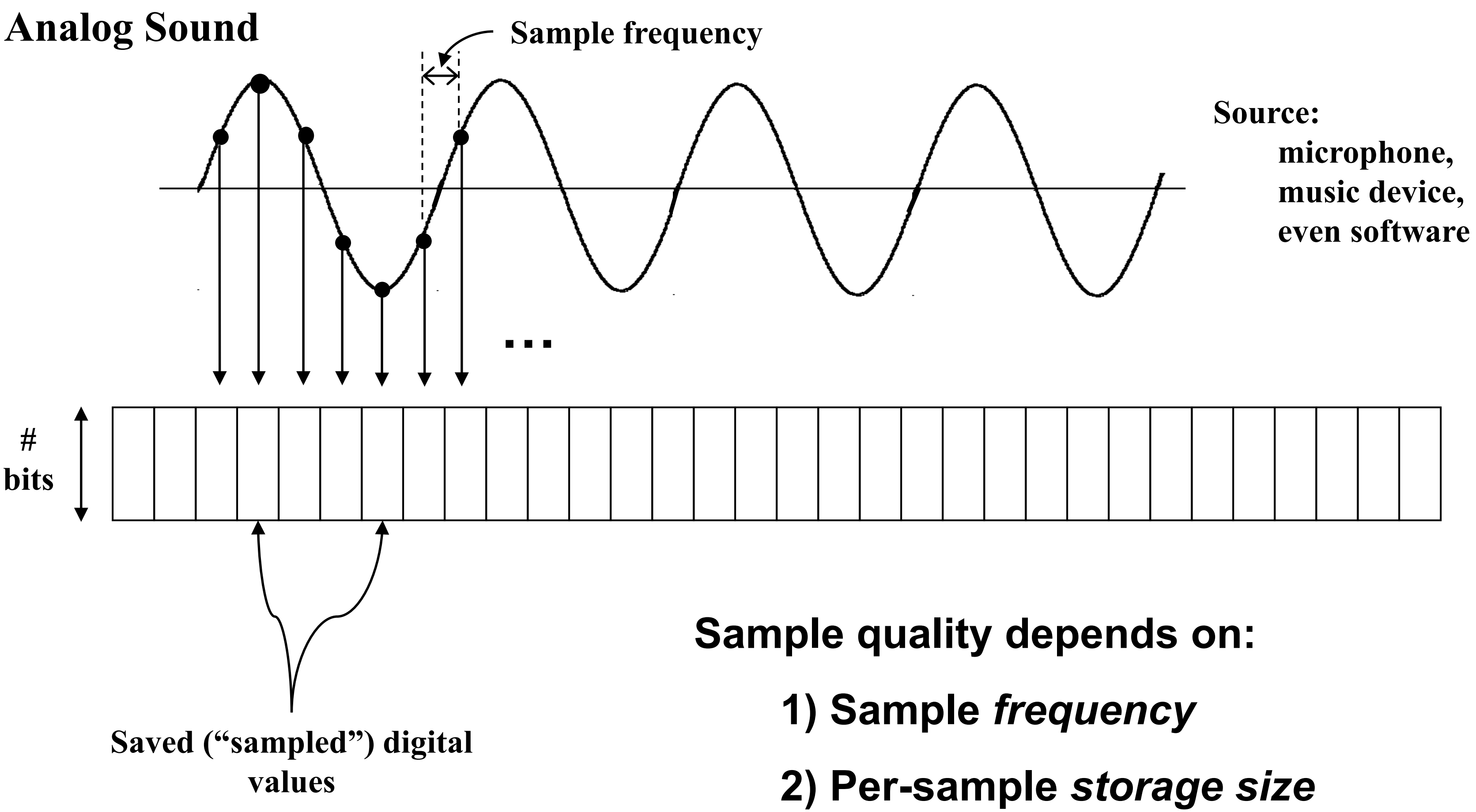
CSc 165
Computer Game Architecture

15 - Sound

Overview

- Sound Characteristics & File Formats (CSc 133)
- Sound APIs
- 3D Sound
- OpenAL and JOAL
- Audio support in TAGE

