

## Using SIGIL in Visual Studio

Although there are multiple versions of Visual Studio, these instructions (which depict VC2015) still apply to other versions. The first thing you should do is create a new **Empty Project**.

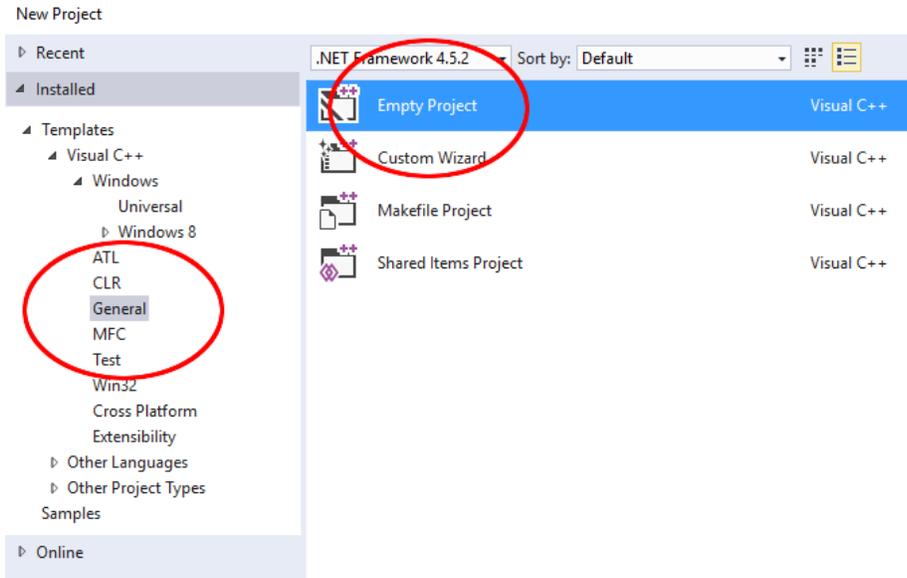


Figure 1. Click File -> New -> Project to open the **New Project** window.

Before you can configure your project for SIGIL, you need to create an initial C++ file so that Visual Studio knows your project type. Right click on your project in the **Solution Explorer**, and click **Add -> New Item**.

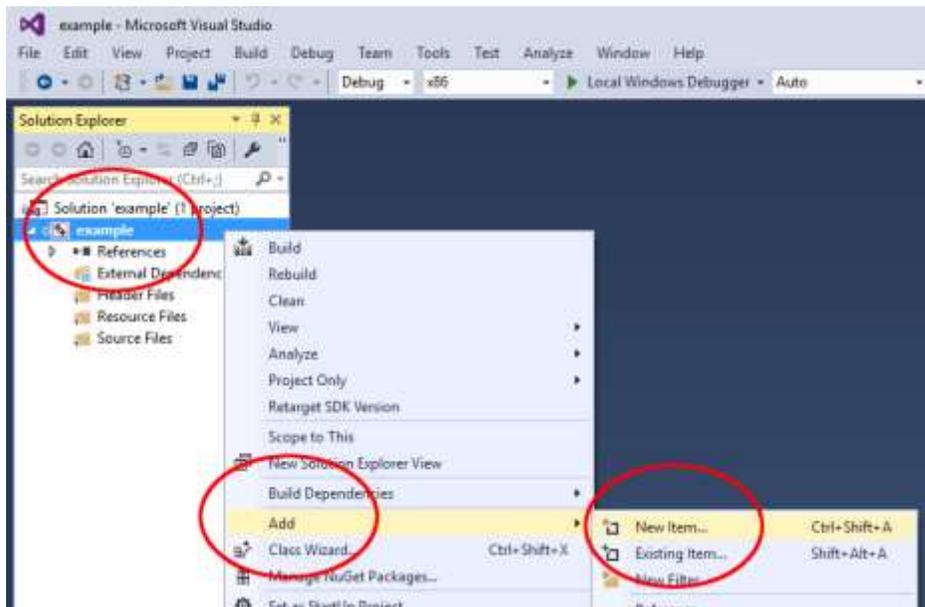


Figure 2. Use the **Solution Explorer** to add your first C++ file.

In the **Add New Item** window, select **C++ File (.cpp)** from the **Visual C++** section. Click **OK**.

