



CMP105 Games Programming

Audio



This week



- Audio
 - Sound vs Music
 - Basic audio setup
 - Improved audio setup
- Coursework
 - Q&A
- Feedback

Audio



- Creating a living, breathing game atmosphere can rarely, if ever, be only down to visuals
- Audio is handled by two sub-systems
 - Sound and Music
- We don't want to load every audio file into memory
- Sound is used to store small sound bytes/effects
 - Jump, explosions, walking, etc
- Music is used to stream large audio files
 - Background music, intro audio, etc

Sound (simple example)



```
sf::SoundBuffer buffer;  
buffer.loadFromFile("sound.wav");
```

```
sf::Sound sound;  
sound.setBuffer(buffer);  
sound.play();
```

Music (simple example)



```
sf::Music music;  
music.openFromFile("music.ogg");  
music.play();
```



Controlling audio



- SFML provides a collection of functions to control audio
- Playback
 - Play
 - Pause
 - Stop
- Set pitch
 - Default is 1
 - Greater value the higher the pitch, and vice versa
- Set the volume
 - Default 100
- Set looping
 - Does the audio file automatically loop
 - Useful for background audio

Controls



```
// start playback  
sound.play();
```

```
// advance to 2 seconds  
sound.setPlayingOffset(sf::seconds(2));
```

```
// pause playback  
sound.pause();
```

```
// resume playback  
sound.play();
```

```
// stop playback and rewind  
sound.stop();
```

```
// Set pitch a little higher  
sound.setPitch(1.2);
```

```
// Set volume to 50%  
sound.setVolume(50);
```

```
// Set audio to automatically loop  
sound.setLoop(true);
```

Supported formats



- Both sound and music support a range of typical formats
 - WAV
 - 8, 16 and 24 bit
 - FLAC
 - 8, 16 and 24 bit
 - OGG
 - Ogg Vorbis audio files

Limitations and challenges



- No .mp3 support
- `Sf::SoundBuffer` is like `sf::Texture`
 - Must remain in **scope** or the audio will be deleted
- `Sf::Music`
 - Can **not** be copied
 - Make it difficult to pass large music files from object to object

Scale



- Working directly with Sound/Music is fine for a small number of audio files
 - But will get unmanageable fast as number of audio files grows
- Solution
 - Build a manager class for the audio files
 - Similar to that used of handling large number of sprites/spawnables
 - Similar approach could be used for textures and other resources

Audio manager



- Handles loading, storing and playback of audio files
 - Contains both **Sound** and **Music**
- Maintains two lists for audio (sound/music)
- Provides functions for
 - Adding sound and music
 - Playback based on name/key
 - Stopping all sound/music
- Could be extended to handle more complex playback control
 - Pause, pitch, volume control, etc

Audio manager example



- Simple example
 - Add a few sound effects
 - Playback sounds based on key presses
 - Add a few music tracks
 - Playback start automatically
 - Track can be swapped on key press
 - Music can be stopped on key press

Audio manager example



- In game.cpp at files to audio manager

```
audioMgr.addMusic("sfx/cantina.wav", "cantina");  
audioMgr.addMusic("sfx/hyrulefield.wav", "hyrule");  
  
audioMgr.addSound("sfx/SMB_jump-small.wav", "jump");  
audioMgr.addSound("sfx/SMB_1-up.wav", "up");  
audioMgr.addSound("sfx/getover.wav", "getover");  
audioMgr.addSound("sfx/TP_Secret.wav", "secret");
```

Audio manager example



- Controlling audio playback

```
void Game::update(float dt)
{
    if (!hasStarted)
    {
        audioMgr.playMusicByName("cantina");
        hasStarted = true;
    }
}
```

Audio manager example



- Play sound effects

```
if (input->isKeyDown(sf::Keyboard::Num1))
{
    input->setKeyUp(sf::Keyboard::Num1);
    audioMgr.playSoundbyName("up");
}
```

```
if (input->isKeyDown(sf::Keyboard::Num2))
{
    input->setKeyUp(sf::Keyboard::Num2);
    audioMgr.playSoundbyName("getover");
}
```

Audio manager example



- Control background music

```
if (input->isKeyDown(sf::Keyboard::BackSpace))
{
    input->setKeyUp(sf::Keyboard::BackSpace);
    audioMgr.stopAllMusic();
}
```

```
if (input->isKeyDown(sf::Keyboard::Num4))
{
    input->setKeyUp(sf::Keyboard::Num4);
    audioMgr.playMusicbyName("hyrule");
}
```


Live demo



- Does the audio play over HDMI...



