

SFML Intro

Yu-Hsuan Chen 2018 May

Outline

- SFML Introduction
- SFML + Visual Studio
 - Visual Studio 2010 + SFML 2.3.2
 - Visual Studio 2017 + SFML 2.4.2 + CMake
- Your First SFML Program (施工中)
- Block Game with SFML (施工中)



- SFML = simple and fast multimedia library
- 跨平台、相容多語言的圖形介面
 主要為C++、.NET,社群亦有Java與Python的版本
- <u>https://sfml-dev.org</u>
- 目前最新的版本:2.5.0(2018/5) 相容Visual Studio 2017

SFML 2.3.2 X Visual Studio 2010

• 官方載點: <u>https://www.sfml-dev.org/download/</u>

sfml/2.3.2/

Download SFML 2.3.2

Windows	Visual C++ 10 (2010) - 32-bit	Download 11.9 MB	Visual C++ 10 (2010) - 64-bit	Download 13.2 MB
	Visual C++ 11 (2012) - 32-bit	Download 13.4 MB	Visual C++ 11 (2012) - 64-bit	Download 15.0 MB
	Visual C++ 12 (2013) - 32-bit	Download 12.8 MB	Visual C++ 12 (2013) - 64-bit	Download 14.3 MB
	Visual C++ 14 (2015) - 32-bit	Download 12.3 MB	Visual C++ 14 (2015) - 64-bit	Download 13.7 MB
	GCC 4.7.1 TDM (SJLJ) - 32-bit	Download 13.5 MB	GCC 4.7.1 TDM (SJLJ) - 64-bit	Download 16.3 MB
	GCC 4.8.1 TDM (SJLJ) - 32-bit	Download 13.3 MB	GCC 4.8.1 TDM (SJLJ) - 64-bit	Download 15.3 MB
	GCC 4.9.2 MinGW (DW2) - 32-bit	Download 13.6 MB	GCC 4.9.2 MinGW (SEH) - 64-bit	Download 14.5 MB

On Windows, choosing 32 or 64-bit libraries should be based on which platform you want to compile for, not which OS you have. Indeed, you can perfectly compile and run a 32-bit program on a 64-bit Windows. So you'll most likely want to target 32-bit platforms, to have the largest possible audience. Choose 64-bit packages only if you have good reasons.

The compiler versions have to match 100%! Here are links to the specific MinGW compiler versions used to build the provided packages: TDM 4.7.1 (32-bit), TDM 4.7.1 (64-bit), TDM 4.8.1 (32-bit), TDM 4.8.1 (64-bit), MinGW Builds 4.9.2 (32-bit), MinGW Builds 4.9.2 (64-bit)

解壓縮之後貼到適當的地方,本例子是放在C槽根
 目錄下

本機	8 > Windows 10 (C:)		
	名稱 ^	修改日期	類型
	🣜 Garena	2018/03/12 00:34	檔案資料夾
ж	MinGW	2018/01/31 18:34	檔案資料夾
×	📜 PerfLogs	2018/04/12 07:38	檔案資料夾
*	📜 Program Files	2018/05/02 19:02	檔案資料夾
*	📜 Program Files (x86)	2018/05/02 19:05	檔案資料夾
*	Windows	2018/05/07 15:44	檔案資料夾
	🧵 使用者	2018/05/01 19:20	檔案資料夾
	SFML-2.3.2	2018/05/07 16:42	檔案資料夾

