

Intro to OpenGL and JOGL

Some OpenGL Pipeline Terminology

Primitive The geometric objects that can be rendered:

- ▶ Point
- ▶ Line (segment)
- ▶ Triangle

They are specified by one, two, or three 3D vertices, respectively.

Fragment The data generated for a single pixel from a primitive.

Frame buffer A set of screen-size 2D buffers, including the color buffer(s) and z-buffer.

OpenGL Basic Structure

OpenGL is state based (~ a set switches). Methods of the API can:

- ▶ Change the OpenGL state
- ▶ Query some part of that state
- ▶ Cause OpenGL to use its current state to render something

An object holding the entire current state and current framebuffer is called an OpenGL [context](#).

OpenGL Basic Structure

OpenGL is state based (~ a set switches). Methods of the API can:

- ▶ Change the OpenGL state
- ▶ Query some part of that state
- ▶ Cause OpenGL to use its current state to render something

An object holding the entire current state and current framebuffer is called an OpenGL **context**.

Other **OpenGL objects** are designed to manage/store parts of the state. **Binding** them to a context 1) will copy their stored state values to the context 2) further state changes to the context (of types relevant for the object) will be copied and stored in their state values.

OpenGL Basic Structure

OpenGL is state based (~ a set switches). Methods of the API can:

- ▶ Change the OpenGL state
- ▶ Query some part of that state
- ▶ Cause OpenGL to use its current state to render something

An object holding the entire current state and current framebuffer is called an OpenGL **context**.

Other **OpenGL objects** are designed to manage/store parts of the state. **Binding** them to a context 1) will copy their stored state values to the context 2) further state changes to the context (of types relevant for the object) will be copied and stored in their state values.

OpenGL objects will on creation be given an integer ID (sometimes called their **name**). Their role is similar to references in Java.

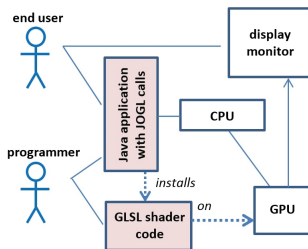
OpenGL Basic Structure

OpenGL is written in C (and should be accessible from many different programming languages), hence no overloading of methods (functions). Instead a naming scheme separates like methods with different parameters.

- ▶ `glProgramUniform3f(...)`: passes three floats
- ▶ `glProgramUniform2iv(...)`: passes a vector (array) of pairs of ints.

Shaders

A *shader* is a small program executing part of the pipeline. Compiled and sent to the GPU by the Java program (at runtime) via OpenGL.

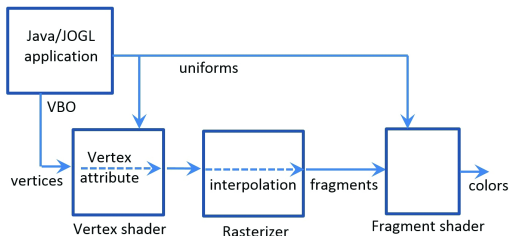


For OpenGL, the programming language of shaders is called GLSL. It is simple and C-like.

Shaders

Communication needed:

Java program \rightarrow shader \rightarrow shader \rightarrow color buffer/screen



- ▶ Vertex attribute variable: interpolated for each primitive during rasterization (vertices \rightarrow fragments).
- ▶ Uniforms variable: same value for all fragments from all primitives during the rendering with this program.

In/out variables in shaders. Identification of equally named variables in different shaders.

Some vertex attributes variables are predefined.

JOGL

- ▶ Is a Java wrapper API over OpenGL API (OpenGL is written in C).
- ▶ Also provides integration with Java GUI API (AWT, Swing), as GUI interaction is not part of OpenGL.

You will need to use both the JOGL and OpenGL documentation.

Often, arguments in OpenGL functions of pointer/array type (type `*`) are often changed to a Buffer type in JOGL.

Some JOGL

Drawing is performed via the callback methods of the interface `GLEventListener`:

- ▶ `init` Called once when OpenGL is first started up. Place set-up code and most loading of model data here.
- ▶ `display` Called when OpenGL rendering should take place (each frame, for instance). Place your main drawing code here.
- ▶ `reshape` Called once after the window has been resized. Reset some of your data, if needed.
- ▶ `dispose` Called when OpenGL shuts down. Allows you to release OpenGL resources, if relevant.

The class `FPSAnimator` makes `display` be called at regular intervals (generating the frames of an animation or game).