Some audio fun



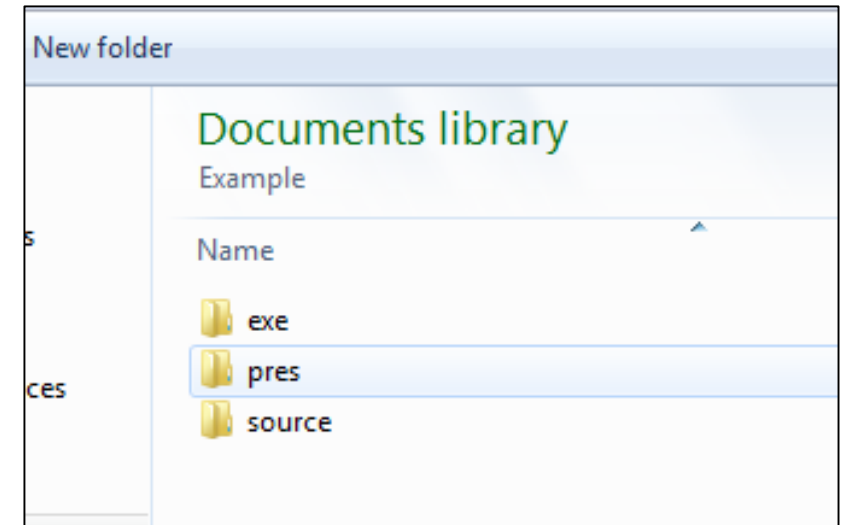
- <https://www.youtube.com/watch?v=jwxN8sCIOOE>



Coursework



- Important dates
 - Due date: Tuesday 25th April 2017
 - Feedback date: 16th May 2017
- Submission process
 - Need to submit
 - Executable version of your game
 - The source (the whole project)
 - Presentation files
 - All in a zip file 123456_SmithJohn.zip