 先設定環境變數:控制台>系統>進階系統設定 加入 C:\SFML-2.3.2\bin 忘記這一步會造成執行階段缺少dll而無法運作

← → * ↑ (+) → * ↑ (+) →	
控制合首頁 電腦各種 硬種 注階 系統保護 遊場 yu-husanchen 的使用者模数(U) 結構得論解	x
 ● 装置管理員 ● 返還設定 ● 返還設定 ● 返還設定 ● 返還設定 ● 回目 ● 認知 ● 回目 ● 回	-
	新增(N)
・ 準備系統設定 設定(S) ・ ・ といいSerstyd-HosancherryappUata(tocar(te) %SystemRoot%(system32 %SystemRoot%) 。	温中可(E)
新増(N) 新増(E) 用除(D) %System32\Wbem 27 %SYSTEMROOT%\System32\WindowsPowerShell\v1.0\	UUU(B)
開於線壁人時的亮面設定 系統異範(5) C\/MinGW\/bin C\Program Files\Java\jdk1.8.0_161\/bin C\Program Files\Java\jdk1.8.0_151\/bin C\Program Files\Java\jick1.8.0_151\/bin	10193 (UV)
単定(E)	上称(U)
DriverData C:\Windows\system32\DriverData NUMBER_OF_PRO 2 OS Windows NT	下称(0)
条統統型・系統失敗・反映構成計 Path C\\Program Files (x86)\Common Files\Orac ↓	机文字(门
設定(I)	
履携變數(N)	
「確定 取消 (A) 「確定 請參問 ★快速存取	1833

 • 啟動Visual Studio,並且建立一個Win32主控台應
 用程式專案

新聞專案								? ×				
最近使用的总本		.NET Fra	amework 4	× 排序依据:	预 设	× 88 🖽	提尋 已安装的筋本	P				
已安装的轮本 ▲ Visual C++		8	Win32 主挂台	應用程式		Visual C++	型別: Visual C++ 建立 Win32 主控台應用程式的	要案				
AIL CLR 一般 MFC			Win32 專案			Visual C++	Win32 應用程式精靈 - SFML-0	1			?	х
Win32							料觀	應用程式。	類型:	對下列加入通用標葉檔:		
 ▶ 具他語言 ▶ 具他專案類型 ▶ 資料庫 ▶ 測試專案 							應用程式設定	⁰ Window ®∓≉≎≁।	ws 應用程式(W) 應用得式(O)	□ATL(<u>A</u>) □ MEC(M)		
線上範本								Optil (P)				
								○ 新慈程:	式庫(S)			
								其他選項:	:			
								□空専来((E)			
								"REH-ST	號(X)			
CT 10/NP.	5 ER 41 - 04							25 先行编辑	譯標頭檔(P)			
直接(14); 位置(1);	chusers/vu-hus	anchen\d	locuments\visu	ual studijo 2011	N Projects	v						
方聚名藓(M);	SFML-01	and an an an a		01010000								
										<上─歩 下一歩	> 完成	取消

- 必須先引入SFML相關的標頭檔。
- 專案>屬性 開啟專案屬性頁
- 點擊組態管理員, 新增一個x64平台

組態管理員						?	\times
使用中的方	ī案組態(C):		使用中的方	案平台(P):			ALC: NOT THE OWNER
Debug			x64				
專案內容 (選取專案組態以建置或部署)(R):						
專案		組態		平台		建置	
SFML-01		Debug	\sim	x64	\sim	~	2

- 組態屬性>C/C++>一般:其他Include目錄 (組態:所有組態)
 加入 C:\SFML-2.3.2\include
- 組態屬性>連結器>一般:其他程式庫目錄
 (組態:所有組態)
 加入 C:\SFML-2.3.2\lib

・ 組態屬性>連結器>輸入: 其他相依性 (組態: Debug)
 加入

sfml-graphics-d.lib; sfml-window-d.lib; sfml-system-d.lib; sfml-audio-d.lib; sfml-network-d.lib; kernel32.lib; user32.lib; gdi32.lib; winspool.lib; comdlg32.lib; advapi32.lib; shell32.lib; ole32.lib; oleaut32.lib; uuid.lib; odbc32.lib; odbccp32.lib; %(AdditionalDependencies)

• 組態屬性>連結器>輸入: 其他相依性 (組態: Release) 加入

sfml-graphics.lib;sfml-window.lib; sfml-system.lib; sfml-audio.lib; sfmlnetwork.lib;kernel32.lib; user32.lib; gdi32.lib; winspool.lib; comdlg32.lib; advapi32.lib;shell32.lib; ole32.lib; oleaut32.lib; uuid.lib; odbc32.lib; odbccp32.lib; %(AdditionalDependencies)

