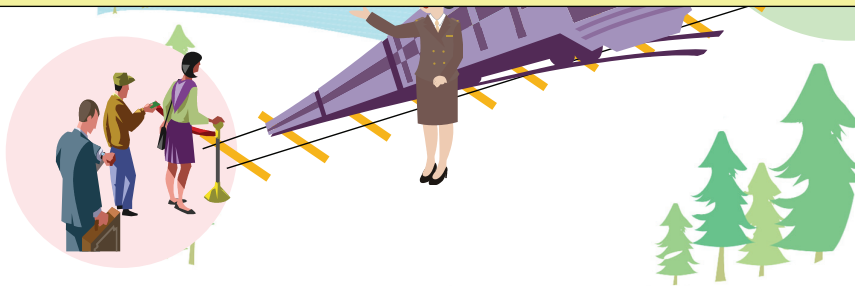


# Sound Programming

## 제10장 Programming Effects Extension in OpenAL



1



## Part1 - Effects Extension Enumeration

- 우선 내 사운드 카드에서 제공되는 **Effect Extension** 관련 기능이 무엇인지 알아보자.
- 내 사운드 카드의 **EFX** 기능 알아내는 순서
  - **Effect Extension** 기능 자체가 지원되는지 여부를 검사
    - `alcIsExtensionPresent()`
  - 생성 가능한 **Auxiliary Effects Slot** 개수를 알아낸다.
    - `alGenAuxiliaryEffectSlots()`
  - **Source** 당 최대 **Auxiliary Effect Slots Send** 개수를 알아낸다
    - `alcGetIntegerv(...,ALC_MAX_AUXILIARY_SENDS, ...);`
  - 사운드 카드에서 지원되는 **Effect**들의 종류를 알아낸다.
    - `alGenEffects() / alEffecti()`
      - REVERB, EAXREVERB, CHORUS, DISTORTION, ECHO, FLANGER, FREQUENCY\_SHIFTER, VOCAL\_MORPHER, PITCH\_SHIFTER, RING\_MODULATOR, AUTOWAH, COMPRESSOR, EQUALIZER



## EFX 기능 알아내는 순서(계속)

- 사운드 카드에서 지원되는 **Filter**들의 종류를 알아낸다.
  - 우선 **Filter** 객체를 만들고(**alGenFilters()**),
  - 모든 **Filter** 유형들을 하나씩 설정(**alFilteri()**)했을 때 오류가 나는 지를 알아보는 방법으로 수행
    - LOWPASS / HIGHPASS / BANDPASS



## EFX Enumeration 예제 코드 (EFXmain.cpp)

```
ALCdevice * pDevice;
ALuint      uiEffectSlots[128] = { 0 };
ALuint      uiEffects[1] = { 0 };
ALuint      uiFilters[1] = { 0 };
ALint       iEffectSlotsGenerated;
ALint       iSends;

// Initialize Framework
ALFWInit();
ALFWprintf("EFX Sample Application Starts.\n");

if (!ALFWInitOpenAL()) {
    ALFWprintf("Failed to initialize OpenAL\n");
    ALFWShutdown();
    return 0;
}

// Get the current AL Device
pDevice = alcGetContextsDevice(alcGetCurrentContext());

// EFX Extension 기능이 지원되는 사운드 카드인지 검사
if (ALFWIsEFXSupported()) {
    // 생성 가능한 Auxiliary Effects Slot 개수를 알아낸다.
    // 최대 개수인 128개까지 오류가 발생할 때까지 생성해간다.
    for (iEffectSlotsGenerated = 0; iEffectSlotsGenerated < 128; iEffectSlotsGenerated++) {
        alGenAuxiliaryEffectSlots(1, &uiEffectSlots[iEffectSlotsGenerated]);
        if (alGetError() != AL_NO_ERROR)
            break;
    }
    ALFWprintf("%d Auxiliary Effect Slot%s\n", iEffectSlotsGenerated, (iEffectSlotsGenerated == 1) ? "" : "s");
    .....
}
```



