EBU5405
3D Graphics Programming Tools

3D Computer Graphics Software

Introduction to OpenGL

Dr. Marie-Luce Bourguet
(marie-luce.bourguet@qmul.ac.uk)

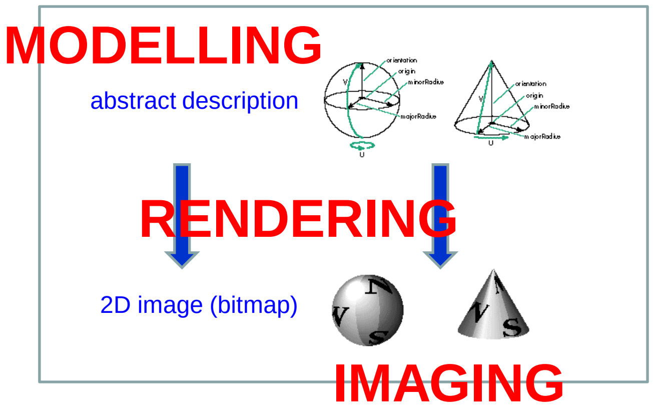
Queen Mary
University of London

Today’s agenda

- Some popular 3D computer graphics software:
  - Blender
  - Unity 3d
  - Autodesk Maya
  - Autodesk 3ds Max (Gmax)
- Introduction to OpenGL

Queen Mary
University of London

How to compare 3D Computer Graphics Software?

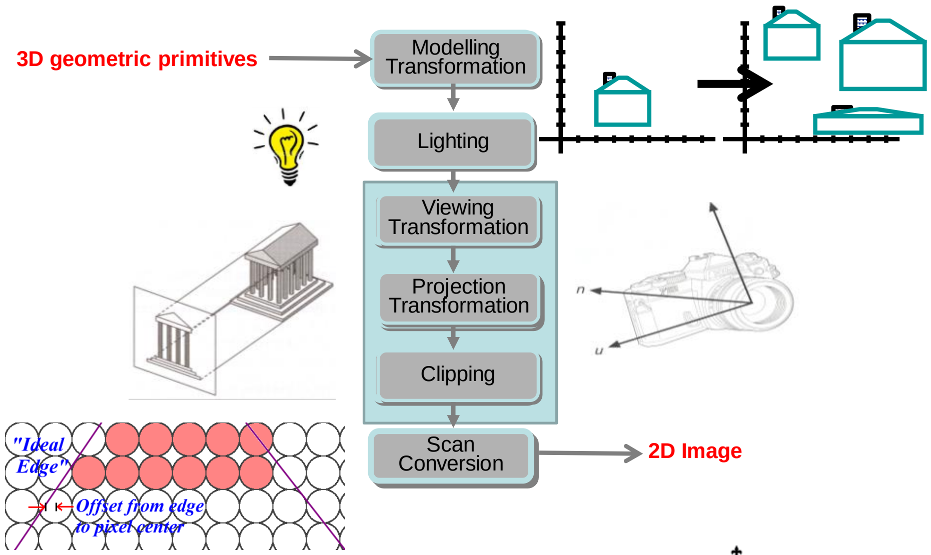


EBU5405

Queen Mary University of London

3

How to compare 3D Computer Graphics Software?



EBU5405

Queen Mary University of London

4

How to compare 3D Computer Graphics Software?

	Modelling	Rendering	Animation	Free
Blender	✓	✓	✓	✓
Unity 3D	✓	✓	✓	?
OpenGL	?	✓	✓	✓
Maya	✓	✓	✓	✗
3ds Max	✓	✓	✓	✗

Blender

- Blender is a public project, made by hundreds of people from around the world, by studios and individual artists, professionals and hobbyists, scientists, students, etc. https://www.blender.org/

- Blender is a free and open source 3D creation suite, that supports modelling and the entirety of the 3D pipeline.

Unity 3D

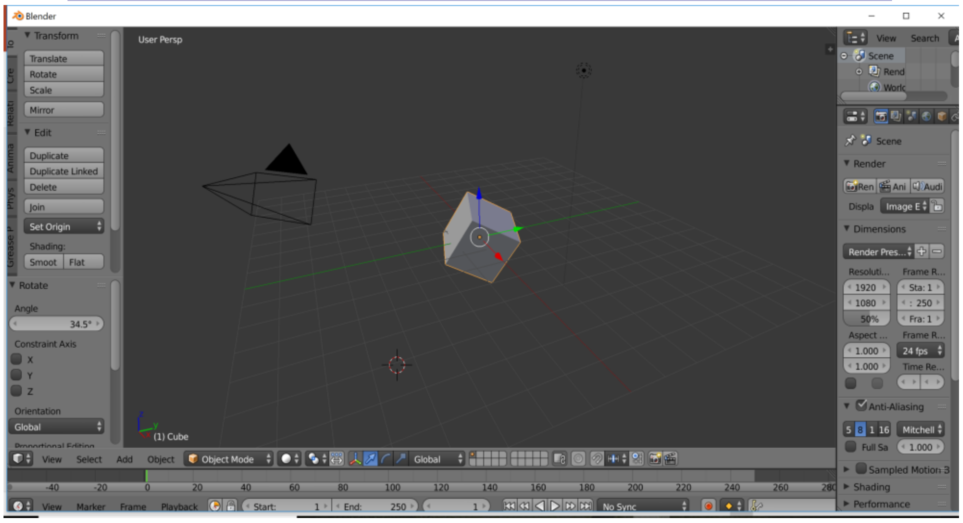
- **Unity** is a cross-platform game engine developed by Unity Technologies, which is primarily used to develop both three-dimensional and two-dimensional video games and simulations for computers, consoles, and mobile devices. https://unity3d.com/