• 留意不要把原來的東西給複寫了

測試程式碼

```
#include <SFML/Graphics.hpp>
int main()
{
    sf::RenderWindow window(sf::VideoMode(200, 200), "SFML works!");
    sf::CircleShape shape(100.f);
    shape.setFillColor(sf::Color::Green);
    while (window.is0pen())
        sf::Event event;
        while (window.pollEvent(event))
        {
            if (event.type == sf::Event::Closed)
                window.close();
        }
        window.clear();
        window.draw(shape);
        window.display();
    return 0;
```



C:\Users\yu-husanchen\documents\visual studio 2010\Projects\SFML-01\x64\Release\SFML-01.exe





CMake & SFML 2.4.2

Build library in VS2017

附註:撰寫本篇時還停留在2.4.2,最新的2.5.0已經在2018.5.10發布, 可以直接相容Visual Studio 2017,可依照前述2010的做法部署環境

Download SFML 2.4.2

	Visual C++ 11 (2012) - 32-bit	Download 16.7 MB	VisualC++ 11 (2012) - 64-bit	Download 18.5 MB			
	Visual C++ 12 (2013) - 32-bit	Download 16.1 MB	VisualC++ 12 (2013) - 64-bit	Download 17.8 MB			
	Visual C++ 14 (2015) - 32-bit	Download 16.0 MB	Visual C++ 14 (2015) - 64-bit	Download 17.6 MB			
Windows	GCC 4.9.2 TDM (SJLJ) - 32-bit	Download 13.8 MB	GCC 4.9.2 TDM (SJLJ) - 64-bit	Download 15.8 MB			
THILDOWS	GCC 6.1.0 MinGW (DW2) - 32-bit	Download 15.3 MB	GCC 6.1.0 MinGW (SEH) - 64-bit	Download 16.2 MB			
	On Windows, choosing 32 or 64-bit libraries should be based on which platform you want to compile for, not which OS you have. Indeed, you can perfectly compile and run a 32-bit program on a 64-bit Windows. So you'll most likelywant to target 32-bit platforms, to have the largest possible audience. Choose 64-bit packages only if you have good reasons. The compiler versions have to match 100% Here are links to the specific MinGW compiler versions used to built the provided packages: TDM 6.9.2 (32-bit), TDM 4.9.2 (64-bit). MinGW Builds 6.1.0 (32-bit). MinGW Builds 6.1.0 (64-bit)						
	GCC- 64-bit			Download 11.7 MB			
Linux	On Liver, if you have a 64-bit OE then up	- have the disk's tealship					
	to install specific packages and/or use sp you're on a 64-bit Linux. If you require a 32-bit build of SFML you	n nave the ovoic contrain secific compilar options to 'I have to build it yourself-	installed by default. Compiling for 32-bit do so. So cownloading the 64-bit lbraries	is possible, but you have is the easiest solution if			
Linux	to install specific packages and/or use sp you're on a 64-bit Linux. If you require a 32-bit build of SFML you	n nave the on-bit coolchan becific compiler options to 'I have to build it yourself-	installed by default. Compiling for 32-bit do so. So cownloading the 64-bit lbraries	is possible, but you have is the easiest solution if			
Mac OS X	Clang - 64-bit (OS X 10.7+, compared	the offic compiler options to the option of	installed by default. Compiling for 32-bit do so. So cownloading the 64-bit Ibraries bc++)	is possible, but you have is the easiest solution if Download 5.34 MB			
Mac OS X	Clang - 64-bit (OS X 10.7+, compatible w Mac 05 X libraries are only compatible w	thave to build it yourself. Thave to build it yourself. tible with C++11 and li	installed by default. Compiling for 32-bit do so. So cownloading the 64-bit Ibraries bc++)	is possible, but you have is the easiest solution if Download 5.34 MB			
Mac OS X	Clang - 64-bit IOS X 10.7+, compatible w Mac 05 X libraries are only compatible w	I have the 64-bit coolchan becific compiler options to I have to build it yourself- tible with C++11 and li with 64-bit systems.	installed by default. Compiling for 32-bit do so. So cownloading the 64-bit Ibraries bc++)	Is possible, but you have is the easiest solution IF Download 5.34 MB			
Mac OS X	Clang - 64-bit IOS X 10.7+, compa Mac 05 X libraries are only compatible w Source code	I have the 64-bit coolchan becific compiler options to I have to build it yourself- tible with C++11 and li with 64-bit systems.	installed by default. Compiling for 32-bit do so. So cownloading the 64-bit Ibraries bc++)	is possible, but you have is the easiest solution if Download 5.34 MB Download 23.8 MB			