## EFX Enumeration 예제 코드 (계속)

```
// Source 당 최대 Auxiliary Effect Slots Send 개수를 알아낸다.
alcGetIntegerv(pDevice, ALC_MAX_AUXILIARY_SENDS, 1, &iSends);
ALFWprintf("%d Auxiliary Send%s per Source\n", iSends, (iSends == 1) ? "" : "s");

// 사운드 카드에서 지원되는 Effect들의 종류를 알아낸다. 우선 Effect 객체를 만들고,
// 여러 Effect 유형을 설정했을 때 오류가 나는지를 알아보는 방법으로 수행한다.
ALFWprintf("\nEffects Supported: -\n");
alGenEffects(1, &uiEffects[0]);
if (alGetError() == AL_NO_ERROR) {
    // 알고 있는 모든 Effect Type을 설정해서 오류가 나는지 검사한다.
    alEffecti(uiEffects[0], AL_EFFECT_TYPE, AL_EFFECT_REVERB);
    ALFWprintf("'Reverb' Support      %s\n", (alGetError() == AL_NO_ERROR) ? "YES" : "NO");

    alEffecti(uiEffects[0], AL_EFFECT_TYPE, AL_EFFECT_EAXREVERB);
    ALFWprintf("'EAX Reverb' Support      %s\n", (alGetError() == AL_NO_ERROR) ? "YES" : "NO");

    alEffecti(uiEffects[0], AL_EFFECT_TYPE, AL_EFFECT_CHORUS);
    ALFWprintf("'Chorus' Support          %s\n", (alGetError() == AL_NO_ERROR) ? "YES" : "NO");

    alEffecti(uiEffects[0], AL_EFFECT_TYPE, AL_EFFECT_DISTORTION);
    ALFWprintf("'Distortion' Support      %s\n", (alGetError() == AL_NO_ERROR) ? "YES" : "NO");

    ..... // OpenAL에서 처리할 수 있는 모든 유형의 Effect들에 대해 다 검사해본다.

    alEffecti(uiEffects[0], AL_EFFECT_TYPE, AL_EFFECT_EQUALIZER);
    ALFWprintf("'Equalizer' Support      %s\n", (alGetError() == AL_NO_ERROR) ? "YES" : "NO");
}
.....
```



## EFX Enumeration 예제 코드 (계속)

```
// 사운드 카드에서 지원되는 Filter들의 종류를 알아낸다. 우선 Filter 객체를 만들고,
// 여러 Filter 유형을 설정했을 때 오류가 나는지를 알아보는 방법으로 수행한다.
ALFWprintf("\nFilters Supported: -\n");

// 지원되는 Filter 유형에 무엇이 있는지 알아내기 위해 일단 Filter 객체 하나를 생성한다.
alGenFilters(1, &uiFilters[0]);
if (alGetError() == AL_NO_ERROR) {
    // 알려진 Filter 유형들을 차례로 설정해본다.
    alFilteri(uiFilters[0], AL_FILTER_TYPE, AL_FILTER_LOWPASS);
    ALFWprintf("'Low Pass' Support      %s\n", (alGetError() == AL_NO_ERROR) ? "YES" : "NO");

    alFilteri(uiFilters[0], AL_FILTER_TYPE, AL_FILTER_HIGHPASS);
    ALFWprintf("'High Pass' Support      %s\n", (alGetError() == AL_NO_ERROR) ? "YES" : "NO");

    alFilteri(uiFilters[0], AL_FILTER_TYPE, AL_FILTER_BANDPASS);
    ALFWprintf("'Band Pass' Support      %s\n", (alGetError() == AL_NO_ERROR) ? "YES" : "NO");
}

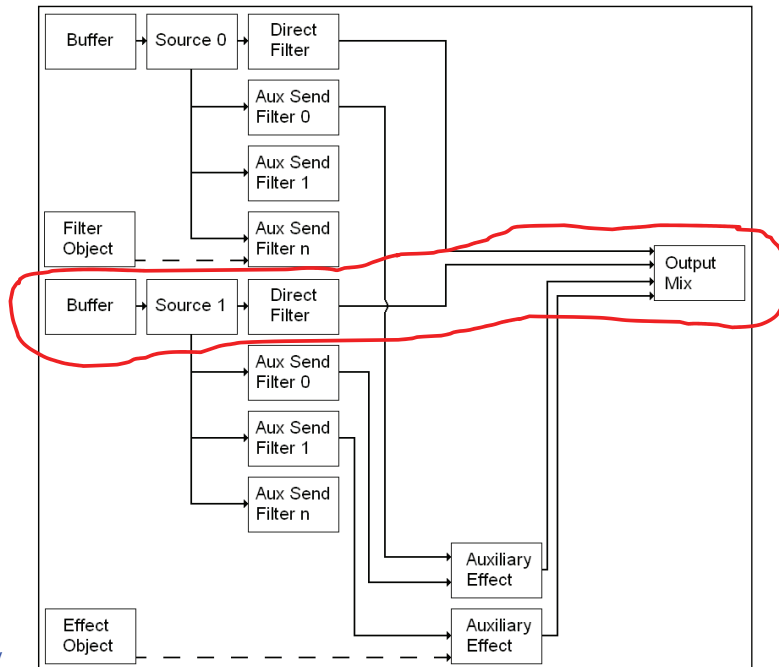
// Clean-up
alDeleteFilters(1, &uiFilters[0]); // Delete Filter
alDeleteEffects(1, &uiEffects[0]); // Delete Effect

// Delete Auxiliary Effect Slots
alDeleteAuxiliaryEffectSlots(iEffectSlotsGenerated, uiEffectSlots);
} else
ALFWprintf("EFX support not found\n");
```



