



# Media Programming

[afewhee@gmail.com](mailto:afewhee@gmail.com)





- 1. Direct Music
- 2. Direct Sound





## ● Music

- ◆ Music: 음을 재생할 수 있는 방법을 적은 디지털 악보
- ◆ Sound: 아날로그 음원을 디지털 신호로 변환한 것

## ● 라이브러리

- ◆ `#pragma comment(lib, "dxguid.lib")`
- ◆ `#pragma comment(lib, "ole32.lib")`
- ◆ `#pragma comment(lib, "advapi32.lib")`

## ● 헤더 파일

- ◆ `#include < dmusicc.h>`
- ◆ `#include < dmusici.h>`





- COM 초기화
  - if (FAILED(CoInitialize(NULL)))  
return -1;
- 로더 생성
  - ◆ if (FAILED(CoCreateInstance(CLSID\_DirectMusicLoader  
, NULL, CLSCTX\_INPROC,  
IID\_IDirectMusicLoader8, (void\*\*) &m\_pLoader)) )  
return -1;
- 로더 해제
  - ◆ DirectX의 모든 객체들은 COM을 상속. Release() 함수로 해제
  - ◆ m\_pLoader->Release();
- COM 해제
  - CoUninitialize();
- 찾기 폴더 설정
  - ◆ 유니코드 문자열로 변환: MultiByteToWideChar(...)
  - ◆ 찾기 폴더 설정: m\_pLoader->SetSearchDirectory(...)





# 1. Direct Music - Performance, Segment

- Performance: Music 연주기
  - ◆ 퍼포먼스 생성  
if (FAILED(CoCreateInstance(...)))  
return -1;
  - ◆ 퍼포먼스 오디오 경로 설정  
if(FAILED(m\_pPerform->InitAudio(...)))  
return -1;
- 연주악기 다운로드
  - ◆ 유니코드로 파일이름 변경  
MultiByteToWideChar(...)
  - ◆ 신서사이저에 대한 연주악기 다운로드  
m\_pSegment->Download(m\_pPerform);
- 재생  
m\_pSegment->SetStartPoint(0);  
m\_pPerform->PlaySegmentEx();
- 중간 멈춤  
if(S\_OK == m\_pPerform->IsPlaying(m\_pSegment, NULL))  
m\_pPerform->StopEx(m\_pSegment, 0, 0);





# 1. Direct Music - Audio Path

- 볼륨 등 정교한 조정, 삼차원 음원 등을 만들 때 반드시 오디오 패스 필요
- 퍼포먼스를 통해서 오디오 패스 생성  
if (FAILED(m\_pPerform->CreateStandardAudioPath(...)))  
return -1;
- 연주 시작 전 오디오 패스 연결  
m\_pSegment->SetStartPoint(0);  
m\_pPerform->PlaySegmentEx(m\_pSegment  
    , ...  
    , **m\_pAudioPath**);
- 볼륨 조절
  - ◆ 볼륨의 크기: 데시벨(dB). [-9600, 0]  
m\_pAudioPath->SetVolume(m\_dVolume, 0);



- Music Class

```
class DxMusic
{
    IDirectMusicPerformance8*    m_pPerform;
    IDirectMusicSegment8*       m_pSegment;

    void    Play();
    void    Stop();
    void    Reset();
    void    Pause();
};
```

- Music Manager

```
class CDxMusicManager
{
    static IDirectMusicLoader8*    m_pLoader ;

    INT    Init();
    void    Destroy();
    INT    Create(DxMusic** pMusic, char* sFile);
};
```





- 라이브러리/헤더파일
  - ◆ #include <mmsystem.h>
  - ◆ #pragma comment(lib, "winmm.lib")
- wave 파일 열기와 이벤트를 얻기 위한 callback 함수 연결

```
if(MMSYSERR_NOERROR != waveOutOpen(&m_WavOut, WAVE_MAPPER, &m_WavFMT,
(DWORD)CMpSnd::WaveOutProc, 0, CALLBACK_FUNCTION))
return -1;
```
- Wave Procedure 함수 예

```
void CALLBACK CMpSnd::WaveOutProc(HWAVEOUT, UINT uMsg, DWORD, DWORD, DWORD)
{
    switch(uMsg)
    {
        case WOM_OPEN:
            break;

        case WOM_DONE:
            break;

        case WOM_CLOSE:
            break;
    }
}
```







- DirectX SDK 에서 wave 파일의 저 수준 입출력을 편리하게 사용하기 위해 지원하는 객체
- Sound 지원 객체 내용
  - ◆ IDirectSound8
  - ◆ IDirectSoundBuffer8
  - ◆ IDirectSound3DBuffer8
  - ◆ IDirectSound3DListenr8
  - ◆ IDirectSoundCapture8
- 3D Sound 사용을 안 한하는 경우
  - ◆ IDirectSound8
  - ◆ IDirectSoundBuffer8 만 필요



## 2. Direct Sound

- 객체 생성 → 협력수준 설정 → 주 사운드 버퍼 설정
- 생성
  - ◆ IDirectSound 객체 생성  
if( FAILED(DirectSoundCreate8( NULL, &m\_pDS, NULL ) ) )  
return -1;
- 협력레벨 설정  
if( FAILED(m\_pDS->SetCooperativeLevel( m\_hWnd, DSSCL\_PRIORITY ) ) )  
return -1;
- 주 사운드 버퍼 설정  
if( FAILED( pDSBPrimary->SetFormat(&wfx) ) )  
return -1;
- 해제
  - ◆ Release()





## 2. Direct Sound-Wizard Modify

- DirectX SDK dsUtil.cpp, dsUtil.h 파일을 사용

- 라이브러리/ 헤더 파일

- ◆ #pragma comment(lib, "dxerr9.lib")
- ◆ #pragma comment(lib, "dxguid.lib")
- ◆ #pragma comment(lib, "dsound.lib")
- ◆ #pragma comment(lib, "winmm.lib")
- ◆ #include <windows.h>
- ◆ #include <mmsystem.h>
- ◆ #include <dsound.h>
- ◆ #include "dsutil.h"

- 볼륨 조절

- ◆ CSoundManager::Create ... 부분에  
dwCreationFlags |= DSBCAPS\_CTRLVOLUME; 을 설정

- {Set|Get} Volume 추가

```
void CDxSound::SetVolume(LONG IVal)
{
    for( DWORD i=0; i< m_dwNumBuffers; ++i)
    {
        if( m_apDSBuffer[i] )
            m_apDSBuffer[i]->SetVolume( IVal);
    }
}
```

```
LONG CDxSound::GetVolume()
{
    LONG IVal=0;
    if(m_apDSBuffer)
        m_apDSBuffer[0]->GetVolume(&IVal);
    return IVal;
}
```

