## SFML Installation and Usage on Microsoft Visual Studio

These instructions apply to MS Visual Studio Enterprise 2017. They should also be *close* to some earlier versions of MSVS as well. These instructions are somewhat detailed and apply to particular configurations. They are not very efficient, but hopefully, with a few prayers, they should work.

## Download and Installation

https://www.sfml-dev.org/download/sfml/2.5.1/

- Visual C++ 15 (2017) 32 bit (select Download | 16.3 MB)
   File: SFML-2.5.1-windows-vc15-32-bit.zip
- Extract into a directory where you can find it. The extraction directory name will be SFML-2.5.1

## Building an SFML Project

- Create an empty project.
- Add in or create new source code for your project
- Identify the include directories

ws Property Pages		? ×
Configuration: Debug	<ul> <li>Platform: Win32</li> </ul>	<ul> <li>✓ Configuration Manager</li> </ul>
<ul> <li>✓ Configuration Properties General Debugging VC++ Directories</li> <li>▶ C/C++</li> <li>▶ Linker</li> <li>▶ Manifest Tool</li> <li>▷ XML Document Generator</li> <li>▷ Browse Information</li> <li>▷ Build Events</li> <li>▷ Custom Build Step</li> <li>▷ Code Analysis</li> </ul>	Additional Include Directories         Additional #using Directories         Debug Information Format         Common Language RunTime Support         Consume Windows Runtime Extension         Suppress Startup Banner         Warning Level         Treat Warnings As Errors         Warning Version         Diagnostics Format         SDL checks         Multi-processor Compilation	C:\Software\SFML-2.5.1\include;C:\temp\wandering_star\include;%(Add Program Database for Edit And Continue (/ZI) Yes (/nologo) Level3 (/W3) No (/WX-) Classic (/diagnostics:classic) Yes (/sdl) Tempt reports with semi-sclears if more than one(/[fasth])
< >>		OK Cancel Apply

• Add the library directories, including the lib directory in the SFML directory.

ws Property Pages ? X						
Configuration: Debug	V Platform: Win32	✓ Configuration Manager				
General Optimization Preprocessor Code Generation Language Precompiled Heade Output Files Browse Information Advanced All Options Command Line Linker General Input Manifest File Debugging System Optimization Embedded IDL Windows Metadata Advanced All Options	Output File Show Progress Version Enable Incremental Linking Suppress Startup Banner Ignore Import Library Register Output Per-user Redirection Additional Library Directories Link Library Dependencies Use Library Dependencies Use Library Dependency Inputs Link Status Prevent DII Binding Treat Linker Warning As Errors Force File Output Create Hot Patchable Image Specify Section Attributes	\$(OutDir)\$(TargetName)\$(TargetExt) Not Set Yes (/INCREMENTAL) Yes (/NOLOGO) No No C:\Software\SFML-2.5.1\ib;%(AdditionalLibraryDirectories) Yes No				
Manifest Tool ✓	Allows the user to override the environmental library path. (/LIBPATH:folder)					
		OK Cancel Apply				

Before you add in the specific libraries you should specify the Configuration type. We will approach it for first, *All Configurations*, then for *Debug* configuration.

• Select *All Configurations* at the top of the *Configure Properties* window.

ws Property Pages ? X					
Configuration: All Configurations	Platform: Win32	✓ Configuration Manager			
Configuration Properties	Output File	\$(OutDir)\$(TargetName)\$(TargetExt)			
General	Show Progress	Not Set			
Debugging	Version				
VC++ Directories	Enable Incremental Linking	<different options=""></different>			
▷ C/C++	Suppress Startup Banner	Yes (/NOLOGO)			
▲ Linker	Ignore Import Library	No			
General	Register Output	No			
Input	Per-user Redirection	No			
Manifest File	Additional Library Directories	<different options=""></different>			
Debugging	Link Library Dependencies	Yes			
Ontimination	Use Library Dependency Inputs	No			
Embedded IDI	Link Status				
Windows Metadata	Prevent DII Binding				
Advanced	Treat Linker Warning As Errors				
All Options	Force File Output				
Command Line	Create Hot Patchable Image				
Manifest Tool	Specify Section Attributes				
XML Document Generator					
Browse Information					
Build Events					
Custom Build Step					
Code Analysis     Uture the     The Code Analysis     The Code Analysis					
	The 7001 option overrides the default name and loca	ation of the program that the linker creates.			
× /					
		OK Cancel <u>Apply</u>			