## Part2 - EFX Filter Programming

- Filtering도 안 하고 Effect도 없는 재생: Dry Sound
- Wet Sound 기법 1: Direct Filter



### Part2 - EFX Filter Programming



## Direct Filter 예제 코드

```

ALvoid PlaySource(ALuint uiSource)
{
    ALint iState;

    // Play Source
    alSourcePlay(uiSource);
    do {
        alGetSourcei(uiSource, AL_SOURCE_STATE, &iState);
        Sleep(10);
    } while (iState == AL_PLAYING);
}

ALboolean PlayDry(ALuint uiSource)
{
    ALFWprintf("Source played dry\n");
    PlaySource(uiSource);

    return AL_TRUE;
}

```



## Direct Filter 예제 코드(계속)

```

ALboolean PlayDirectFilter(ALuint uiSource)
{
    ALuint uiFilter;

    ALFWprintf("Source played through a direct lowpass filter\n");

    if (AL_FALSE == CreateFilter( &uiFilter, AL_FILTER_LOWPASS, 1.0f, 0.5f))
        return AL_FALSE;

    // Assign filter to the source
    alSourcei(uiSource, AL_DIRECT_FILTER, uiFilter);
    if (alGetError() != AL_NO_ERROR)
        return AL_FALSE;

    PlaySource(uiSource);

    // Cleanup
    alSourcei(uiSource, AL_DIRECT_FILTER, AL_FILTER_NULL);
    alDeleteFilters( 1, &uiFilter );

    return AL_TRUE;
}

```



## Direct Filter 예제 코드(계속)

```

ALboolean
CreateFilter(ALuint *filter, ALenum filterType, ALfloat gain, ALfloat gainHF)
{
    alGetError(); // Clear AL Error

    alGenFilters(1, filter); // Create a Filter object
    if (alGetError() != AL_NO_ERROR)
        return AL_FALSE;

    if (alIsFilter((*filter))) { // 인자 filter가 Filter 객체를 가리키는지 확인
        // filter 유형을 인자로 들어온 filterType으로 설정
        alFilteri( (*filter), AL_FILTER_TYPE, filterType );
        if (alGetError() != AL_NO_ERROR)
            return AL_FALSE;
    }

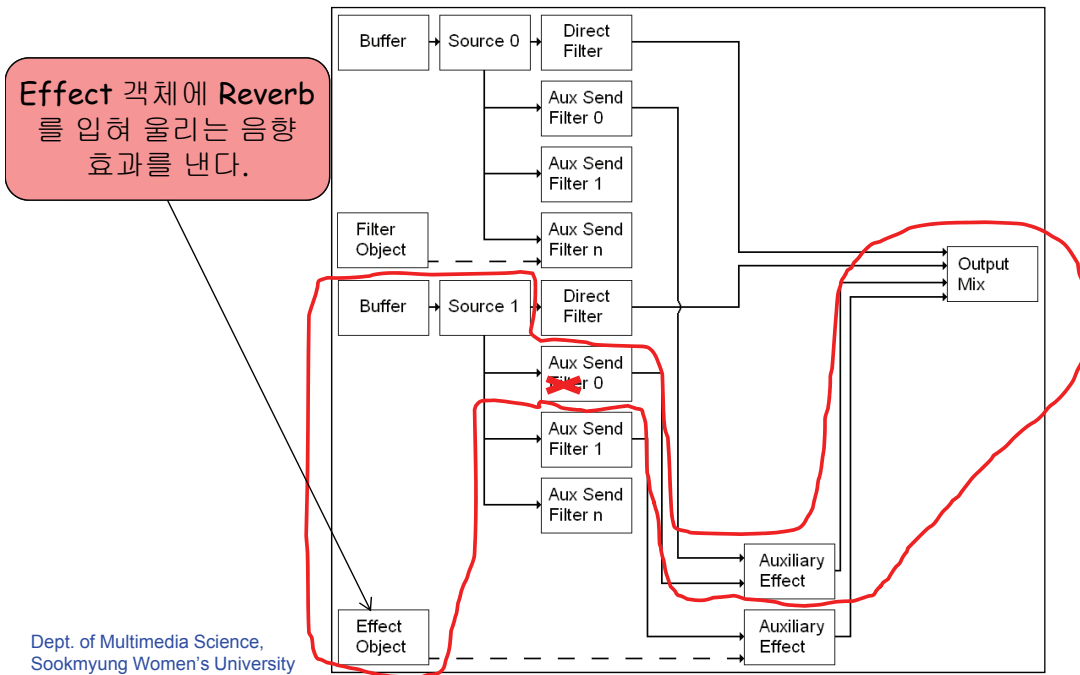
    alFilterf((*filter), AL_LOWPASS_GAIN, gain); // 저주파수 통과율
    alFilterf((*filter), AL_LOWPASS_GAINHF, gainHF); // 고주파수 통과율

    return AL_TRUE;
}