EBU5405

Queen Mary University of London

7

Blender

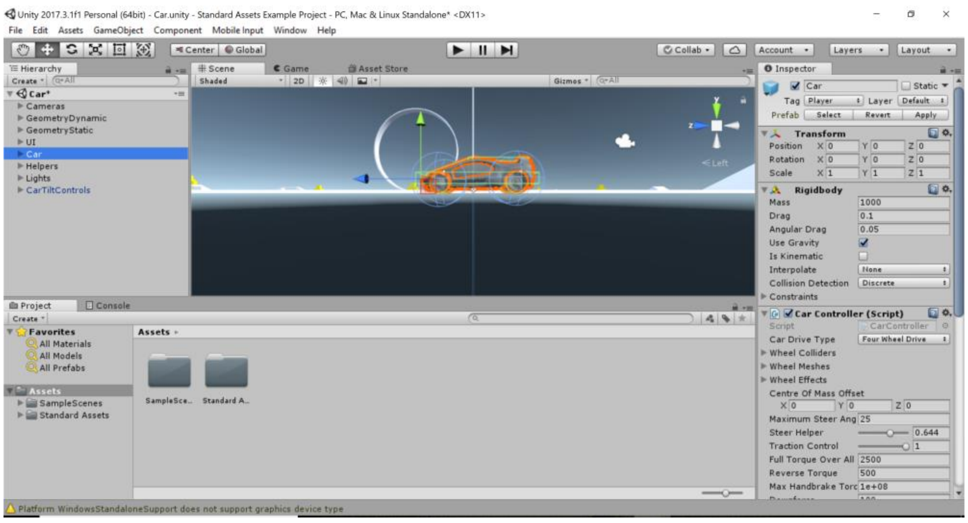


EBU5405

Queen Mary University of London

8

Unity 3D


Unity 2017.3.1f1 Personal (64bit) - Car.unity - Standard Assets Example Project - PC, Mac & Linux Standalone* <DX11>

Inspector

Car	Static
Tag	Player
Layer	Default

Transform

	X	Y	Z
Position	0	0	0
Rotation	0	0	0
Scale	1	1	1

Rigidbody

Mass	1000
Drag	0.1
Angular Drag	0.05
Use Gravity	✓
Is Kinematic	
Interpolate	None
Collision Detection	Discrete

Car Controller (Script)

Script	CarController
Car Drive Type	Four Wheel Drive
Centre Of Mass Offset	X 0 Y 0 Z 0
Maximum Steer Ang	25
Steer Helper	0.644
Traction Control	1
Full Torque Over All	2500
Reverse Torque	500
Max Handbrake Torc	1e+08

Platform WindowsStandaloneSupport does not support graphics device type

EBU5405

OpenGL

OpenGL is a platform-independent API that is

- Easy to use
- Close enough to the hardware to get excellent performance
- Focus on rendering

http://www.opengl.org

What you need:

- OpenGL core library
  - OpenGL32 on Windows
- OpenGL Utility Library (GLU)
  - Uses OpenGL core but avoids having to rewrite code
- OpenGL Utility Toolkit (GLUT)

- a C/C++ compiler

OpenGL

Programming OpenGL in C (with Eclipse):

https://www.ntu.edu.sg/home/ehchua/programming/opengl/HowTo_OpenGL_C.html#zz-1

https://bohr.wlu.ca/cp411/eclipseCDT.php

A very simple C/OpenGL programme

```
#include <GL/glut.h>

void display() {
  glClearColor(0.0, 0.0, 0.0, 0.0);
  glClear(GL_COLOR_BUFFER_BIT);
  glFlush();
}

int main(int argc, char** argv) {
  glutInit(&argc, argv);
  glutCreateWindow("Hello GLUT!");
  glutDisplayFunc(display);
  glutMainLoop();
}
```

GLUT

- OpenGL Utility Toolkit (GLUT)
  - Links with window system (e.g. AGL for Macintosh)
  - Provides functionality common to all window systems
    - Open a window
    - Get input from mouse and keyboard
    - Menus
    - Event-driven
  - Code is portable but GLUT lacks the functionality of a good toolkit for a specific platform
    - E.g. No slide bars

A very simple C/OpenGL programme

```
#include <GL/glut.h>

void display() {
  glClearColor(1.0, 0.0, 0.0, 0.0);
  glClear(GL_COLOR_BUFFER_BIT);
  glFlush();
}

int main(int argc, char** argv) {
  glutInit(&argc, argv);
  glutCreateWindow("Hello GLUT!");
  glutDisplayFunc(display);
  glutMainLoop();
}
```

The main function

- Note that the **`main`** function of the simple.c program defines a *display callback* function named **`display`**
  - Every program must have a display callback
  - The display callback is executed whenever OpenGL decides the display must be refreshed, for example when the window is opened.