 2017沒有直接支援, 請下載Source Code

 與此同時請安裝好 Cmake <u>https://cmake.org/</u>

8 > 下載	▲ CMake 3.11.1 - C:/Users/yu-husanchen/Downloads/ – □ ×
 ▶ 下載 名稱 ② [07] Being a JSP.pptx ③ [08] Script-Free Pages.pptx ③ [08] Script-Free Pages.pptx ② cmake-3.11.1-win64-x64.msi ③ SFML-2.3.2-windows-vc10-64-bit.zip ③ SFML-2.4.2-sources.zip 	▲ CMake 3.11.1 - C:/Users/yu-husanchen/Downloads/ — □ × File Tools Options Help Where is the source code: C:/Users/yu-husanchen/Downloads/SFML-2.4.2 Browse Source Where to build the binaries: sers/yu-husanchen/Downloads/SFML-VC15 ∨ Browse Build Search: □ Grouped Advanced ♣ Add Entry ♀ Remove Entry Name Value
SFML-2.4.2 2 SFML-VC15 2	

選擇source code 路徑與輸出路徑
 記得用資料夾裝起來(不然會很多檔案直接炸開)
 點選下面的Configure繼續

		?	×
÷			
SI	pecify the generator for this project		
1	Visual Studio 15 2017		•
0	ptional toolset to use (argument to -T)		
C	Use default native compilers		
С) Specify native compilers		
С) Specify toolchain file for cross-compiling		
C	Specify options for cross-compiling		
	Finish	Car	ncel

確定版本是Visual Studio 15 2017或是Visual Studio
 2017 Win64,點Finish
 選定一個平台後後面建置也必須選擇相同的環境

🛦 CMake 3.11.1 - C:/Users/yu-husanchen/Downloads/SFML-VC15 – 🛛 X						
File Tools Options	Help					
Where is the source code: 0	Misers/yn-husanchen/Downloads/SFML-2.4.2 Erowse Source					
Where to build the binaries:	C:/Users/yu husanchen/Downloads/SFML VC15 ~ Browse Build					
Search:	🗌 Grouped 🔲 Advanced 🗳 Add Entry 🔅 Remove Entry					
Name	Value					
CMARE_BUILD_TYPE CMARE_CONFIGURATIO CMARE_INSTALL_PREFIX SFML_BUILD_DOC SFML_BUILD_EXAMPLES SFML_GENERATE_PD3 SFML_OPENGL_ES SFML_USE_STATIC_STD_	Release N.T Debug;Release;MinSizeRel;RelWithDebInfo C:/Program Files (x85)/SFML LIBS					
Press Configure to update a	nd display new values in red, then press Generate to generate selected build files.					
Configure Generate	Open Project Current Generator: Visual Studio 15 2017					
Found OpenAL: C:/User Found VORBIS: C:/User Contiguring done	ers/yu-husanchen/Downloads/SFML-2.4.2/extlibs/libs- ers/yu-husanchen/Downloads/SFML-2.4.2/extlibs/libs- s/yu-husanchen/Downloads/SFML-2.4.2/extlibs/libs-ms					
C	>					

