SFML and Visual Studio

Contents

[Introduction 2](#_Toc414529696)

[Download and Extract 2](#_Toc414529697)

[Set Up The Project 2](#_Toc414529698)

[Specify The Location of the SFML Include Folder 2](#_Toc414529699)

[Specify The Location of the SFML Library Folder 2](#_Toc414529700)

[Specify the Static Linking Preprocessor Directive 3](#_Toc414529701)

[Statically Link Parts of the SFML Library 3](#_Toc414529702)

[Link the Libraries that SFML Depends On 3](#_Toc414529703)

[Table of SFML Dependencies 4](#_Toc414529704)

[Code 5](#_Toc414529705)

[Make A Project Template 6](#_Toc414529706)

# Introduction

The following instructions describe how to create SFML libraries for use with Visual Studio 2010 in Debug mode using static linking.

The steps may be trivially modified to create SFML libraries to be dynamically linked.

Follow the online tutorials and make some amazing 2D games.

# Download and Extract

1. Download the latest stable version of SFML
2. Extract in Documents.
3. You should now have a folder entitled ‘SFML-X.X’ in Documents.

# Set Up The Project

Create a new C++ project of type ‘Win32 Console Application’

Name it ‘SFML Project Template’

Application Window: Choose ‘Empty Project’

Solution Explorer > Source Files: add a ‘main.cpp’ file

# Specify The Location of the SFML Include Folder

Project > (name of project) Properties > C/C++ > General > Additional Include Directories >

(full path to Documents/SFML/include)

Configuration: All Configurations

# Specify The Location of the SFML Library Folder

Project > (name of project) Properties > Linker > General > Additional Library Directories >

(full path to Documents/SFML/lib)

Configuration: All Configurations

# Specify the Static Linking Preprocessor Directive

Project > (name of project) Properties > C/C++ > Preprocessor > Preprocessor Definitions > SFML\_STATIC

# Statically Link Parts of the SFML Library

Project > (name of project) Properties > Linker > Input > Additional Dependencies

sfml-system-s-d.lib

sfml-window-s-d.lib

sfml-graphics-s-d.lib

If you use other parts of the SFML Library, link those as well

e.g. audio will require sfml-audio-s-d.lib

# Link the Libraries that SFML Depends On

Project > (name of project) Properties > Linker > Input > Additional Dependencies

winmm.lib

opengl32.lib

gdi32.lib

freetype.lib

jpeg.lib

# Table of SFML Dependencies

| **Module** | **Dependencies** |
| --- | --- |
| sfml-system-s.lib | * winmm.lib |
| sfml-network-s.lib | * sfml-system-s.lib * ws2\_32.lib |
| sfml-audio-s.lib | * sfml-system-s.lib * sndfile.lib * openal32.lib |
| sfml-window-s.lib | * sfml-system-s.lib * opengl32.lib * gdi32.lib * winmm.lib |
| sfml-graphics-s.lib | * sfml-window-s.lib * sfml-system-s.lib * glew.lib * freetype.lib * jpeg.lib * opengl32.lib |

# Code

Copy the following code to the main.cpp file:

#include <SFML/Graphics.hpp>

int main()

{

sf::RenderWindow window(sf::VideoMode(640, 480), “Hello SFML World!”);

sf::CircleShape circle(100.0f);

while(window.isOpen())

{

sf::Event event;

while(window.pollEvent(event))

{

if(event.type == sf::Event::Closed)

window.close();

}

window.clear();

window.draw(circle);

window.display();

}

return EXIT\_SUCCESS;

}

When pasting into Visual Studio, it is necessary to delete the double quotes around “Hello SFML World!” and type them in again.

# Make A Project Template

Save this project and solution.

For every new SFML project, make a copy of the ‘SFML Project Template’ folder and rename it.

Use this for the new project.

Once this copy has been opened in Visual Studio, rename the project and other files as appropriate.

This alleviates the need to perform the above steps for every new project.