Sampled Audio (from CSc-133)



Sample quality depends on:

1) Sample *frequency*

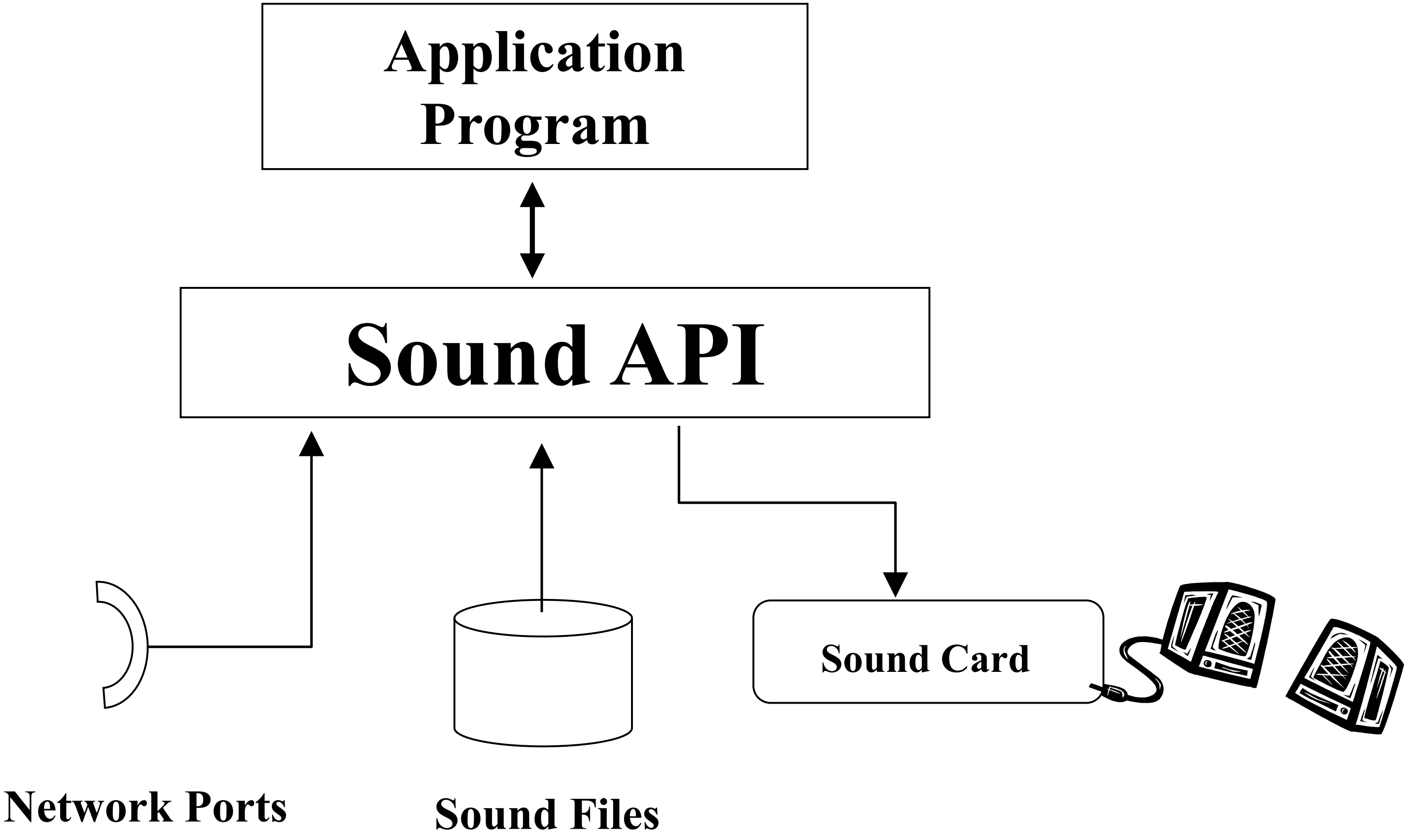
2) Per-sample *storage size* *(bits per sample)*