看到一堆紅色不要緊張,
 下面已經顯示Configuring
 done表示一切就緒

不需要更動任何東西,點
 Generate

← → ~ ↑ 🖡	> 本機	» 下載 » SFML-VC15			~
		名稱 ^	修改日期	類型	大小
★ 快速存取 ▲ 桌面	*	CMakeFiles	2018/05/07 18:34	檔案資料夾	
東京		📜 src	2018/05/07 18:30	檔案資料夾	
📕 下車00	×	all_BUILD.vcxproj	2018/05/07 18:34	VC++ Project	58 KB
🎦 文件	*	ALL_BUILD.vcxproj.filters	2018/05/07 18:34	VC++ Project Filter	1 KB
🔚 圖片	*	cmake_install.cmake	2018/05/07 18:34	CMAKE 檔案	4 KB
📜 Projects	*	CMakeCache.txt	2018/05/07 18:30	TXT 檔案	21 KB
		INSTALL.vcxproj	2018/05/07 18:34	VC++ Project	12 KB
5 本機		INSTALL.vcxproj.filters	2018/05/07 18:34	VC++ Project Filter	1 KB
学 網路		SFML.sln	2018/05/07 18:34	Microsoft Visual St	11 KB
		ali ZERO_CHECK.vcxproj	2018/05/07 18:34	VC++ Project	48 KB
		ZERO_CHECK.vcxproj.filters	2018/05/07 18:34	VC++ Project Filter	1 KB

完成後回到剛剛的目標資料夾,雙擊sln檔開啟專案,進行後續操作





分別以Debug及Release作為目標平台,建置專案(Ctrl + Shift + B)
 缺少任何一個都會使後續寫SFML程式時產生LNK2019 Error
 (留意目標環境是Win32 or x64)

完成之後現在你有兩個東西 原來的source code以及剛剛用Visual Studio 編譯過



木機 > 下載 > SFML-VC15-Proj			0 v			
▲ 名稱 ▲	修改日期	類型	7 3			
 ▲福 CMakeFiles Ib Release Src Win32 ALL_BUILD.vexproj ALL_BUILD.vexproj ALL_BUILD.vexproj Cmake_install.emake CMakeCoche.txt INSTALL.vexproj INSTALL.vexproj INSTALL.vexproj 	修改日期 2018/5/8 00:14 2018/5/8 00:11 2018/5/8 00:11 2018/5/8 00:04 2018/5/8 00:05 2018/5/8 00:05 2018/5/8 00:05 2018/5/8 00:05 2018/5/8 00:05 2018/5/8 00:05	類型 構案資料夾 構案資料夾 構案資料夾 構案資料夾 WC++ 専案 WC++ 専案師 CMAKE 構築 文字文件 WC++ 専案師 VC++ 専案師 VC++ 専案師	→ 本巻 → 下載 → SFML-2.4.2-source ▲ 名稱 ● cmake ● doc ● examples	修改口期 2018/5/8 00:03 2018/5/8 00:03 2018/5/8 00:03 2018/5/8 00:03	類型 榴案資料夾 檔案資料夾 檔案資料夾	大小
SFMLsIn ZERO CHECK.vcxproj ZERO_CHECK.vcxproj.filters	2018/5/8 00:05 2018/5/8 00:05 2018/5/8 00:05	Visual Studio VC++ 専奏師	extlibs include src tools editorconfig changelog.txt CMakeLists.txt CONTRIBUTING icense.txt readme.txt	2018/5/8 00:03 2018/5/8 00:03 2018/5/8 00:03 2018/5/8 00:03 2017/2/8 03:29 2017/2/8 03:29 2017/2/8 03:29 2017/2/8 03:29 2017/2/8 03:29	 檔案資料夾 檔案資料夾 檔案資料夾 檔案資料夾 日田ORCONHG 文字文件 文字文件 横案 文字文件 文字文件 文字文件 	1 KB 21 KB 17 KB 1 KB 2 KB 2 KB

