RAGE
3D Objects / Cameras

RAGE game builds objects and moves the camera:

(only setupScene() is shown – the rest is mostly unchanged)

```
protected ManualObject makePyramid(Engine eng, SceneManager sm)
                                                    throws IOException
{  ManualObject pyr = sm.createManualObject("Pyramid");
   ManualObjectSection pyrSec =
         pyr.createManualSection("PyramidSection");
   pyr.setGpuShaderProgram(sm.getRenderSystem().
         getGpuShaderProgram(GpuShaderProgram.Type.RENDERING));

   float[] vertices = new float[]
   {  -1.0f, -1.0f, 1.0f, 1.0f, -1.0f, 1.0f, 0.0f, 1.0f, 0.0f,    //front
      1.0f, -1.0f, 1.0f, 1.0f, -1.0f, -1.0f, 0.0f, 1.0f, 0.0f,    //right
      1.0f, -1.0f, -1.0f, -1.0f, -1.0f, -1.0f, 0.0f, 1.0f, 0.0f,  //back
      -1.0f, -1.0f, -1.0f, -1.0f, -1.0f, 1.0f, 0.0f, 1.0f, 0.0f,  //left
      -1.0f, -1.0f, -1.0f, 1.0f, -1.0f, 1.0f, -1.0f, -1.0f, 1.0f, //LF
      1.0f, -1.0f, 1.0f, -1.0f, -1.0f, -1.0f, 1.0f, -1.0f, -1.0f  //RR
   };
   float[] texcoords = new float[]
   {  0.0f, 0.0f, 1.0f, 0.0f, 0.5f, 1.0f,
      0.0f, 0.0f, 1.0f, 0.0f, 0.5f, 1.0f,
      0.0f, 0.0f, 1.0f, 0.0f, 0.5f, 1.0f,
      0.0f, 0.0f, 1.0f, 0.0f, 0.5f, 1.0f,
      0.0f, 0.0f, 1.0f, 1.0f, 0.0f, 1.0f,
      1.0f, 1.0f, 0.0f, 0.0f, 1.0f, 0.0f
   };
   float[] normals = new float[]
   {  0.0f, 1.0f, 1.0f, 0.0f, 1.0f, 1.0f, 0.0f, 1.0f, 1.0f,
      1.0f, 1.0f, 0.0f, 1.0f, 1.0f, 0.0f, 1.0f, 1.0f, 0.0f,
      0.0f, 1.0f, -1.0f, 0.0f, 1.0f, -1.0f, 0.0f, 1.0f, -1.0f,
      -1.0f, 1.0f, 0.0f, -1.0f, 1.0f, 0.0f, -1.0f, 1.0f, 0.0f,
      0.0f, -1.0f, 0.0f, 0.0f, -1.0f, 0.0f, 0.0f, -1.0f, 0.0f,
      0.0f, -1.0f, 0.0f, 0.0f, -1.0f, 0.0f, 0.0f, -1.0f, 0.0f
   };
   int[] indices = new int[] { 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17 }

   FloatBuffer vertBuf = BufferUtil.directFloatBuffer(vertices);
   FloatBuffer texBuf  = BufferUtil.directFloatBuffer(texcoords);
   FloatBuffer normBuf = BufferUtil.directFloatBuffer(normals);
   IntBuffer indexBuf  = BufferUtil.directIntBuffer(indices);

   pyrSec.setVertexBuffer(vertBuf);
   pyrSec.setTextureCoordsBuffer(texBuf);
   pyrSec.setNormalsBuffer(normBuf);
   pyrSec.setIndexBuffer(indexBuf);

   Texture tex =
         eng.getTextureManager().getAssetByPath("chain-fence.jpeg");
   TextureState texState = (TextureState)sm.getRenderSystem().
         createRenderState(RenderState.Type.TEXTURE);
   texState.setTexture(tex);
   FrontFaceState faceState = (FrontFaceState) sm.getRenderSystem().
         createRenderState(RenderState.Type.FRONT_FACE);

   pyr.setDataSource(DataSource.INDEX_BUFFER);
   pyr.setRenderState(texState);
   pyr.setRenderState(faceState);
   return pyr;
}
```

```
protected void setupScene(Engine eng, SceneManager sm)
                                                    throws IOException
{  setupInputs();

   // make manual objects – in this case a pyramid
   ManualObject pyr = makePyramid(eng, sm);

   SceneNode pyrN =
         sm.getRootSceneNode().createChildSceneNode("PyrNode");
   pyrN.scale(0.75f, 0.75f, 0.75f);
   pyrN.attachObject(pyr);

   // set up earth
   Entity earthE = sm.createEntity("myEarth", "earth.obj");
   earthE.setPrimitive(Primitive.TRIANGLES);
   SceneNode earthN = sm.getRootSceneNode().
         createChildSceneNode(earthE.getName() + "Node");
   earthN.attachObject(earthE);
   earthN.setLocalPosition(-2.0f, 2.0f, -1.0f);
   earthN.setLocalScale(0.2f, 0.2f, 0.2f);

   RotationController rc = new
         RotationController(Vector3f.createUnitVectorY(), .02f);
   rc.addNode(earthN);
   rc.addNode(pyrN);
   sm.addController(rc);

   // set up lights
   sm.getAmbientLight().setIntensity(new Color(.3f, .3f, .3f));
   Light plight = sm.createLight("testLamp1", Light.Type.POINT);
   plight.setAmbient(new Color(.1f, .1f, .1f));
   plight.setDiffuse(new Color(0.8f, 0.8f, 0.8f));
   plight.setSpecular(new Color(1.0f, 1.0f, 1.0f));
   plight.setRange(20f);

   SceneNode plightNode =
         sm.getRootSceneNode().createChildSceneNode("plightNode");
   plightNode.attachObject(plight);
   plightNode.setLocalPosition(1.0f, 1.0f, 5.0f);
}
```

an Action class for moving the camera forward:

```
import ray.input.action.AbstractInputAction;
import ray.rage.scene.*;
import ray.rage.game.*;
import ray.rml.*;
import net.java.games.input.Event;

public class MoveForwardAction extends AbstractInputAction
{
   private Camera camera;

   public MoveForwardAction(Camera c)
   { camera = c;
   }

   public void performAction(float time, Event e)
   {  Vector3f v = camera.getFd();
      Vector3f p = camera.getPo();
      Vector3f p1 =
         (Vector3f) Vector3f.createFrom(0.01f*v.x(), 0.01f*v.y(), 0.01f*v.z());
      Vector3f p2 = (Vector3f) p.add((Vector3)p1);
      camera.setPo((Vector3f)Vector3f.createFrom(p2.x(),p2.y(),p2.z()));
   }
}
```