

# Compiling Orx dependencies for new Visual Studio editions

This document is a cheat sheet for compiling up orx dependencies for new editions of Visual Studio in order to contribute back the compiled libraries into the release versions of orx.

\_This page is not for general use\_, it is low traffic, and prone to frequent changes and errors. If you are looking for compiled orx dependancies, you can download a precompiled orx, or clone from the the regular repo.

Clone <https://bitbucket.org/orx/orx-extern>

## Build libwebp

1. Switch to 32 bit compiler with: C:\Program Files (x86)\Microsoft Visual Studio 14.0\VC\bin\amd64\_x86\vcvarsamd64\_x86.bat
2. In a VS2015 console, go to the libwebp folder and: `nmake /f Makefile.vc CFG=release-static RTLIBCFG=static OBJDIR=output/32`
3. Ignore errors, locate the lib at: libwebp\output\release-static\x86\lib\webpdecoder.lib
4. Switch to 64 bit compiler mode with: C:\Program Files (x86)\Microsoft Visual Studio 14.0\VC\bin\x86\_amd64\vcvarsx86\_amd64.bat
5. `nmake /f Makefile.vc CFG=release-static RTLIBCFG=static OBJDIR=output/64`
6. Ignore errors, locate the lib at: libwebp\output\release-static\x64\lib\webpdecoder.lib

## Build OpenAL-Soft

1. Install CMake 2.6 or better
2. In the build/windows/vc2015 folder, create a VS2015 project with: `cmake -DLIBTYPE=STATIC -G "Visual Studio 14 2015"`
3. Open the OpenAL.sln into Visual Studio.
4. You may need to adjust the includes paths to be relative instead of absolute.
5. Switch to release, right click on the OpenAL32 project and select build to make the 32-bit static lib.
6. Find the 32 bit version in \Release\OpenAL32.lib
7. For 64bit: Select the dropdown that says win32
8. Click and select Configuration Manager
9. In the Active Solution Platform dropdown, click <New>
10. Select x64 and copy from win32 (leave all options default)
11. In the OpenAL32 project properties
12. - In C/C++→Code Generation
13. - - select Multithreaded (/MT)
14. - In Librarian / Command Line
15. - - remove additional options relating to /machine:X86
16. In the common project properties
17. - In Librarian / Command Line

18. - - remove additional options relating to /machine:X86
19. Right click the OpenAL32 project and select build to make the 64-bit static lib.
20. Find the 64bit version in \openal-soft\x64\Release\OpenAL32.lib

## Build freetype

Follow pretty much the same as OpenAL-Soft

## Build Box2D

1. Copy build VC13 to VC14
2. Create a lib/msvs2014/32 and 64 folders
3. Open solution is VS2015
4. Allow it to convert the project
5. Convert the solution by selecting the solution in solution explorer, and save as over the top of the .sln
6. In both Win32 and x64 configurations, get properties on the Box2D project
7. Change Advanced / Disable Specific Warnings - Add 4577;4456;%(DisableSpecificWarnings) for both debug and release builds
8. In General / Output Directory - change to your new lib path and add a trailing slash.
9. Compile both Win32 and x64 configurations and both release and debug configurations.

## Build GLFW

1. Copy build VC13 to VC14
2. Create a lib/msvs2014/32 and 64 folders
3. Open solution is VS2015
4. Allow it to convert the project
5. Convert the solution by selecting the solution in solution explorer, and save as over the top of the .sln
6. In both Win32 and x64 configurations, get properties on the Box2D project
7. In General / Output Directory - change to your new lib path and add a trailing slash.
8. Compile both Win32 and x64 configurations.

## Build libsndfile

1. Copy build VC13 to VC14
2. Create a lib/msvs2014/32 and 64 folders
3. Open solution is VS2015
4. Allow it to convert the project
5. Convert the solution by selecting the solution in solution explorer, and save as over the top of the .sln
6. In both Win32 and x64 configurations, get properties on the libsndfile project
7. In General / Output Directory - change to your new lib path and add a trailing slash.

8. Compile both Win32 and x64 configurations.
9. Change `#define snprintf _snprintf` in `config.h` to be:
10. `#if _MSC_VER<1900`
11. `#define snprintf _snprintf`
12. `#endif`

From:

<https://www.orx-project.org/wiki/> - **Orx Learning**

Permanent link:

[https://www.orx-project.org/wiki/compiling\\_visual\\_studio\\_deps?rev=1455144569](https://www.orx-project.org/wiki/compiling_visual_studio_deps?rev=1455144569)

Last update: **2025/09/30 17:26 (7 months ago)**