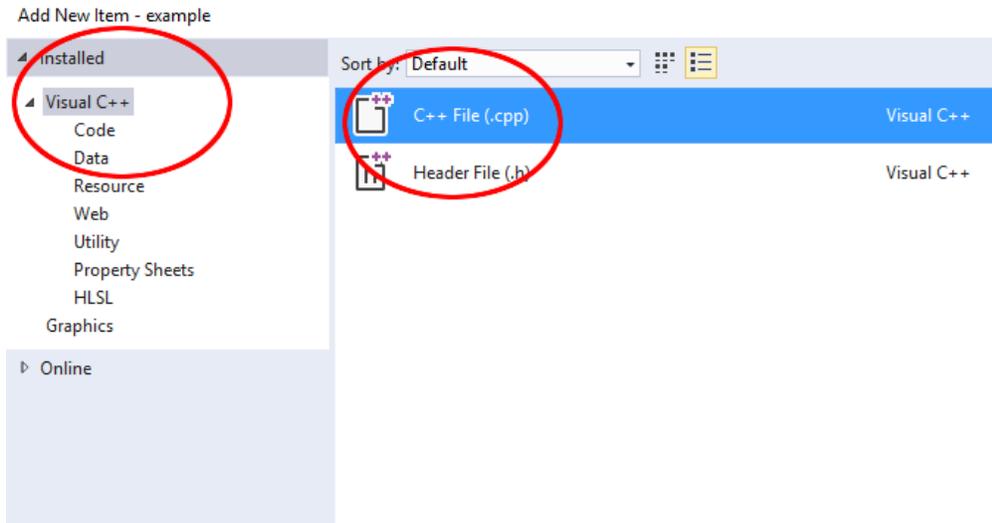
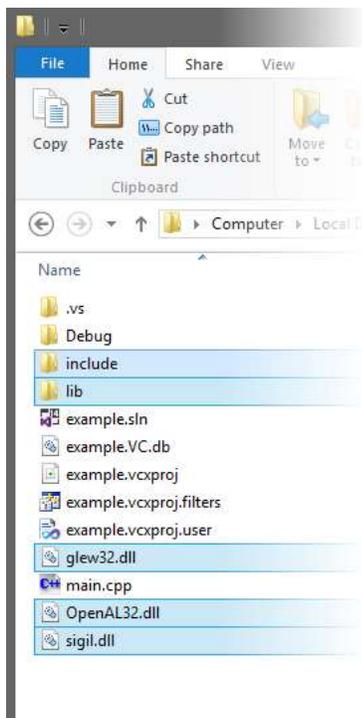
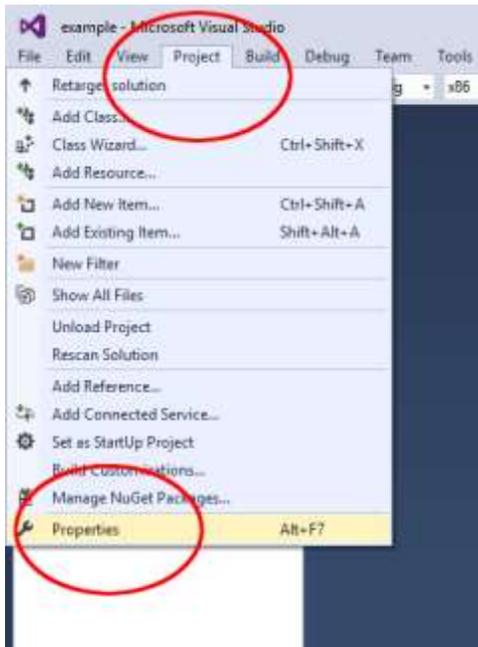


Figure 3. Add a C++ file to your project.



You will need to copy the relevant SIGIL files and libraries to your project folder so that Visual Studio can find them. There are several different ways to do this, but this guide assumes that you want to keep your libraries local to your project. Copy the files inside the **sigil-vcXXXX** folder (depending on your version of Visual Studio) and the **include** folder itself to your project location like this.

Figure 4. The **include** folder should contain **sl.h** and the **lib** folder should contain **sigil.lib**.



Next, you need to tell your Visual Studio project where to find the SIGIL files. Access your project's **Properties** by either (a) right-clicking on your project in the **Solution Explorer** and selecting **Properties**, or (b) selecting **Project -> Properties** from the top menu (which is shown here)

Figure 5. Open your project **Properties**.

In the resulting window, expand **Linker** on the left-hand side, and select **Input**. From here, open the drop-down menu in the **Additional Dependencies** section, and click **Edit**.