• Now, add the libraries to link in for *All Configurations*, including the sfml libraries.

ws Property Pages ? X					
<u>C</u> onfiguratio	n: All Configurations	✓ <u>P</u> latform:	Win32	~	Configuration Manager
✓ Configu Gen Deb VC+ ▷ C/C ✓ Link ( 1 1 1 5 5 6 6 6 8 8 8 8 9 8 9 9 9 9 9 9 9 9 9 9 9	uration Properties eral ugging + Directories ++ cer General Input Manifest File Debugging System Optimization Embedded IDL Windows Metadata Advanced All Options Command Line nifest Tool L Document Generator wse Information d Events toom Build Step le Analysis	Additional Dependencies           Ignore All Default Librarie           Ignore Specific Default Lib           Module Definition File           Add Module to Assembly           Embed Managed Resource           Force Symbol References           Delay Loaded Dlls           Assembly Link Resource	s oraries e File	sfml-audio.lib;sfml-graphics.lib;sfml-window	ı.lib;sfml-system.lib;%(Adq ∨
<	>	Specifies additional items to a	dd to the link command	line. [i.e. kernels2.lib]	Cancel <u>A</u> pply

And now, for the *Debug* configuration ...

• Select **Debug** at the top of the Configure Properties window.

ws Property Pages		? ×
Configuration: Debug	Platform: Win32	✓ Configuration Manager
✓ Configuration Properties General Debugging VC++ Directories ▷ C/C++ ✓ Linker General Input Manifest File Debugging System Optimization Embedded IDL Windows Metadata Advanced	Additional Dependencies Ignore All Default Libraries Ignore Specific Default Libraries Module Definition File Add Module to Assembly Embed Managed Resource File Force Symbol References Delay Loaded DIIs Assembly Link Resource	sfml-audio.lib;sfml-graphics.lib;sfml-window.lib;sfml-system.lib;%(Additic
Command Line Manifest Tool XML Document Generator Browse Information Build Events Custom Build Step Code Analysis	<b>Additional Dependencies</b> Specifies additional items to add to the link command	d line. [i.e. kernel32.lib] OK Cancel <u>A</u> pply

• Now, add the libraries to link in for the **Debug** configuration

ws Property Page	es					? ×
<u>C</u> onfiguration:	Debug	~	<u>P</u> latform:	Win32	~	Configuration Manager
Configuration Properties		Additional Dep	endencies		sfml-audio-d.lib;sfml-graphics-d.lib;sfml-wi	ndow-d.lib;sfml-system-d.l 🗸
General		Ignore All Defa	ult Libraries			
Debugg	ging	Ignore Specific	Default Libr	raries		
VC++ D	Directories	Module Definit	ion File			
▷ C/C++		Add Module to	Assembly			
Linker	▲ Linker		ed Resource	File		
Gen	eral	Force Symbol F	References			
Inpu	ut	Delay Loaded [	Olls			
Mar	nifest File	Assembly Link	Resource			
Deb	ugging					
Syst	iem					
Opt	imization					
Win	dows Matadata					
Ache	anced					
All (	Ontions					
Con	nmand Line					
Manifes	st Tool					
▷ XML Do	ocument Generator					
Browse	Information					
Build Ev	/ents					
Custom	n Build Step					
Code A	nalysis	Additional Depen	dencies			
		Specifies additiona	l items to ac	dd to the link command	d line. [i.e. kernel32.lib]	
<	>					
					ОК	Cancel <u>A</u> pply

Your project should now compile (and link).

It is important that the location your resource files (fonts, images, sound files, and other input files) be referenced in the source code or that they be placed in the directory where the application executes.

The run-time libraries need to be made available to the application. These libraries are located in the **bin** directory of your SFML directory (that may be SFML-2.5.1).

• Copy all of the files in the SFML bin directory into the execution directory of your application. The names of the run-time files are:

openal32.dll sfml-graphics-d-2.dll sfml-system-d-2.dll sfml-audio-2.dll sfml-network-2.dll sfml-window-2.dll sfml-audio-d-2.dll sfml-network-d-2.dll sfml-graphics-2.dll sfml-system-2.dll

The execution directory should be located in the same folder where you created your project. Look for a Debug fold containing files *projectname*.exe, *projectname*.ilk, and *projectname*.pdb.

If you're lucky, your application will run now.