```

# Wet Sound 기법 2: Aux. Effect Slot without Filter

- Aux. Effect Slot 객체에 aux. send filter를 연결
  - 단, Filtering은 하지 않음



Dept. of Multimedia Science,  
Sookmyung Women's University



# Aux. Effect Slot without Filter 예제 코드

```

ALboolean PlayAuxiliaryNoFilter(ALuint uiSource)
{
    ALuint uiEffectSlot;
    ALuint uiEffect;

    ALFWprintf("Source played through an auxiliary reverb without filtering\n");

    // Create an Aux. Effect Slot object
    alGenAuxiliaryEffectSlots( 1, &uiEffectSlot );

    // Create an effect to be loaded in the auxiliary effect slot
    alGenEffects( 1, &uiEffect );

    if (alIsEffect(uiEffect)) { // uiEffect가 Effect 객체이면
        // Configure effect to be Reverb
        alEffecti(uiEffect, AL_EFFECT_TYPE, AL_EFFECT_REVERB);

        if (alIsAuxiliaryEffectSlot(uiEffectSlot)) // uiEffectSlot이 Aux. effect slot 이면
            // Load effect into slot
            alAuxiliaryEffectSloti(uiEffectSlot, AL_EFFECTSLOT_EFFECT, uiEffect);
    }
    .....
}
    
```