- 建立一個新的資料夾
 從source code複製include資料夾
 從專案資料夾中複製lib\Debug及lib\Release的所有
 資料到新的資料夾的lib目錄下
- 結構如右圖
 完成後這個資料夾就是我們
 自己製作的SFML Library
 一樣將它放到C:\底下

me ^	Date Modified	Size	K
include	Today at 00:40		F
SEML	Today at 00:40		F
📄 lib	Today at 01:00		F
sfml-audio-2.dll	Today at 00:13	910 KB	N
🖞 sfml-audio-d-2.cli	Today at 00:58	1.4 MB	N
sfml-audic-d-2.ilk	Today at 00:58	1.9 MB	D
🧃 sfml-audic-d.exp	Today at 00:58	34 KB	5
sfml-audio-d.lib	Today at 00:58	57 KB	D
🗟 sfml-audic.exp	Today at 00:13	34 KB	S
🗋 sfml-audic.lib	Today at 00:13	56 KB	D
sfml-audic.pdb	Today at 00:58	2.6 MB	D
sfml-graphics-2.dll	Today at 00:14	754 KB	N
sfml-graphics-d-2.dll	Today at 00:59	1.4 MB	N
sfml-graphics-d-2.ilk	Today at 00:59	2.5 MB	D
🚽 sfml-graphics-d.exp	Today at 00:59	90 KB	S
^{**} sfml-graphics-d.lib	Today at 00:59	152 KB	D

- 與2010需要做的事情相同,必須要將剛剛的資料夾加入系統環境變數 請加入:C:\SFML-2.4.2\lib
- 開啟Visual Studio 2017
 建立一個Windows 主控
 台應用程式的專案

1.W.S						7	×
		山宮向)			운희 (CH+F)		ρ.
enta 日気袋 Vienal C++ Windows Dus 所 ATL CMake 別別 其他専業項型 第	.kcy		Windows主任印度片短山 Windows M 般却の日 月末 前前時に行力性 (DLL) 静壁短山市 Windows M 裕文和日	Visual C++ Visual C++ Visual C++ Visual C++	가만 Veusion 대왕과가 Windows 1 한 국	ton Π.Rπ	2 ²
R不利性要素 Tist Visual Sta	股的項目嗎? adro 安装伊提						
盟(N)	SPMI Console						
¥0)	C.\Users\Yu=Houan C	heri(Do	cuments\//isual Studio 2017\Projects		温度(F)		
素培稿(3-0 -	SPMI-Console			=] 总方案建立目券(D)] 加入原始經控制中(U)		
					9.JE	\$C	À .

- 專案>屬性導入相關標頭檔
- 組態屬性>C/C++>一般:其他Include目錄 (組態:所有組態)
 加入 C:\SFML-2.4.2\include
- 組態屬性>連結器>一般:其他程式庫目錄
 (組態:所有組態)
 加入 C:\SFML-2.4.2\lib

・ 組態屬性>連結器>輸入: 其他相依性 (組態: Debug)
 加入

sfml-graphics-d.lib; sfml-window-d.lib; sfml-system-d.lib; sfml-audio-d.lib; sfml-network-d.lib; kernel32.lib; user32.lib; gdi32.lib; winspool.lib; comdlg32.lib; advapi32.lib; shell32.lib; ole32.lib; oleaut32.lib; uuid.lib; odbc32.lib; odbccp32.lib; %(AdditionalDependencies)

• 組態屬性>連結器>輸入: 其他相依性 (組態: Release) 加入

sfml-graphics.lib;sfml-window.lib; sfml-system.lib; sfml-audio.lib; sfmlnetwork.lib;kernel32.lib; user32.lib; gdi32.lib; winspool.lib; comdlg32.lib;
advapi32.lib;shell32.lib; ole32.lib; oleaut32.lib; uuid.lib; odbc32.lib;
odbccp32.lib;%(AdditionalDependencies)

• 留意不要把原來的東西給複寫了

做到這裡,與前面的2010操作一樣 你的測試程式應該可以順利編譯執行了

