
Subject: : AmigaOS4

Topic: : GL4ES: another OpenGL over OpenGLES2 emulation - some tech. info and porting progress

Re: GL4ES: another OpenGL over OpenGLES2 emulation - some tech. info and porting progress

Author: : kas1e

Date: : 2019/3/28 11:16:52

URL:

@Hans

Quote:

Could you please just try what I asked?

Sure.

Originaly we draw it like 1,2,3,4,5,6 faces, so i tried that:

2,1,3,4,5,6 - nothing changes , last 2 textures overwrite first 2 ones and 6st one change the place as well.

6,2,1,3,4,5 - the same, last 2 textures , overwrite places of first 2 ones and 6st one change the place as well.

In other words, does not matter what : first 3 faces always draws where they should be and of correct look. But then, if we add 4st or 5st faces (or both at the same time) then they overwrite 3st one and 1st one. And when all 6 faces draws, it mess things even more: only top (2st one) one stays correctly , but other ones change and place and color.

If you need more better explaining, you can ask based on the source coming next, which faces and in which order put, and i will made just a videos.

Quote:

I've taken a peek at GL4ES' code, and it has code to convert both `GL_TRIANGLE_FAN` and `GL_QUADS` to `GL_TRIANGLES`. I don't know if that code is being used, because there are config variables to enable/disable various features. Differing config settings may also result in code working on one system but not another...

Ptitseb says that conversion will happens only if needed. Here, in our case, it is not needed, because the merger is not used. So `GL_TRIANGLES_FAN` will be sent as-is.

@All

At moment i strip lugaru so much, that its now have no menu, nothing, just you run binary, and it show that cube, and "Esc" exit from, few days more and it will be simple test case probabaly.

I even remove totally textures, and use just dummy colored textures (white,black, red, green, blue and yeellow), so i can see easily now what-where.

I.e. that how it looks like now:

```
void SkyBox::draw()
{
    static float size= 25;
    static float size2= -25;

    glEnable(GL_TEXTURE_2D);

    GLuint tex1;
    glGenTextures(1, &tex1);
    uint32_t red = 0xff0000ff; //RGBA : red
    glBindTexture(GL_TEXTURE_2D, tex1);
    glTexImage2D(GL_TEXTURE_2D, 0, GL_RGBA, 1, 1, 0, GL_RGBA, GL_UNSIGNED_BYTE, &red);
    glBegin(GL_TRIANGLE_FAN);
    glNormal3f( 0.0f, 0.0f, -1);
    glTexCoord2f(0, 0); glVertex3f(size2, size2, size);
    glTexCoord2f(1, 0); glVertex3f( size, size2, size);
    glTexCoord2f(1, 1); glVertex3f( size, size, size);
    glTexCoord2f(0, 1); glVertex3f(size2, size, size);
    glEnd();

    GLuint tex2;
    glGenTextures(1, &tex2);
    uint32_t green = 0x00ff00ff; //RGBA : green
    glBindTexture(GL_TEXTURE_2D, tex2);
    glTexImage2D(GL_TEXTURE_2D, 0, GL_RGBA, 1, 1, 0, GL_RGBA, GL_UNSIGNED_BYTE, &green);
    glBegin(GL_TRIANGLE_FAN);
    glNormal3f( 0.0f, 0.0f, 1);
    glTexCoord2f(1, 0); glVertex3f(size2, size2, size2);
    glTexCoord2f(1, 1); glVertex3f(size2, size, size2);
    glTexCoord2f(0, 1); glVertex3f( size, size, size2);
    glTexCoord2f(0, 0); glVertex3f( size, size2, size2);
    glEnd();

    GLuint tex3;
    glGenTextures(1, &tex3);
```

```
uint32_t blue = 0x0000ffff; //RGBA : blue
glBindTexture(GL_TEXTURE_2D, tex3);
glTexImage2D(GL_TEXTURE_2D, 0, GL_RGBA, 1, 1, 0, GL_RGBA, GL_UNSIGNED_BYTE, &blue);
glBegin(GL_TRIANGLE_FAN);
glNormal3f( 0.0f, -1.0f, 0);
glTexCoord2f(0, 1); glVertex3f(size2, size, size2);
glTexCoord2f(0, 0); glVertex3f(size2, size, size);
glTexCoord2f(1, 0); glVertex3f(size, size, size);
glTexCoord2f(1, 1); glVertex3f(size, size, size2);
glEnd();
```

```
GLuint tex4;
glGenTextures(1, &tex4);
uint32_t yellow = 0xffff00ff; //RGBA : yellow
glBindTexture(GL_TEXTURE_2D, tex4);
glTexImage2D(GL_TEXTURE_2D, 0, GL_RGBA, 1, 1, 0, GL_RGBA, GL_UNSIGNED_BYTE, &yellow);
glBegin(GL_TRIANGLE_FAN);
glNormal3f( 0.0f, 1.0f, 0);
glTexCoord2f(0, 0); glVertex3f(size2, size2, size2);
glTexCoord2f(1, 0); glVertex3f(size, size2, size2);
glTexCoord2f(1, 1); glVertex3f(size, size2, size);
glTexCoord2f(0, 1); glVertex3f(size2, size2, size);
glEnd();
```

```
GLuint tex5;
glGenTextures(1, &tex5);
uint32_t white = 0xffffffff; //RGBA : white
glBindTexture(GL_TEXTURE_2D, tex5);
glTexImage2D(GL_TEXTURE_2D, 0, GL_RGBA, 1, 1, 0, GL_RGBA, GL_UNSIGNED_BYTE, &white);
glBegin(GL_TRIANGLE_FAN);
glNormal3f( -1.0f, 0.0f, 0);
glTexCoord2f(1, 0); glVertex3f(size, size2, size2);
glTexCoord2f(1, 1); glVertex3f(size, size, size2);
glTexCoord2f(0, 1); glVertex3f(size, size, size);
glTexCoord2f(0, 0); glVertex3f(size, size2, size);
glEnd();
```

```
GLuint tex6;
glGenTextures(1, &tex6);
uint32_t black = 0x000000ff; //RGBA : black
glBindTexture(GL_TEXTURE_2D, tex6);
glTexImage2D(GL_TEXTURE_2D, 0, GL_RGBA, 1, 1, 0, GL_RGBA, GL_UNSIGNED_BYTE, &black);
glBegin(GL_TRIANGLE_FAN);
glNormal3f( 1.0f, 0.0f, 0);
glTexCoord2f(0, 0); glVertex3f(size2, size2, size2);
glTexCoord2f(1, 0); glVertex3f(size2, size2, size);
glTexCoord2f(1, 1); glVertex3f(size2, size, size);
glTexCoord2f(0, 1); glVertex3f(size2, size, size2);
glEnd();
```

}

And there is video to show how it works now via minigl and gl4es:

https://youtu.be/eML_Fa39GN4

from begining till 0:13 minigl binary , and from 0:14 till end the same code compiled via gl4es.

If watch closer when gl4es vesion runs (between 14 and 16 seconds) can be seen that first 3 faces draws firstly, and then immediately overwrites by next 3 ones, as well as a change a place a bit (just like i describe above)