# Aux. Effect Slot without Filter

## 예제 코드 (계속)

```

.....
// Enable Send 0 from the Source to the Auxiliary Effect Slot without filtering
alSource3i(uiSource, AL_AUXILIARY_SEND_FILTER, uiEffectSlot, 0,
           AL_FILTER_NULL);
if (alGetError() != AL_NO_ERROR)
    return AL_FALSE;

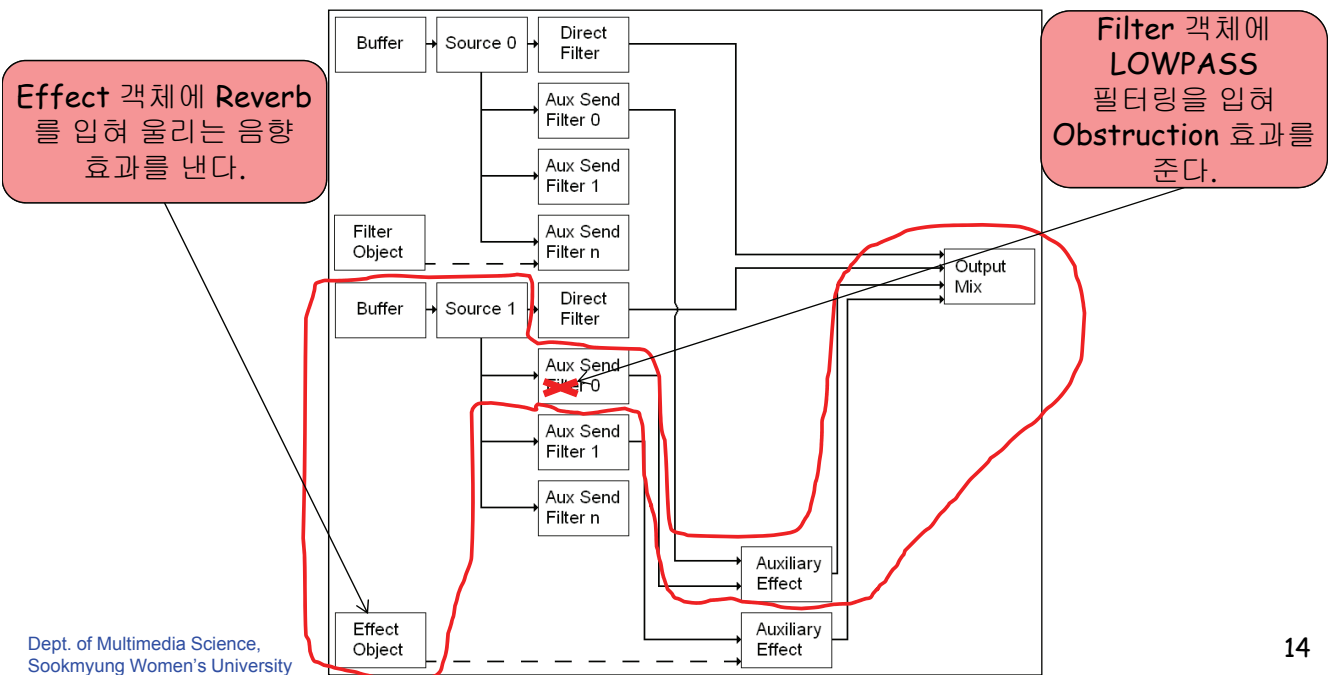
PlaySource(uiSource);

// Cleanup
alSource3i(uiSource, AL_AUXILIARY_SEND_FILTER, AL_EFFECTSLOT_NULL, 0,
           AL_FILTER_NULL);
alDeleteAuxiliaryEffectSlots(1, &uiEffectSlot);
alDeleteEffects(1, &uiEffect);

return AL_TRUE;
}
    
```

# Wet Sound 기법 3: Aux. Effect Slot with Filter

- Aux. Effect Slot 객체에 aux. send filter를 연결
  - 이번엔 Filtering까지 함





## Aux. Effect Slot with Filter

### 예제 코드

```

ALboolean PlayAuxiliaryFilter(ALuint uiSource)
{
    ALuint uiFilter;
    ALuint uiEffectSlot;
    ALuint uiEffect;

    ALFWprintf("Source played through an auxiliary reverb with lowpass filter\n");
    if( AL_FALSE == CreateFilter(&uiFilter, AL_FILTER_LOWPASS, 1.0f, 0.1f))
        return AL_FALSE;

    // Create an aux. effect slot object
    alGenAuxiliaryEffectSlots(1, &uiEffectSlot);

    // Create an effect to be loaded in the auxiliary effect slot
    alGenEffects(1, &uiEffect);

    // Configure effect to be Reverb
    if (alIsEffect(uiEffect)) {
        alEffecti(uiEffect, AL_EFFECT_TYPE, AL_EFFECT_REVERB);

        // Load effect into slot
        if (alIsAuxiliaryEffectSlot(uiEffectSlot))
            alAuxiliaryEffectSloti( uiEffectSlot, AL_EFFECTSLOT_EFFECT, uiEffect );
    }
    .....
}

