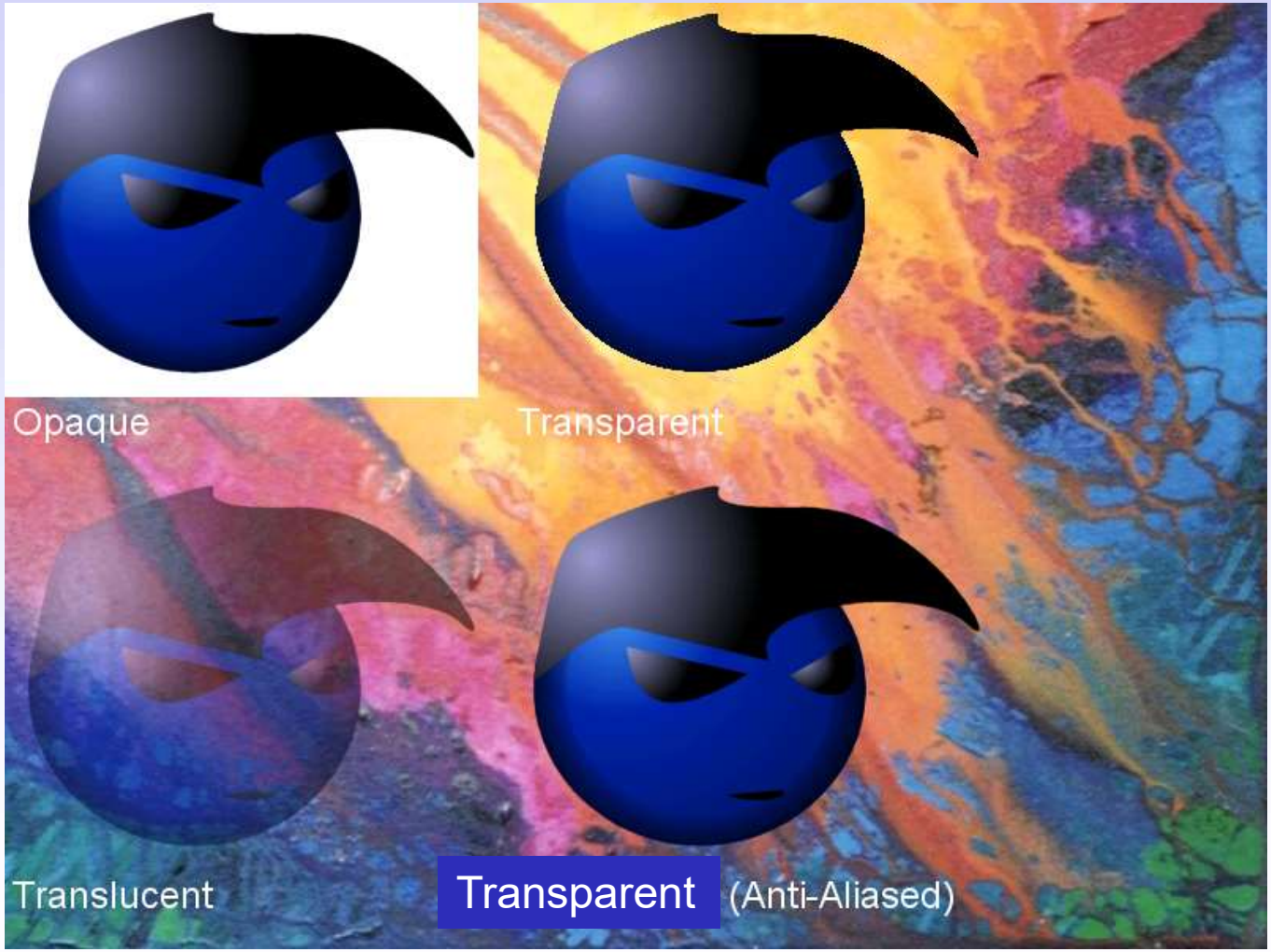


Image Types - Example

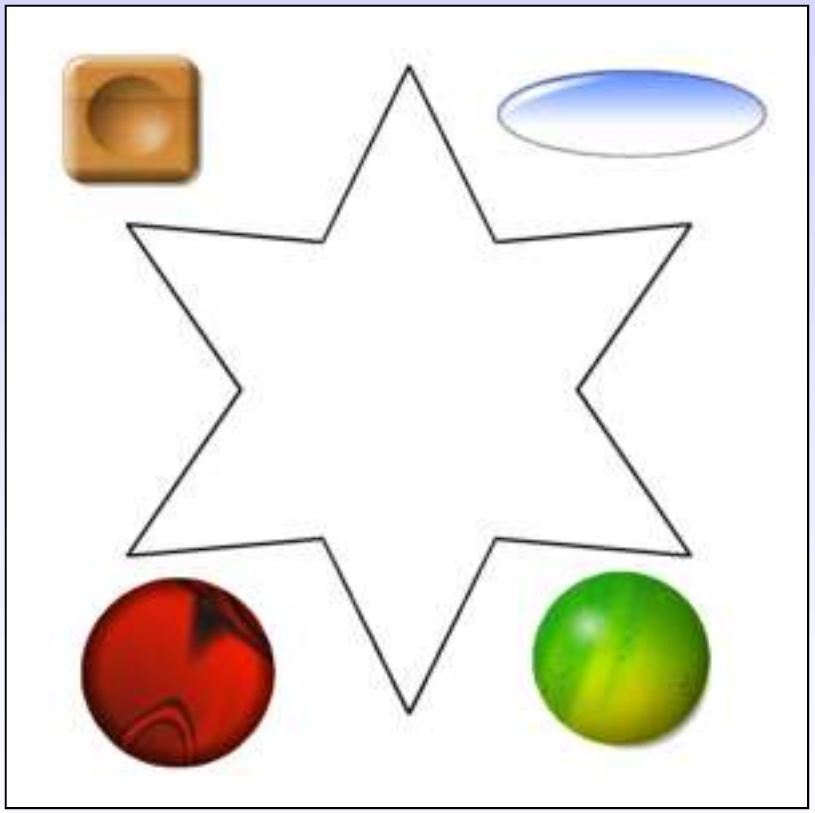


Raster Image Formats

Common Formats in Java

- GIF
  - 8 bit colour resolution (256 colours)
  - Supports transparency but not translucency
  - High compression, not 'lossy'
  - Best suited to graphic design images, not photos
- PNG
  - Newer format, similar to GIF but more capable
  - Variable bit –depth up to 24 bits/pixel