Sound File Formats

`.au`	Sun Audio File (Unix/Linux)
`.aiff`	Audio Interchange File Format (Mac)
`.cda`	CD Digital Audio (track information)
`.mpx`	MPEG Audio (mp, mp2, mp3, mp4)
`.mid`	MIDI file (sequenced, not sampled)
`.ogg`	Ogg-Vorbis file (open source)
`.ra`	Real Audio (designed for streaming)
`.wav`	Windows “wave file”

Finding sound files: **www.findsounds.com**

Sound APIs


Application Program
Sound API
Sound Card
Network Ports
Sound Files

Popular Sound API’s

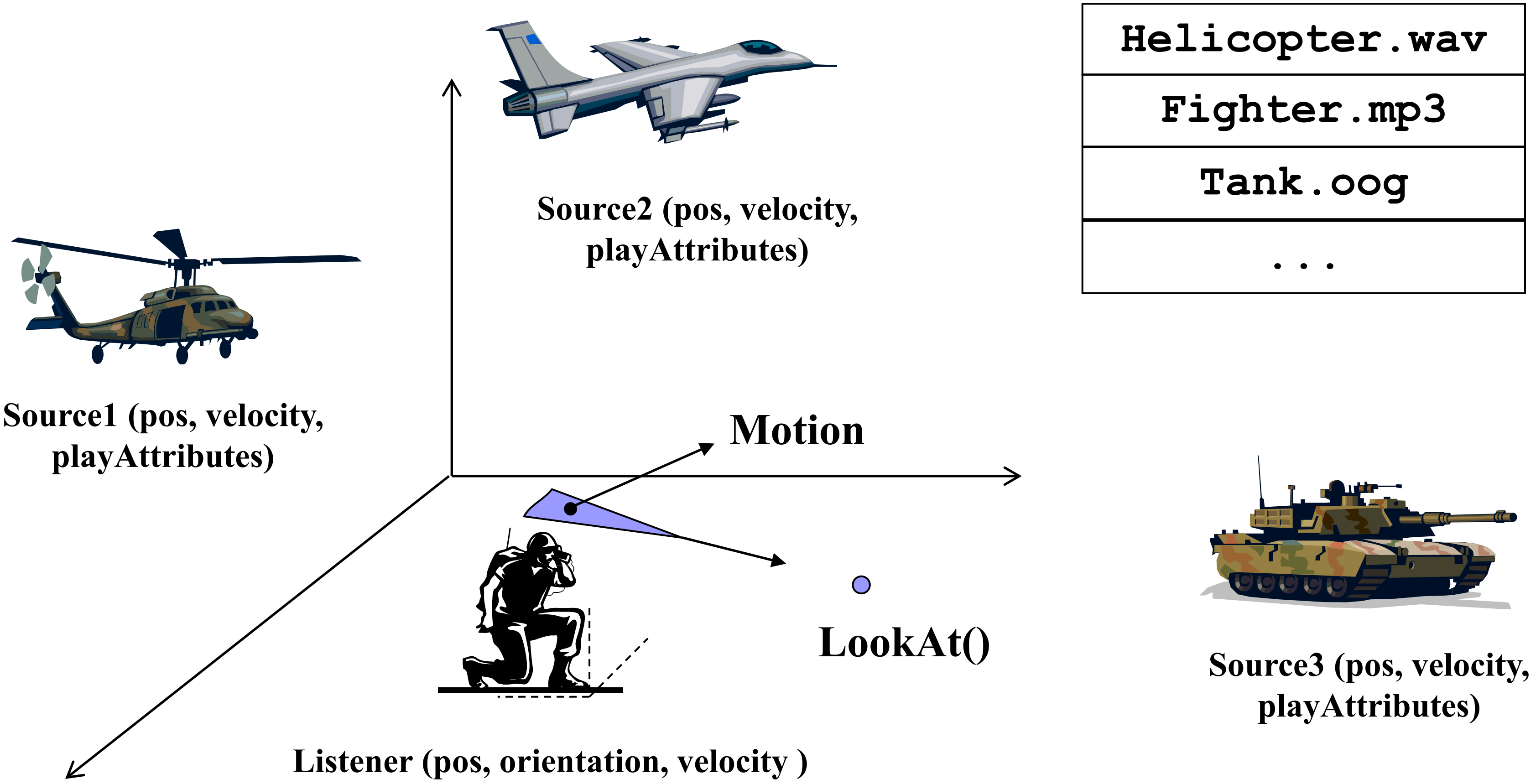
2D Sound

- Java **`AudioClip`**
- JavaSound

3D Sound

- DirectSound / DirectSound3D
- Linux Open Sound System (OSS)
- Advanced Linux Sound Architecture (ALSA)
- OpenAL / JOAL

3D Sound



Source2 (pos, velocity, playAttributes)

Source1 (pos, velocity, playAttributes)

Motion

LookAt()

Source3 (pos, velocity, playAttributes)