```



## Aux. Effect Slot with Filter

### 예제 코드 (계속)

```

.....
// Enable Send 0 from the Source to the Auxiliary Effect Slot with filtering
alSource3i(uiSource, AL_AUXILIARY_SEND_FILTER, uiEffectSlot, 0, uiFilter);
if (alGetError() != AL_NO_ERROR)
    return AL_FALSE;

PlaySource(uiSource);

// Cleanup
alSource3i(uiSource, AL_AUXILIARY_SEND_FILTER, AL_EFFECTSLOT_NULL, 0,
           AL_FILTER_NULL);
alDeleteFilters( 1, &uiFilter );
alDeleteAuxiliaryEffectSlots( 1, &uiEffectSlot );
alDeleteEffects( 1, &uiEffect );

return AL_TRUE;
}

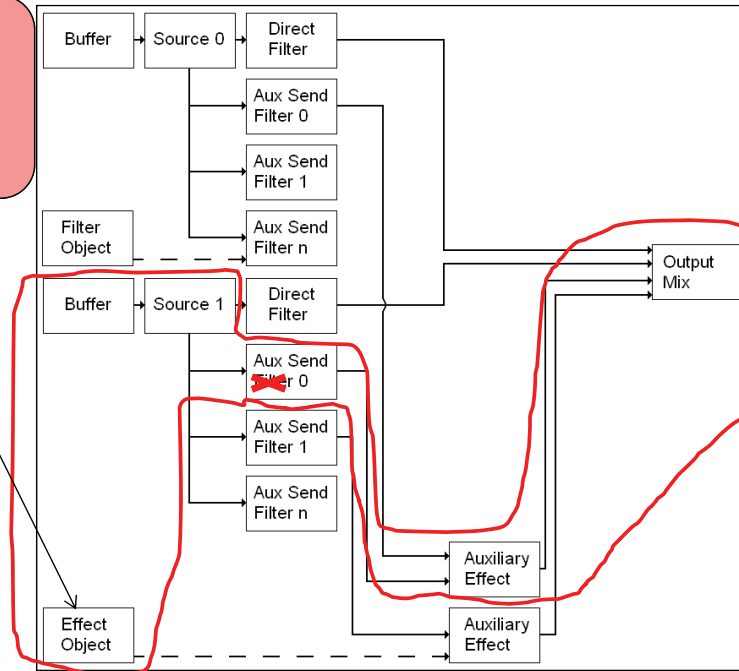
```



# Part3 - EFX EAXReverb Programming

- **EAX**: 대부분의 사운드 카드가 지원하는 음향 환경 효과
  - Environment **A**udio **E**xtension

Effect 객체에 EAXReverb 효과를 입혀 다양한 음향 환경을 시뮬레이션 한다.



Dept. of Multimedia Science,  
Sookmyung Women's University

17

## Part3 - EFX EAXReverb Programming



# EFX EAXReverb 예제 코드

```
#include "Framework.h"
#include "EFX-Util.h"
#include <math.h>

ALboolean CreateAuxEffectSlot(ALuint *puiAuxEffectSlot);
ALboolean CreateEffect(ALuint *puiEffect, AEnum eEffectType);
ALvoid PlaySource(ALuint uiSource);
ALboolean SetEFXEAXReverbProperties(EFXEAXREVERBPROPERTIES *pEFXEAXReverb,
                                   ALuint uiEffect);
int EFXReverbExamples(const char *filename);

EAXREVERBPROPERTIES eaxBathroom = REVERB_PRESET_BATHROOM; // Preset
EAXREVERBPROPERTIES eaxHangar = REVERB_PRESET_HANGAR; // Preset

int EFXReverbExamples(const char *filename)
{
    EFXEAXREVERBPROPERTIES efxReverb;
    ALuint uiEffectSlot, uiEffect;
    ALuint uiSource, uiBuffer;
    ALboolean bEffectCreated = AL_FALSE;

    alGetError(); // Clear AL Error State
    alGenBuffers(1, &uiBuffer); // Generate a Buffer
    .....
}
```

Dept. of Multimedia Science,  
Sookmyung Women's University

18



## EFX EAXReverb 예제 코드 (계속)

```

.....
if (!ALFWLoadWaveToBuffer((char*)ALFWaddMediaPath(filename), uiBuffer)) {
    ALFWprintf("Failed to load %s\n", ALFWaddMediaPath(filename));
    alDeleteBuffers(1, &uiBuffer);
    ALFWShutdownOpenAL();
    ALFWShutdown();
    return 0;
}
alGenSources(1, &uiSource); // Generate a Source
alSourcei(uiSource, AL_BUFFER, uiBuffer); // Attach Buffer to Source

if (ALFWIsEFXSupported()) { // EFX Extension 기능이 지원되는 지 검사
    // Reverb나 EAXReverb 같은 Effect를 주려면 Effect 객체를 담을
    // Auxiliary Effect Slot 객체를 먼저 생성해야 한다.
    if (CreateAuxEffectSlot(&uiEffectSlot)) {
        // Auxiliary Effect Slot 객체 생성에 성공했으면, 이제 Effect 객체를 만들고 Effect 객체의
        // 유형과 Parameter 값들을 설정한다. 그런 후 Effect 객체를 Auxiliary Effect Slot에 적재
        if (CreateEffect(&uiEffect, AL_EFFECT_EAXREVERB))
            bEffectCreated = AL_TRUE;
        else
            ALFWprintf("Failed to Create an EAX Reverb Effect\n");
    }
    else
        ALFWprintf("Failed to generate an Auxiliary Effect Slot\n");

    // Playing Source without EAX Effects
    PlayDry(uiSource);
}
.....

