

# OpenGL & JOGL

Shaoting Zhang, or Tony

September 12, 2007

- 1 OpenGL
  - Brief Intro.
  - OpenGL Program
  
- 2 JOGL
  - Config & Usage
  - HW: Graphing in OpenGL

# I'm OpenGL

- Cross-language (C, C++, C#, Java, Python, Delphi)
- Cross-platform (Linux, Windows, Unix, PS3)
- Low-level (provides only rendering func.)
- State machine  
<http://www.opengl.org/documentation/specs/version1.1/state.pdf>
- Clients/server



# Reading

- The Red Book, OpenGL Programming Guide (course's webpage)
- The Blue Book, OpenGL Reference Manual
- The Green Book, for X Window System
- The Orange Book, The OpenGL Shading Language
- [nehe.gamedev.net](http://nehe.gamedev.net), highly recommended

# OpenGL Program

## packages

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

## Basic routine (main(int argc, char\*\* argv))

```
glutInit(&argc, argv);
glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
glutInitWindowPosition(0,0);
glutInitWindowSize(500,500);
glutCreateWindow(" An Example OpenGL Program");
init();
glutDisplayFunc(drawPrimitives);
glutMainLoop();
```

# OpenGL Program

## Notes:

- You needn't to include `gl` and `glu` if you include `glut`.
- `glut` inits a viewport, or, a screen.
- $(0,0)$  is the lefttop.
- `GLUT_SINGLE` means using one buffer.

# OpenGL Program

```
void init(void)
```

```
glClearColor(1.0, 1.0, 1.0, 0.0);  
glMatrixMode(GL_PROJECTION);  
gluOrtho2D(0.0, 200.0, 0.0, 200.0);
```

```
void drawPrimitives(void)
```

```
glClear(GL_COLOR_BUFFER_BIT);  
glBegin(GL_POINTS);  
    glColor3f(0.0, 1.0, 0.0);  
    glVertex2i(180, 15);  
glEnd();  
glFlush();
```

# OpenGL Program

## Notes:

- `init()` inits a clip window (textbook, figure 6-2).
- If you comment the `gluOrtho2D()`, you will see nothing. Since the default values are  $(-1, 1, -1, 1)$ , the objects are clipped.
- Some OpenGL functions might have no effect between `glBegin` and `glEnd`, like the function to change size. But you can change color between `glBegin` and `glEnd`.

# OpenGL Program

## Practice:

- In linux you can use "gcc -l glut drawDemo.c" to compile it if you have glut installed.
- Change the states to see what happen.

- 1 OpenGL
  - Brief Intro.
  - OpenGL Program
  
- 2 JOGL
  - Config & Usage
  - HW: Graphing in OpenGL

# Configure

- 1 Download JOGL:  
<https://jogl.dev.java.net/servlets/ProjectDocumentList>
- 2 Modify CLASSPATH environment variable to include jogl.jar and gluegen-rt.jar
- 3 Modify PATH(Windows) or LD\_LIBRARY\_PATH(Linux) environment variable to the "lib" directory

**note:** Dropping the JOGL jar and native library into the extension directory of the JRE is strongly discouraged. Doing so can cause conflicts with third-party applications launched via Java Web Start, and causes confusion later when upgrading the distribution.

# Basic Usage

- 1 Import appropriate classes
- 2 GL functions are prefixed with "gl."
- 3 GLU functions are prefixed with "glu."
- 4 Java types are use in place of OpenGL types (maybe need explicit casts or type specifiers)

# Basic Usage

## OpenGL

```
glBegin(GL_POLYGON);  
    glVertex3f(0.25, 0.25, 0.0);  
    glVertex3f(0.75, 0.25, 0.0);  
    glVertex3f(0.75, 0.75, 0.0);  
    glVertex3f(0.25, 0.75, 0.0);  
glEnd();
```

## JOGL

```
gl.glBegin(GL.GL_POLYGON);  
    gl.glVertex3f(0.25f, 0.25f, 0.0f);  
    gl.glVertex3f(0.75f, 0.25f, 0.0f);  
    gl.glVertex3f(0.75f, 0.75f, 0.0f);  
    gl.glVertex3f(0.25f, 0.75f, 0.0f);  
gl.glEnd();
```

# HW: Graphing in OpenGL

- Draw Lissajous curve:  
 $x = \sin(a\theta + \delta), y = \sin(b\theta), \theta \in [t, t + \pi]$   
[http://en.wikipedia.org/wiki/Lissajous\\_curve](http://en.wikipedia.org/wiki/Lissajous_curve)
- Leading point is white and trailing point is black, with a uniform decrease in intensity in between.
- You will be provided with skeleton code for this program, **only need to complete the `DrawGraph.draw()` method.**
- $t$  is already provided in `draw()`. `draw()` is called at regular time intervals.

## Extra credit:

- Variable speed
- Change shape (by adjust  $\alpha$  or  $\delta$ )
- Using color instead of grey

# Next time

- analyze the homework.
- Transform demos.