Using ID TECH Universal SDK Library Files in a C++ Project

Introduction

From time to time, customers who wish to use ID TECH's Universal SDK for Windows (which is .NET-based and comes with C# code examples) ask if it is possible to do development against the SDK solely in C++ (on Windows). The answer is yes. Universal SDK library files (DLLs) are COM-visible and ready to be accessed from C++ code. (SDK runtimes require the .NET Common Language Runtime, but your C++ binaries can still use the SDK.)

Note that while the example shown in this document involves Microsoft's Visual Studio, it is also possible to use SDK libraries in C++ projects created in Eclipse or other IDEs.

How to Use the IDTechSDK.dll File in a C++ Project:

1. Create a Visual C++ project in Visual Studio 2015 (shown below, an MFC Application as an example).

New Project					? ×
▶ Recent		.NET I	ramework 4.5 - Sort by: Default	- # =	Search Installed Templates (Ctrl+E)
▲ Installed			MEC Application	Visual C++	Type: Visual C++
▲ Templates ▷ Visual C# ▷ Visual Basic	4		MFC ActiveX Control	Visual C++	A project for creating an application that uses the Microsoft Foundation Class Library
 Visual C++ Windows	orm y tor Tunes		MFC DLL	Visual C++	
▷ Online			<u>Circk here to go onine and ind templa</u>	<u></u>	
Name:	MFCApplicatio	on1			
Location:	C:\workspace\	c#\TestN	IFC\	•	Browse
Solution name:	MFCApplicatio	on1			Create directory for solution Add to source control
					OK Cancel

2. Change the properties of the Visual C++ project.

Under the **General** tag, set Common Language Runtime Support under Target Platform to "Common Language Runtime Support (/clr)" under Windows.

MFCTest Property Pages								? ×
Configuration: Release		✓ Platform:	Active(Win32)			~	Configuratio	on Manager
Configuration Properties General Debugging VC++ Directories VC++ Directories VC++ Linker Manifest Tool Resources XML Document Generatu Browse Information Build Events Couten Build Step Code Analysis	General Target Platform Target Platform Gutput Directory Intermediate Directory Intermediate Directory Target Extension Extension to Delete on Clean Build Log File Platform Toolset Enable Managed Incremental Build Project Defaults Configuration Type Use of MFC Character Set Common Language Runtime Support Minde ws Store App Support		Windows 8.1 S(ColutionD) S(Configura s(ProjectNa exe *.cdf;r.cach S(imDir)S(U) Visual Stud Yes Application Use MrC in Use MrC in Use MrC in Use Link Tir No	iii)\$(Configuration)\ tion)\ me) \$\$BuildProjectName).lo \$\$BuildProjectName).lo io 2015 (v140) (exe) a Static Library e Character Set anguage Runtime Sup me Code Generation	pdb;*.iobj;*.resources;*.ti 2g opport (/clr)	lb;*.tli;*.tli;*.tln;*.tmp;*.rsp	9;*pgc;*pgd;*r	neta,*.l0g.*.ma
	Target Platform The current target platform of the project.							
, ,	2					確定	取消	套用(A)

3. Under VC++ Directories, add the path to the C# .dll file(s) to Reference Directories.

MFCTest Property Pages						? >
Configuration: Release		✓ <u>P</u> latform:	Active(Win32)		~	Configuration Manager
Configuration Properties General Debugging VC+ + Directories VC+ + Directories VC+ + Namifest Tool Resources XML Document Generatic Browse Information Build Events Cutom Build Step Code Analysis	General Executable Directories Include Directories Reference Directories Library Directories Library WinRT Directories Source Directories Exclude Directories		S(VC_ExecutablePath)x8 S(VC_IncludePath);S(Vin S(VC_ReferenceSPath)x S(VC_LibraryPath)x86);S(S(VindowsSDK_Metada S(VC_SourcePath); S(VC_IncludePath);S(Vin	i6);5(WindowsDK_ExecutablePath); od <u>ows5DK_includePath);</u> && <u>biCvworkspacev</u> c#,TestMFC/MF WindowsSDK_ubraryPath_X86);5(N taBath); ndowsSDK_includePath);5(MSBuild_t	\$(VS_ExecutablePe CTest;) ETFXKisDir)Lib\um ExecutablePath);\$(th);\$(MSBuild_ExecutablePat \x86 /C_LibraryPath_x86);
<	Executable Directories Path to use when searching for executable	files while building a V	'C++ project. Corresponds to environm	ent variable PATH.		
					確定	取消 養用(A)

4. Under C/C++ General, set Common Language Runtime Support to "Common Language Runtime Support (/clr)."

onfiguration: Release	<u> Platform:</u> Active(Win32	!)	Configuration Manager
 Configuration Properties General Debugging VC++ Directories C/C++ General Optimization Preprocessor Code Generation Language Precompiled Headers Output Files Browse Information Advanced All Options Command Line Linker Manifest Tool Resources XML Document Generator Browse Information Build Events Custom Build Step Code Analysis 	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 SDL checks Multi-processor Compilation	Program Database (/Zi) Common Language RunTime Suppo Yes (/nologo) Level3 (/W3) No (/WX-) Yes (/sdl)	brt (/clr)
	Additional Include Directories Specifies one or more directories to add to th (/[nath])	e include path; separate with semi-colo	ns if more than one.