```



## EFX EAXReverb 예제 코드 (계속)

```

.....
if (bEffectCreated) {
    ALFWprintf("Playing Source with Send to 'Hangar' EAX Reverb Effect\n");

    // Reverb Preset 포맷의 Hangar 값들을 EFX EAX Reverb 패러미터 값으로 변환한다.
    ConvertReverbParameters(&eaxHangar, &efxReverb);

    // Set the Effect parameters
    if (!SetEFXEAXReverbProperties(&efxReverb, uiEffect))
        ALFWprintf("Failed to set Reverb Parameters\n");

    // Load Effect into Auxiliary Effect Slot
    alAuxiliaryEffectSloti(uiEffectSlot, AL_EFFECTSLOT_EFFECT, uiEffect);

    // Enable (non-filtered) Send 0 from Source to Auxiliary Effect Slot
    alSource3i(uiSource, AL_AUXILIARY_SEND_FILTER, uiEffectSlot, 0, AL_FILTER_NULL);

    PlaySource(uiSource); // Play Source again

    ALFWprintf("Playing Source with Send to 'Bathroom' EAX Reverb Effect\n");
    // 한번 설정된 EAX Reverb 효과를 바꾸려면, Effect 객체 내 EAX Reverb 효과값을 바꾼 후
    // Auxiliary Effect Slot 객체에 다시 붙여주어야 한다.
    // Reverb Preset 포맷의 Bathroom 값들을 EFX EAX Reverb 패러미터 값으로 변환한다.
    ConvertReverbParameters(&eaxBathroom, &efxReverb);

    // Set the Effect parameters
    if (!SetEFXEAXReverbProperties(&efxReverb, uiEffect))
        ALFWprintf("Failed to set Reverb Parameters\n");
}
.....

```



## EFX EAXReverb 예제 코드 (계속)

```

.....
// Reload Effect into Auxiliary Effect Slot
alAuxiliaryEffectSloti(uiEffectSlot, AL_EFFECTSLOT_EFFECT, uiEffect);

PlaySource(uiSource); // Play Source again

// Clean-up. Remove Effect Send from Source
alSource3i(uiSource, AL_AUXILIARY_SEND_FILTER, AL_EFFECTSLOT_NULL, 0,
           AL_FILTER_NULL);

// Load NULL Effect into Effect Slot
alAuxiliaryEffectSloti(uiEffectSlot, AL_EFFECTSLOT_EFFECT, AL_EFFECT_NULL);
alDeleteEffects(1, &uiEffect); // Delete Effect
alDeleteAuxiliaryEffectSlots(1, &uiEffectSlot); // Delete Auxiliary Effect Slot
}

// Delete Source and Buffer
alSourceStop(uiSource);
alDeleteSources(1, &uiSource);
alDeleteBuffers(1, &uiBuffer);
}
else
    ALFWprintf("AL Device does not support EFX\n");

return 0;
}

```



## EFX EAXReverb 예제 코드 (계속)

```

ALboolean CreateAuxEffectSlot(ALuint *puiAuxEffectSlot)
{
    ALboolean bReturn = AL_FALSE;

    alGetError(); // Clear AL Error state
    alGenAuxiliaryEffectSlots(1, puiAuxEffectSlot); // Generate an Auxiliary Effect Slot
    if (alGetError() == AL_NO_ERROR)
        bReturn = AL_TRUE;

    return bReturn;
}

ALboolean CreateEffect(ALuint *puiEffect, ALenum eEffectType)
{
    ALboolean bReturn = AL_FALSE;

    if (puiEffect) {
        alGetError(); // Clear AL Error State
        alGenEffects(1, puiEffect); // Generate an Effect
        if (alGetError() == AL_NO_ERROR) {
            alEffecti(*puiEffect, AL_EFFECT_TYPE, eEffectType); // Set the Effect Type
            if (alGetError() == AL_NO_ERROR)
                bReturn = AL_TRUE;
            else
                alDeleteEffects(1, puiEffect);
        }
    }

    return bReturn;
}