Listener (pos, orientation, velocity)

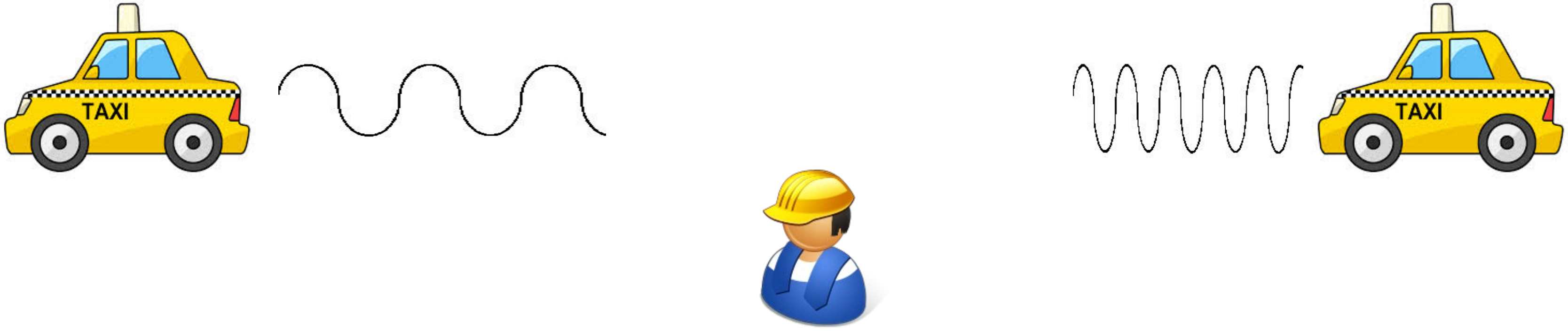
`Helicopter.wav`
`Fighter.mp3`
`Tank.oog`
. . .

directional, distance attenuation, doppler shift, etc….

Doppler Effect

Change in frequency due to relative motion between Source and Listener

- Approaching Source == compressed waves
  == higher frequency
- Receding Source == expanded waves
  == lower frequency



- "Open Audio Library" 3D Audio API

  `https://openal.org`

- Open-source (although one version is proprietary)
- Cross-platform
- Modeled after OpenGL
- Updated fork: "OpenAL Soft"
  (also open source project)

OpenAL Platform Support

- Macintosh (OS 8/9/X)
- Linux (OSS & ALSA)
- BSD
- Solaris
- IRIX
- Android
- iOS

- Windows
- Sony PlayStation 2, 3, Portable
- Microsoft Xbox & 360
- Nintendo GameCube
- Wii

…and many others!