  - Supports transparency and translucency
  - Best suited to graphic design images
- JPEG
  - Opaque 24 bit images
  - High compression factor for photos but 'lossy'

Original (PNG) – 26.2K



JPEG - High Compression -

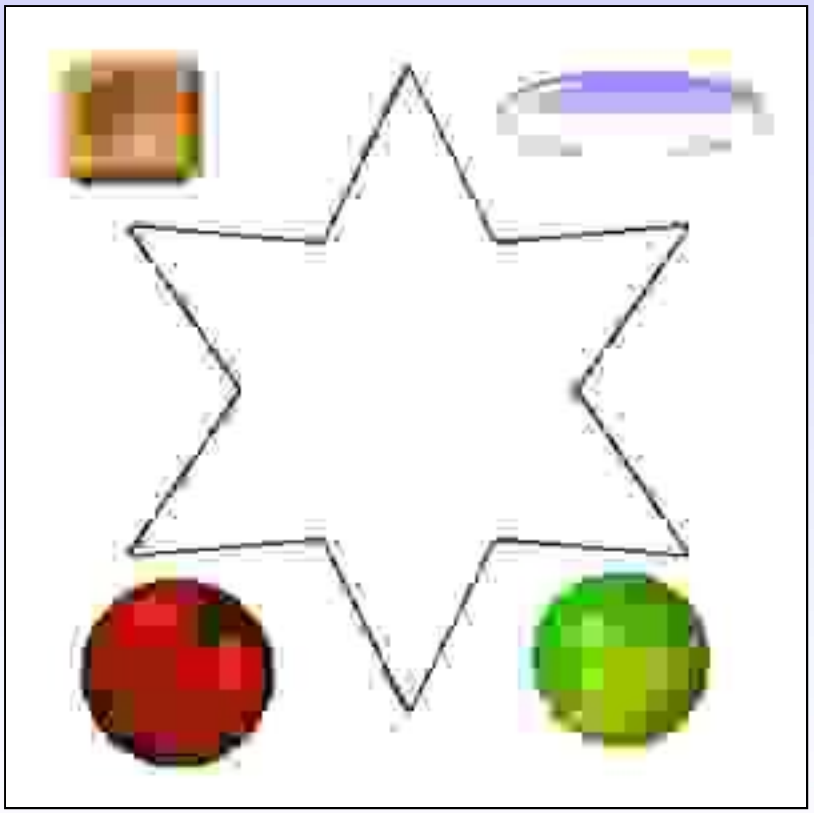


Image Formats - Vector

Instead of storing each point in the image (pixel) as a separate value, vector image formats store an image as a set of drawing commands.