```



# EFX EAXReverb 예제 코드 (계속)

```

ALboolean SetEFXEAXReverbProperties(EFXEAXREVERBPROPERTIES *pEFXEAXReverb, ALuint uiEffect)
{
    ALboolean bReturn = AL_FALSE;

    if (pEFXEAXReverb) {
        alGetError(); // Clear AL Error code

        alEffectf(uiEffect, AL_EAXREVERB_DENSITY, pEFXEAXReverb->fIDensity);
        alEffectf(uiEffect, AL_EAXREVERB_DIFFUSION, pEFXEAXReverb->fIDiffusion);
        alEffectf(uiEffect, AL_EAXREVERB_GAIN, pEFXEAXReverb->fIGain);
        alEffectf(uiEffect, AL_EAXREVERB_GAINHF, pEFXEAXReverb->fIGainHF);
        alEffectf(uiEffect, AL_EAXREVERB_GAINLF, pEFXEAXReverb->fIGainLF);
        alEffectf(uiEffect, AL_EAXREVERB_DECAY_TIME, pEFXEAXReverb->fIDecayTime);
        alEffectf(uiEffect, AL_EAXREVERB_DECAY_HFRATIO, pEFXEAXReverb->fIDecayHFRatio);
        alEffectf(uiEffect, AL_EAXREVERB_DECAY_LFRATIO, pEFXEAXReverb->fIDecayLFRatio);
        alEffectf(uiEffect, AL_EAXREVERB_REFLECTIONS_GAIN, pEFXEAXReverb->fIReflectionsGain);
        alEffectf(uiEffect, AL_EAXREVERB_REFLECTIONS_DELAY, pEFXEAXReverb->fIReflectionsDelay);
        alEffectfv(uiEffect, AL_EAXREVERB_REFLECTIONS_PAN, pEFXEAXReverb->fIReflectionsPan);
        alEffectf(uiEffect, AL_EAXREVERB_LATE_REVERB_GAIN, pEFXEAXReverb->fILateReverbGain);
        alEffectf(uiEffect, AL_EAXREVERB_LATE_REVERB_DELAY, pEFXEAXReverb->fILateReverbDelay);
        alEffectfv(uiEffect, AL_EAXREVERB_LATE_REVERB_PAN, pEFXEAXReverb->fILateReverbPan);
        alEffectf(uiEffect, AL_EAXREVERB_ECHO_TIME, pEFXEAXReverb->fIEchoTime);
        alEffectf(uiEffect, AL_EAXREVERB_ECHO_DEPTH, pEFXEAXReverb->fIEchoDepth);
        alEffectf(uiEffect, AL_EAXREVERB_MODULATION_TIME, pEFXEAXReverb->fIModulationTime);
        alEffectf(uiEffect, AL_EAXREVERB_MODULATION_DEPTH, pEFXEAXReverb->fIModulationDepth);
        alEffectf(uiEffect, AL_EAXREVERB_AIR_ABSORPTION_GAINHF, pEFXEAXReverb->fIAirAbsorptionGainHF);
        alEffectf(uiEffect, AL_EAXREVERB_HFREFERENCE, pEFXEAXReverb->fIHFRreference);
        alEffectf(uiEffect, AL_EAXREVERB_LFREFERENCE, pEFXEAXReverb->fILFRreference);
        alEffectf(uiEffect, AL_EAXREVERB_ROOM_ROLLOFF_FACTOR, pEFXEAXReverb->fIRoomRolloffFactor);
        alEffecti(uiEffect, AL_EAXREVERB_DECAY_HFLIMIT, pEFXEAXReverb->iDecayHFLimit);

        if (alGetError() == AL_NO_ERROR)
            bReturn = AL_TRUE;
    }

    return bReturn;
}

```

## Sound Effects



# 10장 리포트

- 10장 예제 소스를 실행시키고 각자의 PC에서 음향 효과가 제대로 반영되는지 확인할 것!
  - 확인과정에서 LOWPASS의 패러미터들을 조절해가며 필터링 효과가 어떻게 달라지는지 확인할 것!
- 10장 예제에서는 EFXEAXReverb 음향 효과를 위해 REVERB\_PRESET\_BATHROOM과 REVERB\_PRESET\_HANGAR를 사용하고 있다. 이 둘 외의 다른 Preset에는 어떤 것들이 있는지 알아보고 모든 Preset의 효과가 어떻게 다른지 확인할 것!
  - 이 과정에서 각자 기말 프로젝트에 사용할 Preset에는 어떤 것이 적당한지 골라두시기 바랍니다.
  - 기말 프로젝트에서는 팀 당 최소 5개 이상의 Preset을 사용해야 함
    - Preset 값을 미세 조정하여 작품 분위기에 맞출 경우 가산점 있음.



# Q & A