5. Under C/C++ Preprocessor, add _AFXDLL to Preprocessor Definitions.

IFCTest Property Pages			? >
onfiguration: Release	✓ <u>P</u> latform: Ad	ctive(Win32)	✓ Configuration Manager
Configuration Properties General Debugging VC++ Directories C/C++ General Optimization Preprocessor Code Generation Language Precompiled Header: Output Files Browse Information Advanced All Options Command Line Linker Manifest Tool Resources XML Document Generatv Browse Information Build Events Custom Build Step Code Analysis	Preprocessor Definitions Undefine Preprocessor Definitions Undefine All Preprocessor Definitions Ignore Standard Include Paths Preprocess to a File Preprocess Suppress Line Numbers Keep Comments	WIN32;_WINDOWS;NDEBU	; AFXDLL: (PreprocessorDefinitions)
· · · · · · · · · · · · · · · · · · ·	Preprocessor Definitions Defines a preprocessing symbols for your source	ce file.	
			確定 取消 套用(A)

6. Under C/C++ Code Generation, change Runtime Library to "Multi-threaded DLL (/MD)."

onfiguration:	Release	 <u>P</u>latform: Active 	(Win32)	~	Configuration	Manag	er
Configurat Genera Debugg VC++E Gen Opt Preg Cod Lang Prec Out Brow Adv All Con Linker Manifet Resour SMLDc Brow Build Ev Coto	on Properties ping birectories eral mization rocessor 6 Generation guage ompiled Headers but Files use Information anced ptions umand Line t Tool es uscument Generatod Information ents Build Step nalysis	Enable String Pooling Enable Minimal Rebuild Enable C++ Exceptions Smaller Type Check Basic Runtime Checks Runtime Library Struct Member Alignment Security Check Control Flow Guard Enable Function-Level Linking Enable Parallel Code Generation Enable Enhanced Instruction Set Floating Point Model Enable Floating Point Exceptions Create Hotpatchable Image	Yes with SEH Exceptions (/EHa) No Default Multi-threaded DLL (/MD) Default Enable Security Check (/GS) Yes (/Gy) Not Set Precise (/fp:precise)				
		Enable String Pooling Enables the compiler to create a single read-only co in smaller programs, an optimization called string p	opy of identical strings in the program image and ooling. /01, /02, and /ZI automatically set /GF oj	in memo ption.	ry during executi	on, resul	ting

7. Under **Code Analysis General**, change Rule Set to "Microsoft Mixed (C++ /CLR) Recommended Rules."

Configuration: Release Active(Win32) Configuration Manager Command Line Amalifest Tool General Suppress results from generated code (managed only) Rule Set Rule Set Run this rule set. Microsoft Mixed (C++ /CLR) Recommended Rules Command Line Resources General All Options Command Line Swappress results from generated code (managed only) Run this rule set: Microsoft Mixed (C++ /CLR) Recommended Rules Command Line All Options Command Line set: any cutom rule set you create for your C++ projects that support the Common Language Runtime. This ruleset is designed to be configured with the Visual Studio Professional edition and higher. All Options Command Line All Options Command Line All Options Command Line All Options Command Line Browse Information Centrol Mixed Studio Professional edition and higher. Path: Path: Post-Build Events Vole Shall Stevent Pre-Link Event Pre-Link Event Pre-Link Event Pre-L	VIFCTest Property Pages			? >
Command Line Inable Code Analysis on Build General Suppress results from generated code (managed only) Input and Output Rele Set Advanced All Options Command Line Resources General All Options Command Line Description: All Options Description: Command Line Description: All Options Command Line Path: Path: Description: Description: Command Line Path: Build Events Pre-Build Event Pre-Build Event Path: Code Analysis General	Configuration: Release	✓ Platform: Active(Win32) ✓	Configuration N	/lanager
	Command Line Manifest Tool General Input and Output Isolated COM Advanced All Options Command Line Kesources General All Options Command Line XML Document General All Options Command Line Browse Information General All Options Command Line Build Events Pre-Build Event Pre-Link Event Pret-Link Event Pret-Link Event Code Analysis General Code Analysis	Inable Code Analysis on Build Suppress results from generated code (managed only) Rule Set Run this rule set: Microsoft Mixed (C++ /CLR) Recommended Rules Description: These rules focus on the most common and critical problems in your C++ projects that support the Common Language Runtime, including potential security holes, application crashes, and other inportant logic and designed to be configured with the Visual Studio Professional edition and higher. Path: v		

- 8. Use **IDTechSDK.dll** in your .cpp file.
- a. Open a .cpp file in the Visual C++ project (MainFrm.cpp, for example).

b. Add #using "IDTechSDK.dll" and using namespace IDTechSDK below any #include and #define statements.



c. Declare an object and call the functions in IDTechSDK.dll.



9. Finally, clean and build the Visual C++ project. Copy IDTechSDK.dll and all the other provided SDK .dll files to Debug and Release folders in the Visual C++ project.