Setting Up SDL on Visual C++ Express 2005

Visual C++ Express is Microsoft's free version of its Visual C++ development environment. I'm going to assume most of you who are running Windows are using Visual C++ Express so I'm devoting this section to setting things up so you can create SDL applications with Visual C++ Express.

If you haven't already done so, download and install Visual C++ Express from Microsoft's web site. It's in the Developer Tools section of the site. You can either perform a web installation or you can download Visual C++ Express and install it locally from your hard drive or a CD-ROM.

Installing the Windows Platform SDK

The basic Visual C++ Express installation is set for writing applications that use the .NET framework. The .NET framework is designed for people who want to create GUI and Web applications quickly. But it doesn't contain the OpenGL libraries and headers so if you try to write an OpenGL program, it won't compile. You need to install the Windows Platform SDK, also known as the Win32 SDK.

The Windows Platform SDK contains all the libraries and headers you need to write native software for 32-bit versions of Windows: XP, 2000, Me, 98, 95, and NT. The OpenGL libraries and headers are part of the Windows Platform SDK.

You can install the Windows Platform SDK from Microsoft's site at the same location you downloaded Visual C++ Express. Like Visual C++, you can either perform a web installation or download the SDK and install it locally from your hard drive or a CD-ROM. The SDK is large and takes a while to install so be patient. After installing the Windows Platform SDK, you must modify two files to reflect the fact you installed the SDK: the Visual C++ properties file and the Application Wizard's settings file.

Updating the Visual C++ Properties File

The `corewin_express.vsprops` file contains low-level properties Visual C++ uses. You must modify the file so Visual C++ links against some low-level libraries that are needed to build Win32 applications. Open the file, which is located at `C:\Program Files\Microsoft Visual Studio 8\VC\VCProjectDefaults`, in a text editor. Change the following line of code in the file:

```
AdditionalDependencies="kernel32.lib"
```

To the following:

```
AdditionalDependencies="kernel32.lib user32.lib gdi32.lib winspool.lib
comdlg32.lib advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib"
```

Make sure you save the file as plain text.

Updating the Application Wizard's Settings File

The Application Wizard Visual C++ Express uses to help you create new projects initially disables the ability to create Win32 projects. Win32 projects are disabled because Visual C++ is set to write .NET applications only. You must tell the Application Wizard to enable the creation of Win32 applications. To enable the creation of Win32 applications, you must modify the `AppSettings.htm` file, which you can find at the following location:

```
C:\ProgramFiles\Microsoft Visual Studio 8\VC\VCWizards\AppWiz\Generic\
Application\html\1033
```

Open the `AppSettings.htm` file in a text editor. To enable the creation of Win32 applications, you must comment out four lines of code that disable the creation of Win32 applications. Comment out lines 441-444 by typing `//` in front of them, which are the following lines of code:

```
WIN_APP.disabled = true;
WIN_APP_LABEL.disabled = true;
DLL_APP.disabled = true;
DLL_APP_LABEL.disabled = true;
```

Installing the SDL Libraries

If you haven't done so, download SDL from the SDL website. At the download site, you will see runtime and development libraries. You want to download the development library, which contains all the libraries and header files you need to write SDL games. Download the development library for Visual C++. Place the libraries in whatever directory you wish.

The code in this book also uses the SDL_image and SDL_mixer libraries. SDL_image makes loading graphics files easy, and SDL_mixer makes playing sound effects and background music easy. Make sure you download the development libraries for SDL_image and SDL_mixer and install them.

Telling Visual C++ Where to Find Your SDL Headers and Libraries

Now that you've installed the Win32 SDK and SDL, you must tell Visual C++ where to find the libraries and header files you installed. If Visual C++ can't find the libraries and header files, your OpenGL and SDL code will not compile.

1) Launch Visual C++.
2) Choose Tools > Options to open the options dialog box.
3) Select Projects and Solutions from the list on the left.
4) Select VC++ Directories. The dialog box should look similar to Figure 1.
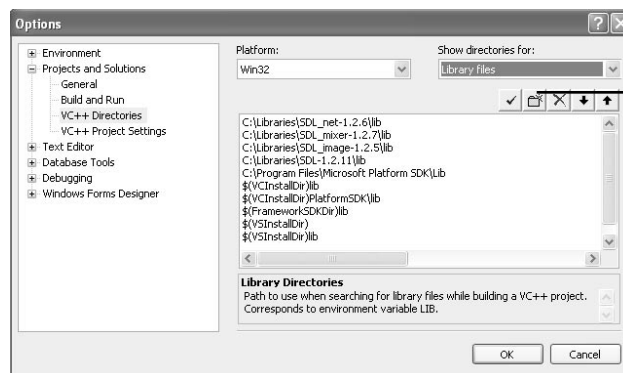
New line button

Figure 1

Options dialog box.

