# OpenAL Sounds

2008년 여름
박경신

## OpenAL 3D Sound Library

- OpenAL 3D Sound Library
  - A cross-platform 3D audio API
  - 1인 Listener를 위한 3차원 사운드 지원
  - OpenGL AP와 비슷한 용도로 제작되었음.
- Basic OpenAL objects
  - Listener
    - 1 Listener object은 소리를 듣는 사용자의 위치를 말함
  - Buffers
    - audio data를 저장하는 곳
  - Sources
    - 오디오 사운드가 3차원 공간에 있는 위치



## Programming with OpenAL

- OpenAL로 static audio file을 play하는 프로그램 작성하기
  1. Header files을 include 함
  2. OpenAL 초기화
  3. 오디오 데이터 가진 buffer 생성
  4. Source 생성
  5. Source를 buffer에 첨부
  6. Source를 play/stop 함
  7. Clean up



## Programming with OpenAL

1. Include Header files

```
// We'll use the framework code from the samples of OpenAL Source
#include "Framework.h"
```

2. Initialize OpenAL

```
ALuint uiBuffer;      // buffer hold sound data
ALuint uiSource;      // sources are points emitting sound
ALint iState;

ALFWInit();  // Initialize framework
ALFWInitOpenAL();  // Initialize OpenAL
```

## Programming with OpenAL

3. Generate buffers with audio data

```
alGenBuffers(1, &uiBuffer);  // Generate an AL Buffer
ALFWLoadWaveToBuffer((char *)
ALFWaddMediaPath("audio_file"), uiBuffer));
```

4. Create source

```
alGenSources(1, &uiSource);
```

5. Attach source to buffer

```
alSourcei(uiSource, AL_BUFFER, uiBuffer);
```



## Programming with OpenAL

6. Play/Stop source

```
alSourcePlay(uiSource);
alSourceStop(uiSource);
```

7. Cleanup

```
alDeleteSources(1, &uiSource);
alDeleteBuffers(1, &uiBuffer);
ALFWShutdownOpenAL();
ALFWShutdown();
```



## Programming with OpenAL

- alSource* 함수를 이용하여 더욱 많은 기능을 넣을 수 있다.

```
// Position of the source sound
ALfloat sourcePos[] = { 0.0, 0.0, 0.0 };

// Velocity of the source sound
ALfloat sourceVel[] = { 0.0, 0.0, 0.0 };

alSourcefv(uiSource, AL_POSITION, sourcePos);
alSourcefv(uiSource, AL_VELOCITY, sourceVel);
alSourcei(uiSource, AL_LOOPING, 1); // set loop mode
```



## Programming with OpenAL

- Listener의 속성(properties)

```
// Position of the listener
ALfloat listenerPos[] = { 0.0, 0.0, 0.0 };
// Velocity of the listener
ALfloat listenerVel[] = { 0.0, 0.0, 0.0 };
// Orientation of the listener
// First 3 elements are "at", second 3 are "up"
ALfloat listenerOri[] = { 0.0, 0.0, -1.0, 0.0, 1.0, 0.0 };

alListenerfv(AL_POSITION, listenerPos);
alListenerfv(AL_VELOCITY, listenerVel);
alListenerfv(AL_ORIENTATION, listenerOri);
```

## Multiple Sounds

1. 여러 개의 Buffer와 Source (각 Source마다 Position과 Velocity 속성값 지정)를 생성하고 하나의 Listener 생성

```
#include “Framework.h”       // include openal sound framework class
// three sound sources
#define MAX_NUM 3
ALuint uiBuffers[MAX_NUM];;
ALuint uiSources[MAX_NUM];
ALfloat sourcePos[MAX_NUM][3];
ALfloat sourceVel[MAX_NUM][3];
// one listener
ALfloat listenerPos[] = { 0., 0., 0.};
ALfloat listenerVel[] = { 0., 0., 0.};
ALfloat listenerOri[] = { 0., 0., -1., 0., 1., 0.};
```



## Multiple Sounds

2. OpenAL 초기화

```
void InitSound()
{
    // Initialize Framework
    ALFWInit();

    if (!ALFWInitOpenAL()) {
     ALFWprintf("Failed to initialize OpenAL\n");
     ALFWShutdown();
    }
}
```



## Multiple Sounds

3. 멀티 사운드 Source와 Listener를 지정

```
void SetSoundSources()
{
    // Generate three AL Buffers
    alGenBuffers(MAX_NUM, uiBuffers );

    // Load a Wave file into OpenAL Buffer
    if (!ALFWLoadWaveToBuffer("stereo.wav", uiBuffers[0])) {
     ALFWprintf("Failed to load rnjAR.wav \n");
    }
    …..
    // Generate a Source to playback the Buffer
    alGenSources(MAX_NUM, uiSources );
```



## Multiple Sounds

```
    // Generate a Source Pos & Vel
    …
    // Attach Source to Buffer
    alSourcei( uiSources[0], AL_BUFFER, uiBuffers[0] );
    alSourcefv( uiSources[0], AL_POSITION, sourcePos[0]);
    alSourcefv( uiSources[0], AL_VELOCITY, sourceVel[0]);
    alSourcei( uiSources[0], AL_LOOPING, AL_TRUE);
    alSourcePlay( uiSources[0] );
    …
    alSourcei( uiSources[2], AL_BUFFER, uiBuffers[2] );
    alSourcefv( uiSources[2], AL_POSITION, sourcePos[2]);
    alSourcefv( uiSources[2], AL_VELOCITY, sourceVel[2]);
}
```

## Multiple Sounds

```
void SetListner() {
    alListenerfv(AL_POSITION,    listenerPos);
    alListenerfv(AL_VELOCITY,    listenerVel);
    alListenerfv(AL_ORIENTATION, listenerOri);
}
```

4. 멀티 사운드 Source와 Buffer의 Cleanup

```
void CleanupSound() {
    for (int i = 0; i < 3; i++)
        alSourceStop(uiSources[i]);
    alDeleteSources(MAX_NUM, &uiSources[0]);
    alDeleteBuffers(MAX_NUM, &uiBuffers[0]);
    ALFWShutdownOpenAL();
    ALFWShutdown();
}
```



## Multiple Sounds

5. 멀티 사운드 Play/Stop

```
void ToggleMove()   // when object is moving, play sound
{
    active = !active;
    if (active)
     alSourcePlay(uiSources[1]);       // activate sound1, too
    else
     alSourceStop(uiSources[1]);       // stop playing sound1
}

void PlaySound()     // play “click.wav” sound when a user hits ‘s’-key
{
    alSourcePlay(uiSources[2]);
}
```