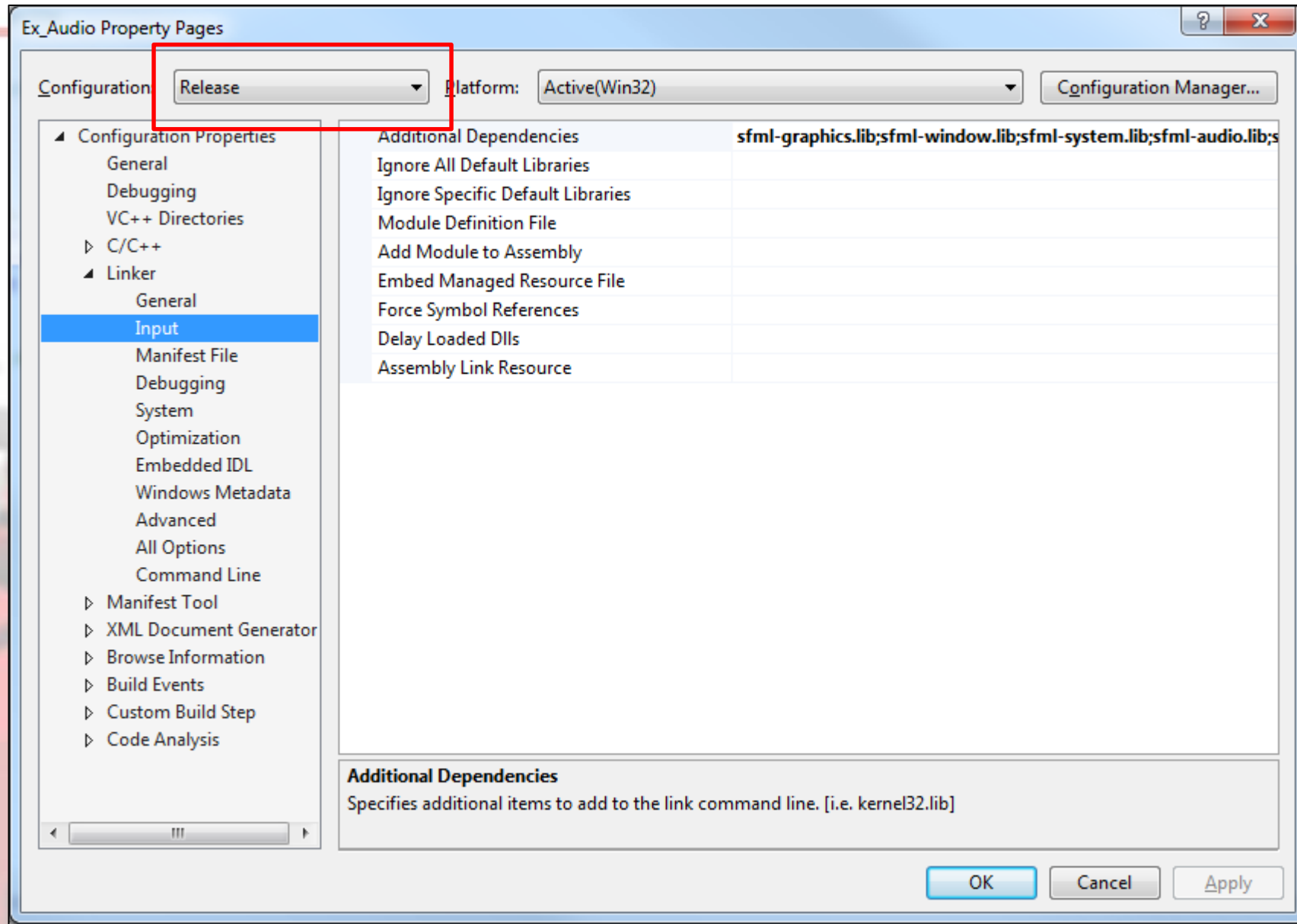


Release build



- Setup libraries
 - Currently we have libraries setup for debug, not for release
- Copy .dlls
 - Need to put release .dlls file in the correct place
- Copy content
 - Game content needs to be copied into release folder
 - Maintaining folder structure
 - Fonts, gfx and sfx