- For example:
  - Draw line from 10,10 to 40,40
  - Draw circle at 30,30 with radius 50
- Vector image formats can be very compact
- They remain precise at all resolutions (scale well)
- An excellent format for a compact representation of line drawn images but not good for encoding photographs
- A common web format is Scalable Vector Graphics (SVG)

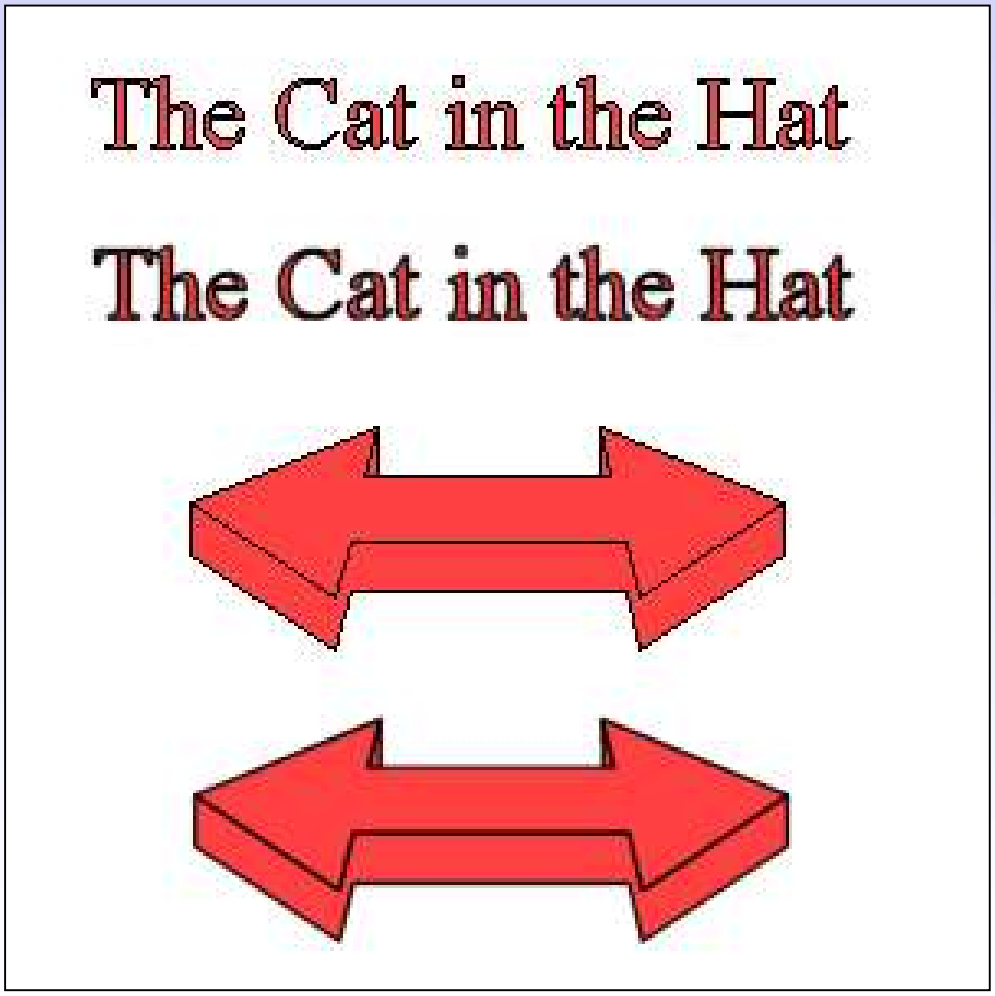
Anti-Aliasing

Anti-aliasing is a technique used to smooth out jagged sections within an image by blurring them with their background.

Most frequently used with text but newer graphics cards also support full-screen anti-aliasing with 3D graphics objects

- This makes objects look smoother and more natural
- Computing this in real time adds a processor overhead
- Basic AA will smooth against adjacent pixels
  - Checking immediate neighbours results in 8 comparisons per pixel.
  - More advanced AA will smooth over an increasing number of neighbouring pixels, e.g, 2 pixels out equals 24 comparisons per pixel
  - 1920 x 1200 x 24 / per frame = 55,296,000
  - 40 frames per second = 55,296,000 x 40 equals 2,211,840,000 comparisons per second...

Anti-Aliasing - Example



Summary

Overview of Standard Graphics Concepts

- Display Modes
- ScreenResolutions
- Bit Depth
- Image Types & Formats
- Anti-Aliasing

Next

- Drawing Graphics in Java