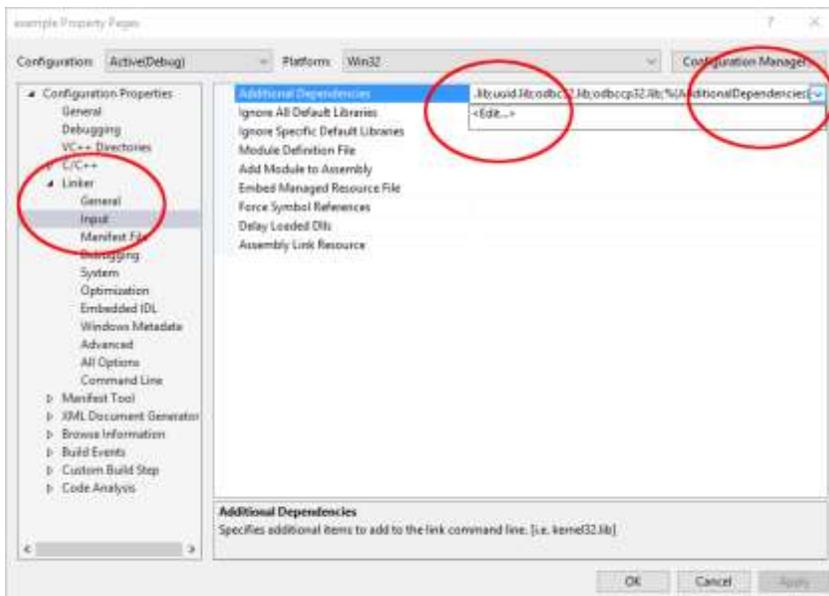


Figure 6. Edit your project dependencies settings.

The resulting **Additional Dependencies** window is shown below. In the first text box, enter the line **lib/sigil.lib**. This will allow your project to link correctly to SIGIL. This is what the **Additional Dependencies** window should look like after you do this. Click **OK**.

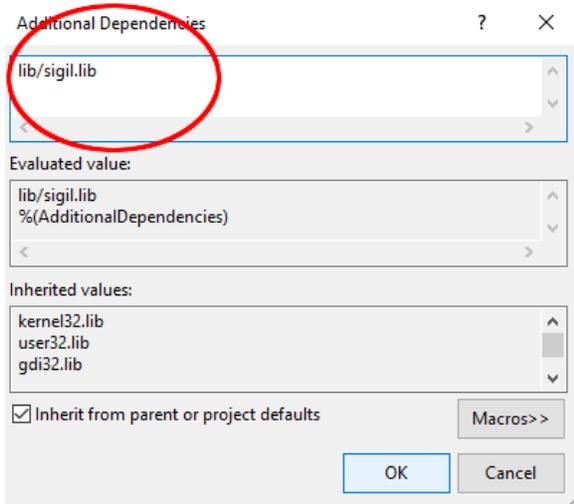


Figure 7. Instruct your project to link against SIGIL.

Next, you need to tell your project file where to find the SIGIL header. Still in your project's **Properties** window, expand **C/C++** and select **General**. From here, enter the value **include** into the **Additional Include Directories** section. Click **OK**.

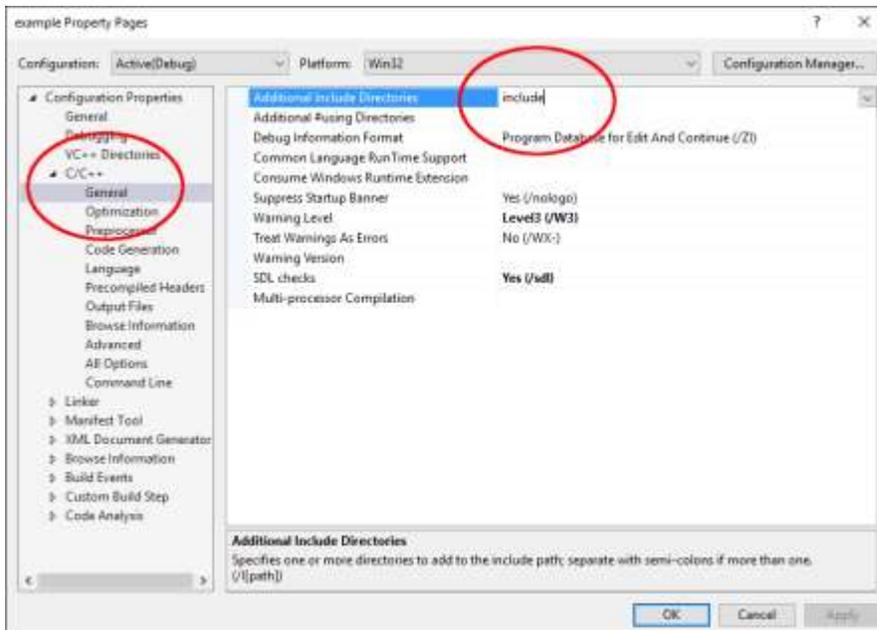


Figure 8. Tell your project where to look for the SIGIL header.

Save your project with **File -> Save All**. You're all set, and can start developing your SIGIL project.