Libraries



Libraries

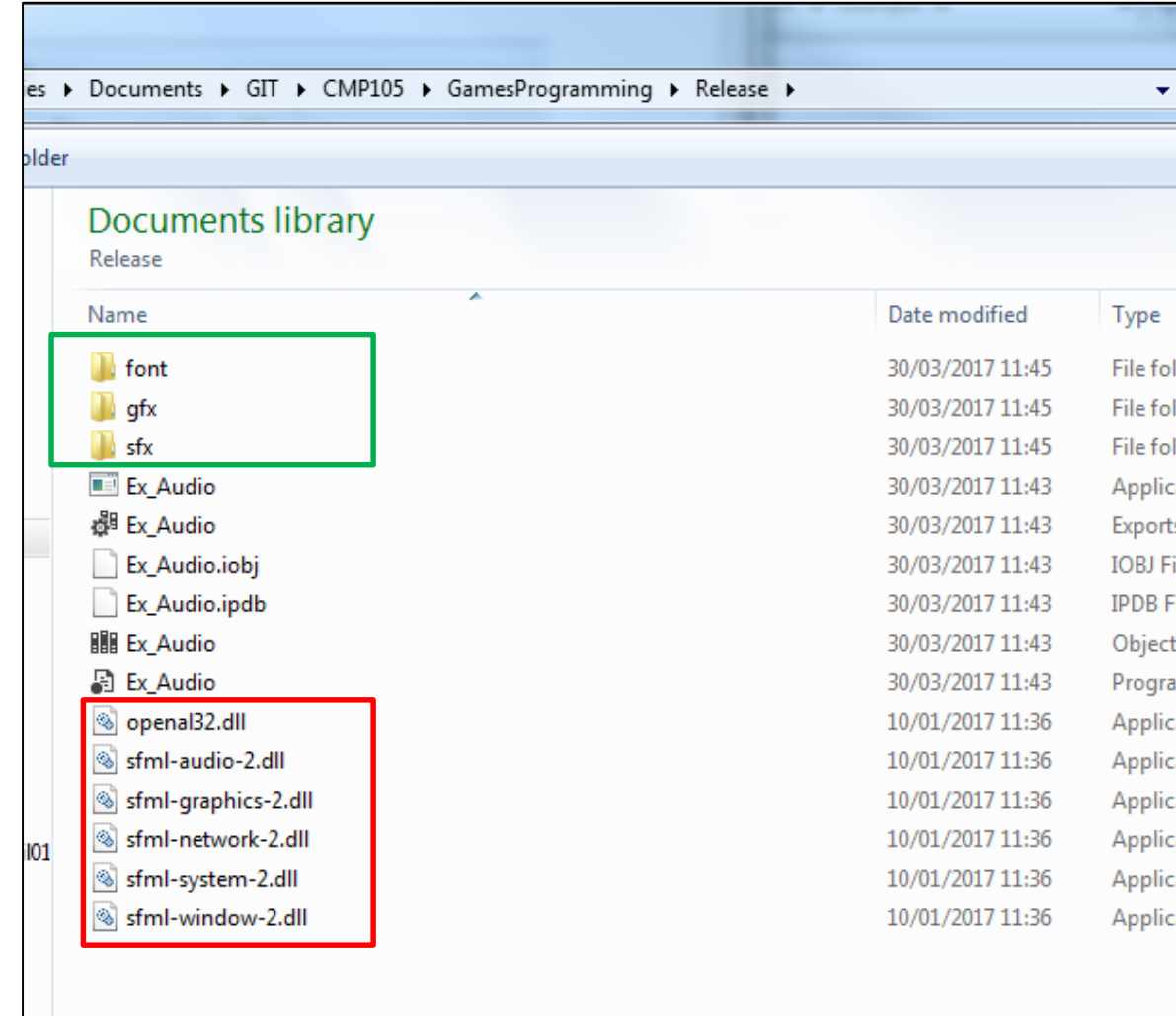


- Include folder, same as debug
 - \$(SolutionDir)/SFML-2.4.0/include
- Library folder, same as debug
 - \$(SolutionDir)/SFML-2.4.0/lib
- Additional dependencies, different
 - sfml-graphics.lib
 - sfml-window.lib
 - sfml-system.lib
 - sfml-audio.lib
 - sfml-main.lib
- Debug versions have a "-d" suffix