JOAL

The Java OpenAL Binding

- a “sibling” of JOGL
- part of the Sun Java Gaming Initiative

Works much like JOGL

- Java wrappers around OpenAL components
- method calls mimic C calls
- uses OpenAL Soft

`http://jogamp.org`

Main OpenAL Components

Sources

- A source of sound generation in the world
- Attributes include position, orientation, velocity…

Listeners

- An entity that *hears* sound(s)
- One (per context), usually attached to the *Player*
- Also has attributes for location and orientation

Buffers

- Attached to *sources*
- Hold *audio data* (e.g. sound files)
- “Play” their contents

“AL” Function Categories

“Source” functions

- Create one or more ‘source’ objects and return their “ID”s

```
alGenSources(int numSrcs, int [] sourceIDs, int offset)
```

- Set properties of a source

```
alSourcef (int srcID, int prop, float value)
alSourcefv (int srcID, int prop, float [] values, int offset)
```

Example properties: AL_POSITION, AL_VELOCITY, AL_DIRECTION,
AL_BUFFER, AL_LOOPING, AL_SOURCE_RELATIVE, …

- Get properties of a source

```
alGetSourcei (int srcID, int prop, int [] value, int offset)
alGetSourcefv (int srcID, int prop, float [] values, int offset)
```

- Manage source sound(s)

```
alSourcePlay/Pause/Stop/Rewind (int srcID)
alSourcePlayv (int numSrcs, int[]srcIDs, int offset)
alSourceQueueBuffers (int srcID, int num, int)
```

“AL” Function Categories (cont.)

“Buffer” functions

- Create one or more ‘buffer’ objects and return their IDs

```
alGenBuffers (int numBufs, int [] bufferIDs, int offset)
```

- Set properties of a buffer

```
alBufferf (int bufID, int prop, float value)
alBufferfv (int bufID, int prop, float [] values, int offset)
```

Example properties: AL_FREQUENCY, AL_BITS, AL_CHANNELS, AL_SIZE …

- Get properties of a buffer

```
alGetBufferi (int bufID, int prop, int [] value, int offset)
alGetBufferfv (int bufID, int prop, float [] values, int offset)
```

- Load sound data into a buffer

```
alBufferData (int bufID, int format, Buffer data, int size, int freq)
```

"AL" Function Categories (cont.)

"Listener" functions

- Set properties of a listener

```
alListenerf (int prop, float value)
alListenerfv (int prop, float [] values, int offset)
```

Example properties: AL_POSITION, AL_VELOCITY, AL_ORIENTATION...

- Get properties of a listener

```
alGetListeneri (int prop, int [] value, int offset)
alGetListenerfv (int prop, float [] values, int offset)
```

OpenAL Code Example

```
import com.jogamp.openal.AL;
import com.jogamp.openal.ALFactory;
import com.jogamp.openal.util.ALut;

/** This class demonstrates the use of a single non-moving OpenAL Source
 *  to play a sound. It was adapted from the OpenAL Tutorial Lesson #1
 *  at  http://www.openal.org
 */
public class SingleFixedSourceDemo
{   private AL al ;
    private int[] bufferID ;          // OpenAL ID of buffer to hold sound data
    private int[] sourceID ;          // OpenAL ID of sound source

    // specify sound Source position and velocity
    private float[] sourcePos = { 0.0f, 0.0f, 0.0f };
    private float[] sourceVel = { 0.0f, 0.0f, 0.0f };

    //specify the Listener’s position and velocity
    private float[] listenerPos = { 0.0f, 0.0f, 0.0f };  // Position = origin
    private float[] listenerVel = { 0.0f, 0.0f, 0.0f };  // Velocity = zero

    // Set the orientation of the listener:  "lookAt” followed by "up"
    private float[] listenerOri = { 0.0f, 0.0f, -1.0f,  0.0f, 1.0f, 0.0f };
    ...
```

OpenAL Code Example (cont.)

```
/** This constructor obtains an “AL” object, initializes the ALUT toolkit,
 *  sets up the OpenAL listener object, and invokes a “run()” method to
 *  read input keys and play sounds.
 */

public SingleFixedSourceDemo()
{   al = ALFactory.getAL();          // get an initialized AL object
    ALut.alutInit();                 // initialize the toolkit
    al.alGetError();                 // clear the error bit

    al.alListenerfv(AL.AL_POSITION, listenerPos);  // initialize the listener
    al.alListenerfv(AL.AL_VELOCITY, listenerVel);
    al.alListenerfv(AL.AL_ORIENTATION, listenerOri);

    run();                    // read keys, play corresponding sound file
}
```