The code in this book uses the OpenGL, GLU, SDL, SDL_image, and SDL_mixer libraries. The OpenGL and GLU libraries are part of the Win32 SDK. The SDL, SDL_image, and SDL_mixer libraries are wherever you chose to install them. Let's start by telling Visual C++ where to find the libraries.

5) Choose Library files from the Show directories for pop-up menu.
6) Click the New Line button to add a library path.
7) Click the … button.
8) Navigate to the location of the Win32 SDK libraries, which should be `C:\Program Files\Microsoft Platform SDK\Lib`. Click the Open button to add the path.
9) Repeat Steps 6-8 to add the locations to the SDL, SDL_image and SDL_mixer libraries.

Now it's time to add the paths to the header files.

10) Choose Include files from the Show directories for pop-up menu.
11) Click the New Line button to add a library path.
12) Click the … button.
13) Navigate to the location of the Win32 SDK include directory, which should be `C:\Program Files\ Microsoft Platform SDK\Include`. Click the Open button to add the path.
14) Repeat Steps 11-13 to add the locations for the header files of the SDL, SDL_image and SDL_mixer libraries.

One last thing is to set the path to executable files for the Win32 SDK.

15) Choose Executable files from the Show directories for pop-up menu.
16) Click the New Line button to add an executable path.
17) Click the … button.
18) Navigate to the location of the Win32 SDK bin directory, which should be `C:\Program Files\ Microsoft Platform SDK\Bin`. Click the Open button to add the path.

Creating a SDL Project

You must create a project in Visual C++ if you want to write SDL applications.

1) Choose New > Project to open the New Project dialog box.
2) Select Win32 Console Application from the list of projects.
3) Name your project.
4) Click the OK button to open the Application Settings Wizard, which you can see in Figure 2.
5) Select Application Settings on the left size of the wizard.
6) Click the Windows application radio button.
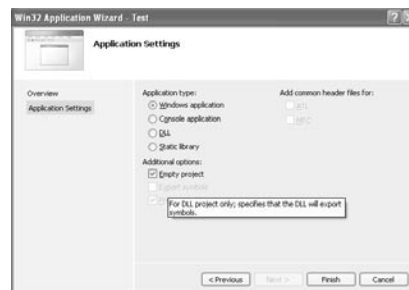7) Select the Empty project checkbox.
8) Click the Finish button.

Figure 2

Application Settings Wizard.

Project Settings

After creating your project there are a couple of project settings you must modify for your code to compile. Visual C++ projects start with two build configurations: debug and release. The debug configuration makes debugging your program easier, and the release configuration builds a shipping product. Remember to modify the project settings for both configurations.

1) Select the name of your project.
2) Choose Project > Properties.
3) Select C++ then Code Generation.
4) Make sure the runtime library is either Multi-threaded DLL or Multi-threaded Debug DLL.
5) Select Linker then Input.
6) Select Additional Dependencies. A dialog box opens, which you can see in Figure 3.
7) Add the following dependencies: `OpenGL32.lib`, `Glu32.lib`, `SDL.lib`, `SDLmain.lib`, `SDL_image.lib`, and `SDL_mixer.lib`.
8) Click the OK button.

Adding the SDL Libraries to Your Project

You should be able to compile a project at this point. But I was unable to compile a project until I explicitly added the SDL and OpenGL libraries to my project.

1) Choose Project > Add Existing Item to open the add file dialog box.
2) Choose All Files from the pop-up menu.
3) Navigate to the locations of the following libraries: `OpenGL32.lib`, `Glu32.lib`, `SDL.lib`, `SDL_image.lib`, and `SDL_mixer.lib`.

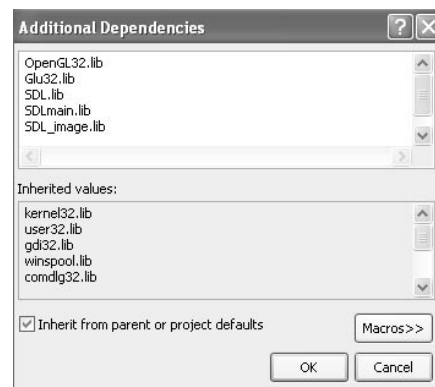Now you're ready to write SDL code with Visual C++.

Figure 3

Additional dependencies dialog box.