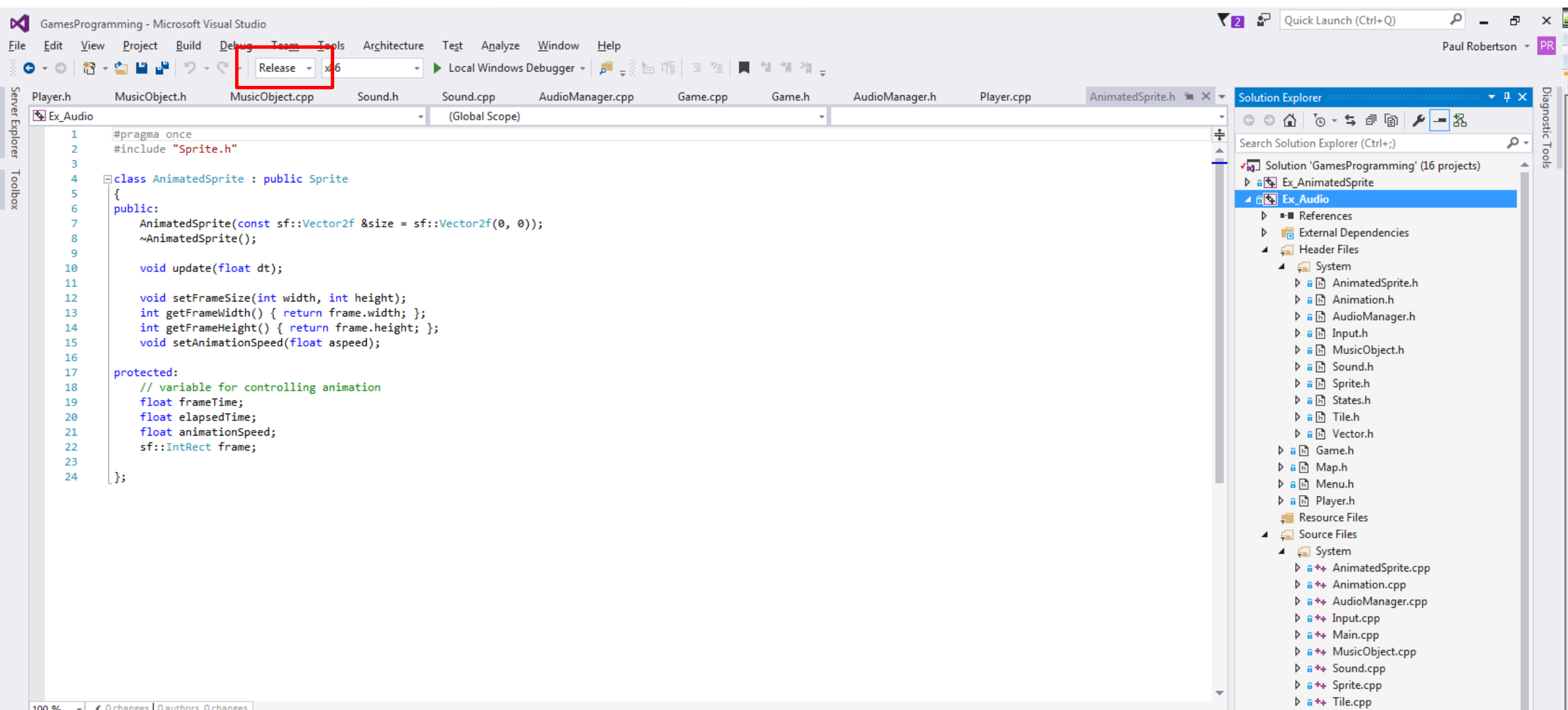
Copy .dll files



- Previous copied .dll into the debug folder
- Now copy the release libraries into the release folder
 - Release folder won't exist until you build a release version of the application
- Copy game content to same location
 - Keeping the folder structure
- Test it!