OpenAL Code Example (cont.)

```
/** Read a character and load/play the corresponding audio file */
private void run()
{  boolean done = false;
   char inputChar ;
   int result ;
   while (!done)
   {  ... code here to read a keyboard char
     switch (inputChar)
     {  case '1' :
          result = loadWavFileData(“Hello.wav") ;
          if (result == AL.AL_FALSE)
          {  throw new RuntimeException ("Error loading selected file");
          }  else
          {  //play the sound file
             al.alSourcePlay(source[0]);
          }
          break;
        //code here for additional character cases...
        ...
     }
   }
   shutDownAL();
}
```

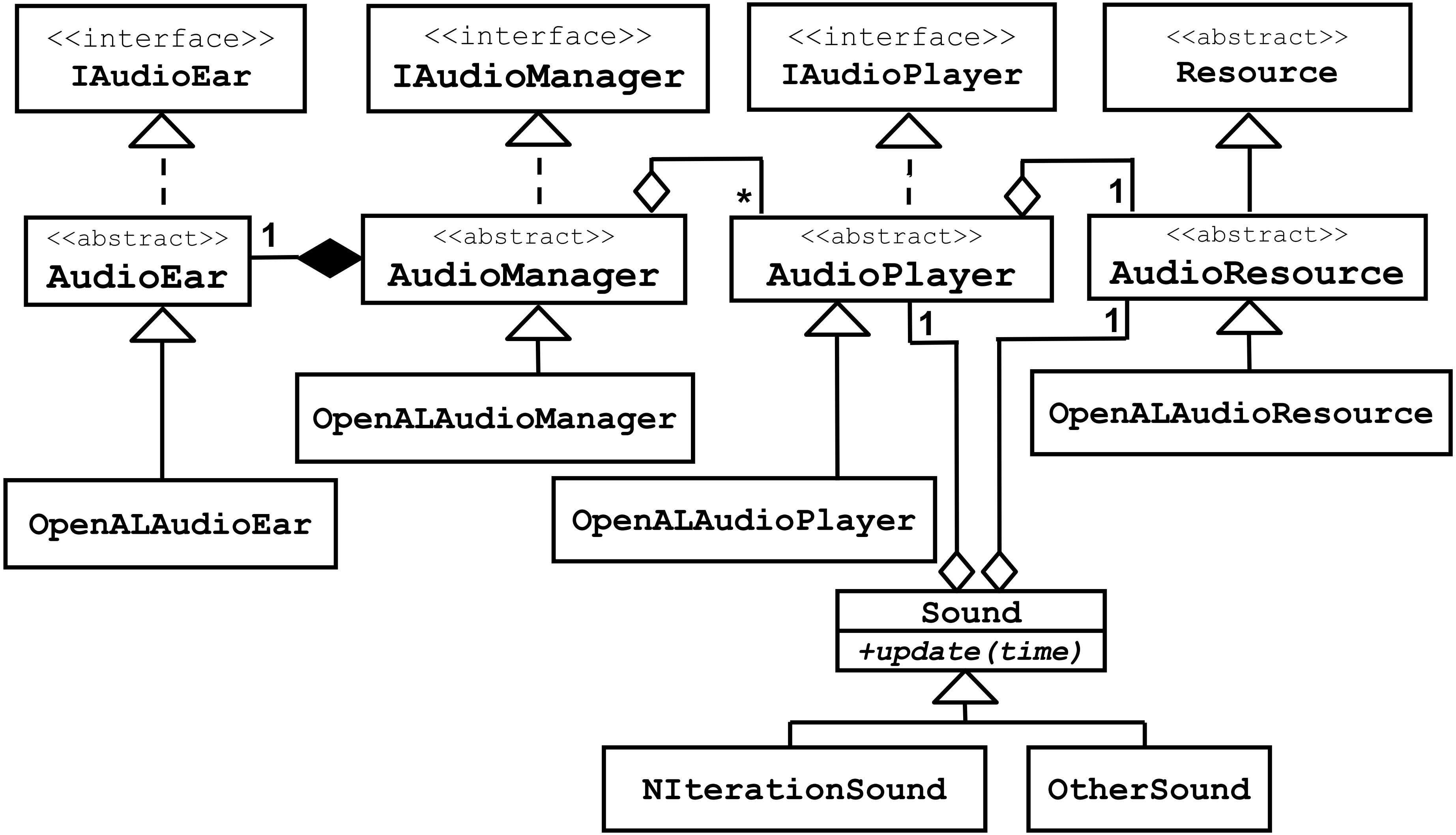
OpenAL Code Example (cont.)

```
private int loadWavFileData(String filename)
{   //create arrays to hold the wav file information
    int[] format = new int[1];
    int[] size = new int[1];
    ByteBuffer[] data = new ByteBuffer[1];
    int[] freq = new int[1];
    int[] loop = new int[1];

    // Load wav information from 'filename' into program arrays
    ALut.alutLoadWAVFile(filename, format, data, size, freq, loop);

    //get an OpenAL buffer ID
    bufferID = new int [1];
    al.alGenBuffers(1, bufferID, 0);
    if (al.alGetError() != AL.AL_NO_ERROR)
         return AL.AL_FALSE;

    //load the wav file data into an OpenAL buffer
    al.alBufferData(bufferID[0], format[0], data[0], size[0], freq[0]);

    //get an OpenAL source ID
    sourceID = new int[1];
    al.alGenSources(1, sourceID, 0);
    if (al.alGetError() != AL.AL_NO_ERROR)
         return AL.AL_FALSE;

    ... continued ...
```

