Sound – RAGE / JOAL

```
. . .
import ray.audio.*;
import com.jogamp.openal.ALFactory;

public class MyGame extends VariableFrameRateGame
{ . . .
  private Tessellation tessTest;
  private float movemt = 0.01f;

  IAudioManager audioMgr;
  Sound oceanSound, hereSound;

  //  constructor, setupWindow, setupCamera, and main as before

  protected void setupScene(Engine eng, SceneManager sm)
                                                    throws IOException
  {
   // make dolphin avatar
   . . .

   // make robot and load animation
   . . .

   // make waterfall object
   . . .

   // set up lights
   . . .

   // set up orbit camera
   . . .

   // set up terrain
   tessTest = sm.createTessellation("tessTest", 7);
   tessTest.setSubdivisions(4f);
   SceneNode tessTestNode =
      sm.getRootSceneNode().createChildSceneNode("node_tessTest");
   tessTestNode.attachObject(tessTest);
   tessTestNode.scale(10, 3, 10);
   tessTest.setHeightMap(this.getEngine(), "smallMoonHeight.jpg");
   tessTest.setTexture(this.getEngine(), "smallMoonTexture.jpg");

   //  tile the terrain
   tessTest.getTextureState().setWrapMode(
                                        WrapMode.REPEAT_MIRRORED);
   tessTest.setHeightMapTiling(4);
   tessTest.setTextureTiling(4);
   tessTest.setNormalMapTiling(4);

   initAudio(sm);
  }


  public void setEarParameters(SceneManager sm)
  { SceneNode dolphinNode = sm.getSceneNode("dolphinNode");
    Vector3 avDir = dolphinNode.getWorldForwardAxis();

    //  note - should get the camera's forward direction
    //    - avatar direction plus azimuth

    audioMgr.getEar().setLocation(dolphinNode.getWorldPosition());
    audioMgr.getEar().setOrientation(avDir, Vector3f.createFrom(0,1,0));
  }
```

```
  protected void update(Engine engine)
  {
   // update the inputmanager, controllers, AI, animation, etc.
   . . .

   SceneManager sm = engine.getSceneManager();
   SceneNode robotN = sm.getSceneNode("robotNode");
   SceneNode earthN = sm.getSceneNode("earthNode");

   // move robot
   . . .

   hereSound.setLocation(robotN.getWorldPosition());
   oceanSound.setLocation(earthN.getWorldPosition());
   setEarParameters(sm);
  }

  public void initAudio(SceneManager sm)
  { AudioResource resource1, resource2;
    audioMgr = AudioManagerFactory.createAudioManager(
                                        "ray.audio.joal.JOALAudioManager");
    if (!audioMgr.initialize())
    { System.out.println("Audio Manager failed to initialize!");
      return;
    }
    resource1 = audioMgr.createAudioResource("here.wav",
                                        AudioResourceType.AUDIO_SAMPLE);
    resource2 = audioMgr.createAudioResource("ocean.wav",
                                        AudioResourceType.AUDIO_SAMPLE);
    hereSound = new Sound(resource1,
                                        SoundType.SOUND_EFFECT, 100, true);
    oceanSound = new Sound(resource2,
                                        SoundType.SOUND_EFFECT, 100, true);
    hereSound.initialize(audioMgr);
    oceanSound.initialize(audioMgr);
    hereSound.setMaxDistance(10.0f);
    hereSound.setMinDistance(0.5f);
    hereSound.setRollOff(5.0f);
    oceanSound.setMaxDistance(10.0f);
    oceanSound.setMinDistance(0.5f);
    oceanSound.setRollOff(5.0f);

    SceneNode robotN = sm.getSceneNode("robotNode");
    SceneNode earthN = sm.getSceneNode("earthNode");
    hereSound.setLocation(robotN.getWorldPosition());
    oceanSound.setLocation(earthN.getWorldPosition());
    setEarParameters(sm);

    hereSound.play();
    oceanSound.play();
  }
}
```