Building release



Feedback / Module survey



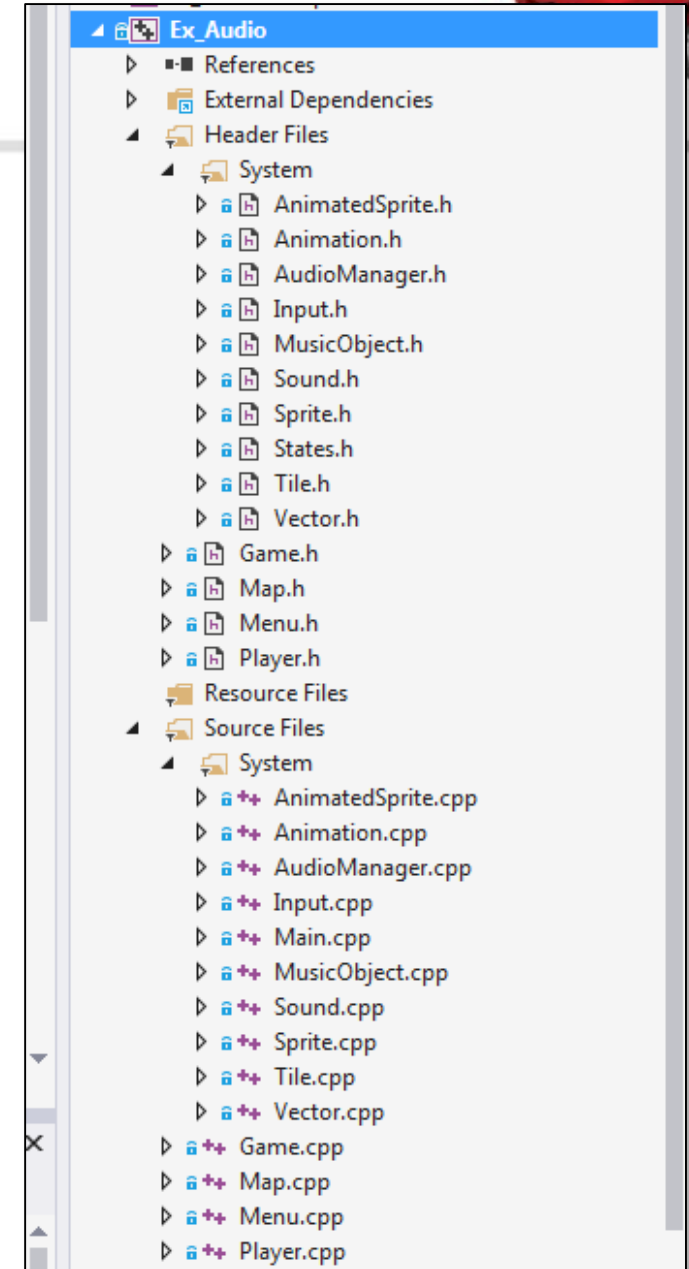
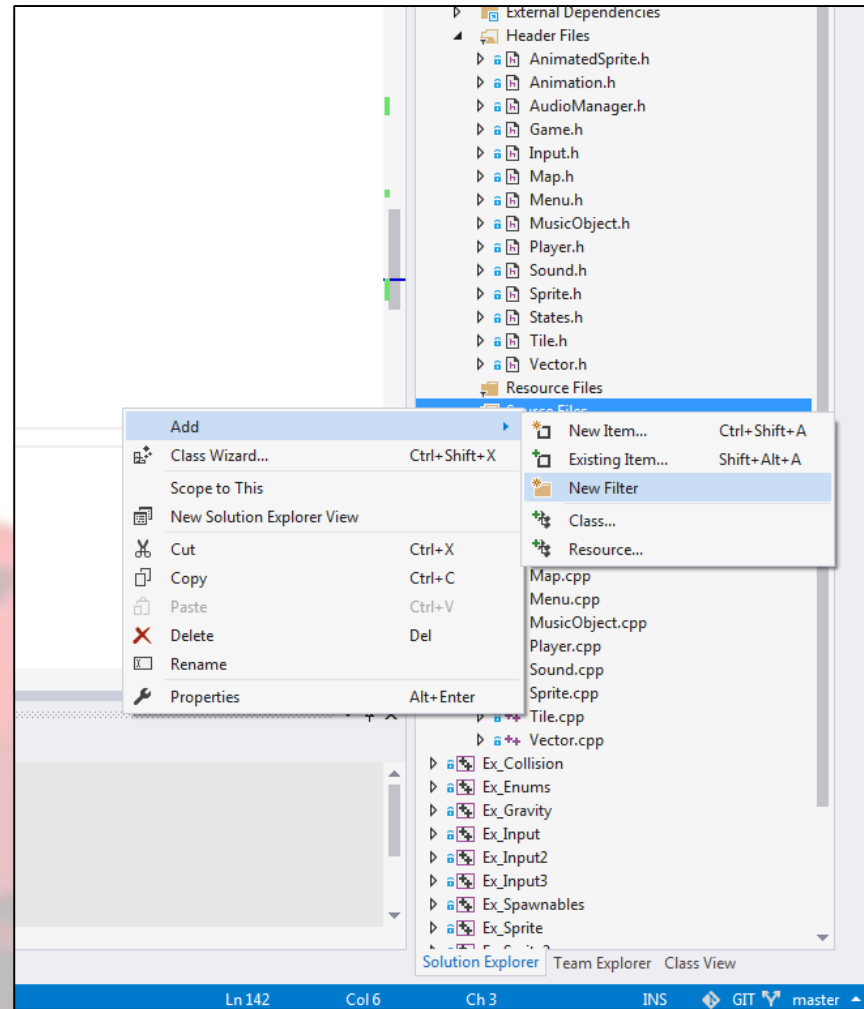
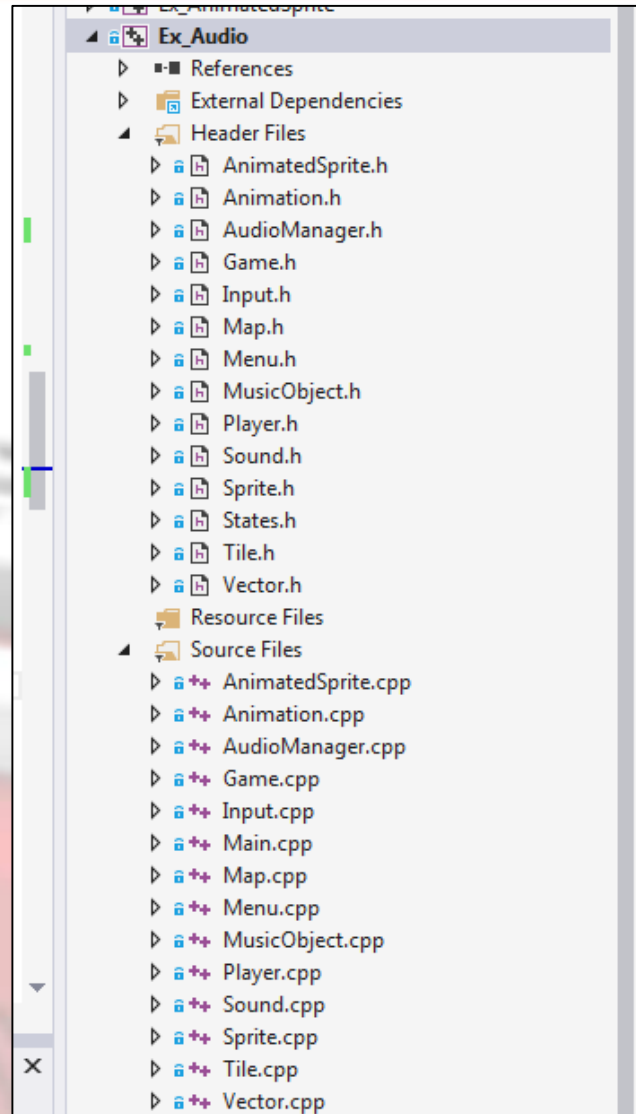
- Very good response rate (over 50)
 - Best I've had
- A large number of you seem very happy with the module, delivery and content
- No I will not DAB at the end of lectures
 - Never!