OpenAL Code Example (cont.)

```
        //... loadWavFileData continued...

        // Bind buffer with source
        al.alSourcei(sourceID[0], AL.AL_BUFFER, bufferID[0]);

        //set source characteristics
        al.alSourcefv(sourceID[0], AL.AL_POSITION, sourcePos, 0);
        al.alSourcefv(sourceID[0], AL.AL_VELOCITY, sourceVel, 0);
        al.alSourcef(sourceID[0], AL.AL_PITCH, 1.0f);
        al.alSourcef(sourceID[0], AL.AL_GAIN, 1.0f);
        al.alSourcei(sourceID[0], AL.AL_LOOPING, loop[0]);

        // Do another error check and return.
        if (al.alGetError() = AL.AL_NO_ERROR)
            return AL.AL_TRUE;
        else return AL.AL_FALSE;
    }//end loadWavFileData()

    private void shutdownAL()
    {   al.alDeleteBuffers(1, bufferID, 0);
        al.alDeleteSources(1, sourceID, 0);
        ALut.alutExit();
    }

}//end class SingleFixedSourceDemo
```

Audio Support in TAGE



TAGE audio example

```
AudioResource resource1;

audioMgr =
      AudioManagerFactory.createAudioManager("tage.audio.joal.JOALAudioManager");
if(!audioMgr.initialize())
{  System.out.println("Audio Manager failed to initialize");
   return;
}

resource1 = audioMgr.createAudioResource("assets/sounds/meow.wav",
                                         AudioResourceType.AUDIO_SAMPLE);

npcSound = new Sound(resource1, SoundType.SOUND_EFFECT, 100, true);
npcSound.initialize(audioMgr);
npcSound.setMaxDistance(50.0f);
npcSound.setMinDistance(3.0f);
npcSound.setRollOff(5.0f);
npcSound.setLocation(npc.getWorldLocation());

setEarParameters();
.  .  .
npcSound.play();
```

TAGE audio example (cont.)

```
public void setEarParameters(SceneManager sm)
{   Camera camera =
        (engine.getRenderSystem()).getViewport("MAIN").getCamera();
    audioMgr.getEar().setLocation(avatar.getWorldLocation());
    audioMgr.getEar().setOrientation(camera.getN(),
                                    new Vector3f(0.0f, 1.0f, 0.0f));
}
```