- The **`main`** function ends with the program entering an event loop

15

15

main

```
int main(int argc, char** argv) {
  glutInit(&argc, argv);
   glutCreateWindow(“Hello GLUT!");
   glutDisplayFunc(display);
   glutMainLoop();
}
```

16

16

Another Simple Program:

Generating a square on a solid background

```
#include <GL/glut.h>
void mydisplay(){
 glClear(GL_COLOR_BUFFER_BIT);
  glBegin(GL_POLYGON);
      glVertex2f(-0.5, -0.5);
      glVertex2f(-0.5, 0.5);
      glVertex2f(0.5, 0.5);
      glVertex2f(0.5, -0.5);
  glEnd();
  glFlush();
}
int main(int argc, char** argv){
  glutInit(&argc, argv);
  glutCreateWindow("simple");
  glutDisplayFunc(mydisplay);
  glutMainLoop();
}
```

Lack of Object Orientation

- OpenGL is not object oriented so there are multiple functions for a given logical function
  - **glVertex3f**
  - **glVertex2i**
  - **glVertex3dv**

OpenGL function format

function name

dimensions

`glVertex3f(x,y,z)`

belongs to core GL library

`x,y,z` are `floats`

`glVertex3fv(p)`

`p` is a pointer to an array

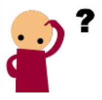
19

19

OpenGL function format

`glVertex3fv`

`How many versions of the glVertex function exist ?`



20

20

OpenGL function format

```
    glBegin(GL_POLYGON);
        glVertex2f(-0.5, -0.5);
        glVertex2f(-0.5, 0.5);
        glVertex2f(0.5, 0.5);
        glVertex2f(0.5, -0.5);
    glEnd();
```

```
How to replace glVertex2f with
glVertex2fv?
```

21

Queen Mary
University of London

OpenGL function format

```
GLfloat vertices[4][2] = {{-0.5, -0.5},
{-0.5, 0.5}, {0.5, 0.5}, {0.5, -0.5}};

    glBegin(GL_POLYGON);
        glVertex2fv(vertices[0]);
        glVertex2fv(vertices[1]);
        glVertex2fv(vertices[2]);
        glVertex2fv(vertices[3]);
    glEnd();
```

22

Queen Mary
University of London

OpenGL #defines

- Most constants are defined in the include files **`gl.h`**, **`glu.h`** and **`glut.h`**
  - Note **`#include <GL/glut.h>`** should automatically include the others
  - Examples:
    - **`glBegin(GL_POLYGON)`**
    - **`glClear(GL_COLOR_BUFFER_BIT)`**

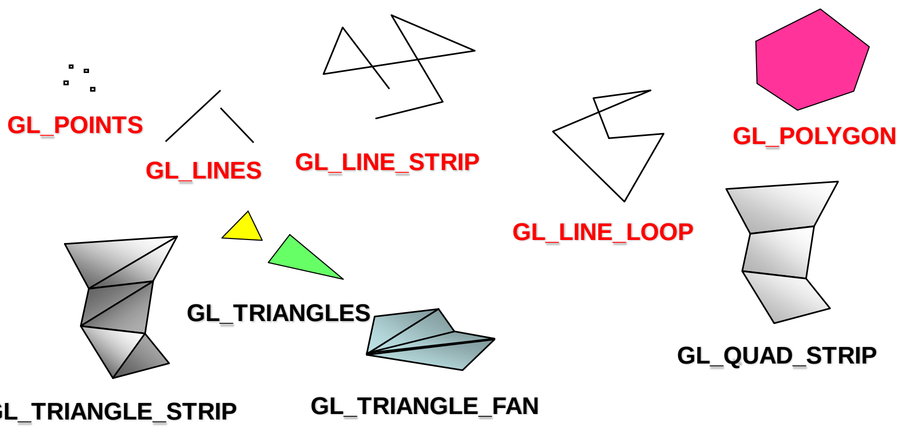
- include files also define OpenGL data types: **`GLfloat`**, **`GLdouble`**,….

23

Queen Mary University of London

23

OpenGL Primitives



24

Queen Mary University of London

24

Defaults

- Note that this program makes heavy use of variable default values for:
  - Colours (background and foreground)
  - Window parameters (e.g. size)
  - Viewing (camera parameters, e.g. where the square appears in the window …)