Feedback / Module survey

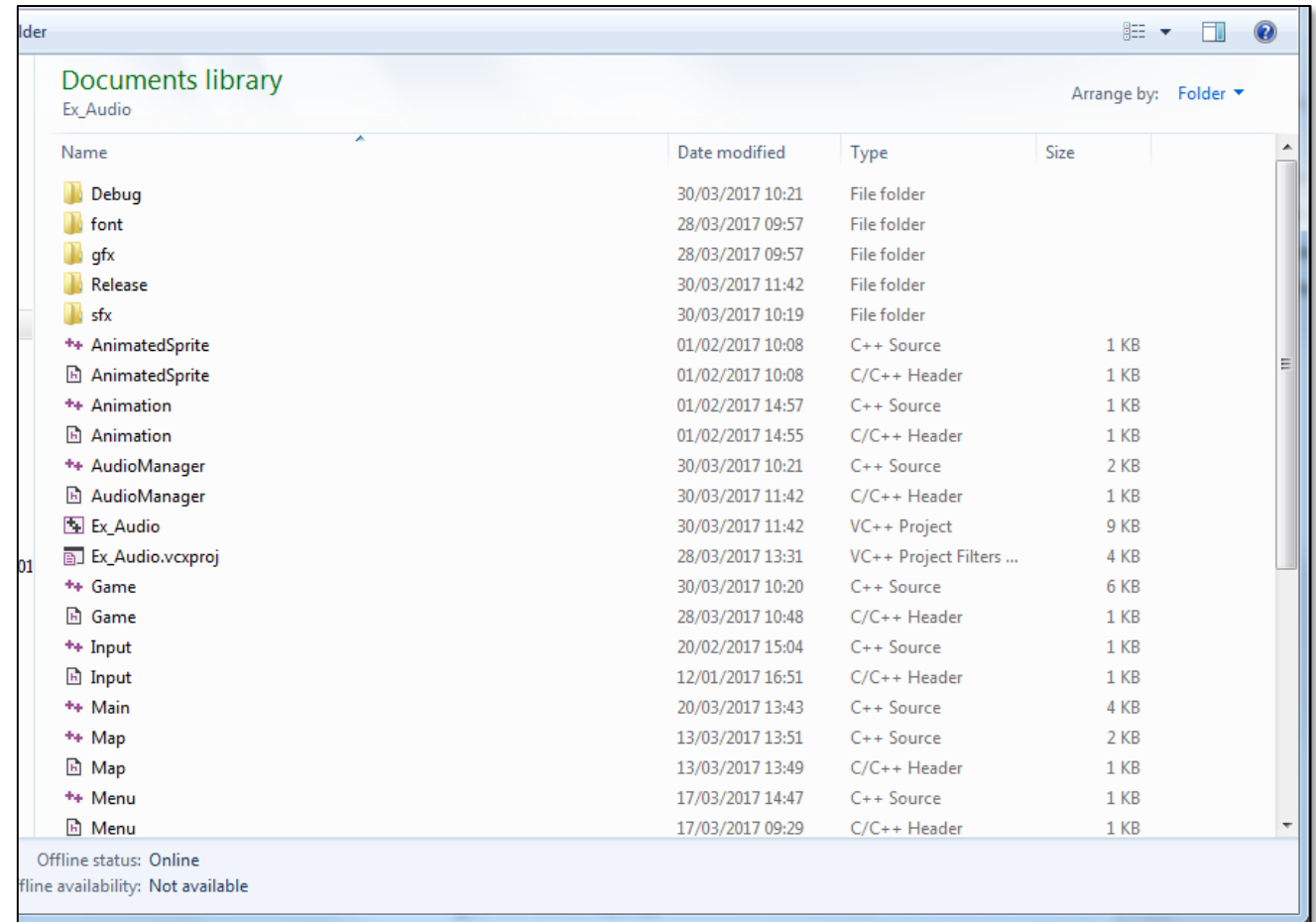
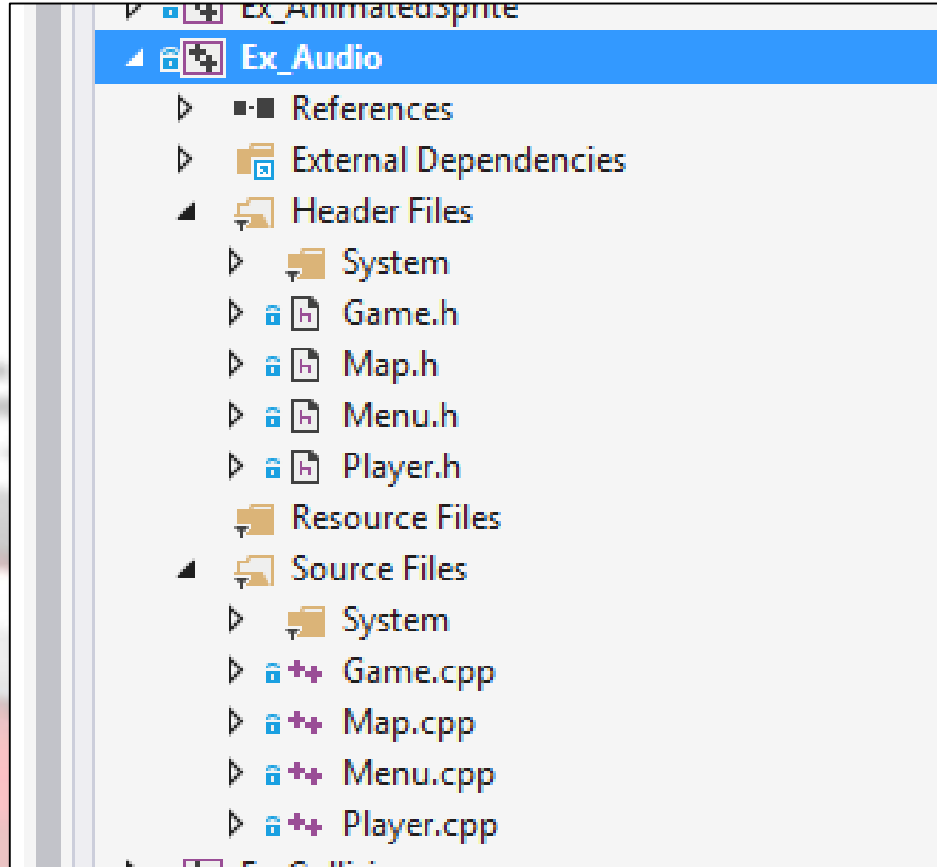


- A few feel the module is:
 - Too fast
 - Too slow
 - Too easy
 - Too complex
- No books on SFML in the library
 - There is now
- Some lab tasks are a little confusing (building classes etc)
 - I will be fixing that
- Too many files in visual studio!

Folder structure in Visual Studio



Folder structure in Visual Studio



In the labs



- Adding the audio manager to the project
- Working with audio
- **BRING HEADPHONES!**
- This was the last lecture
- Labs will run this week and next week
- Words of wisdom
 - Do some programming over the summer!