

blendreCave Windows build

This version is not up to date !

These indications are for Windows. If you use Linux/MacOSX, see [build](#) page.

To ease further references, lets suppose we work in a folder named C:\BlenderWorkspace. Any file or directory creation, if not mentioned otherwise, takes place in that folder.

We will use [TortoiseSVN](#) for repository access and patch application. Examples are given with screenshots using the GUI version of TortoiseSVN, but you can achieve same result using command-line (in any case, its a good idea to install command-line tools of svn when installing TortoiseSVN).

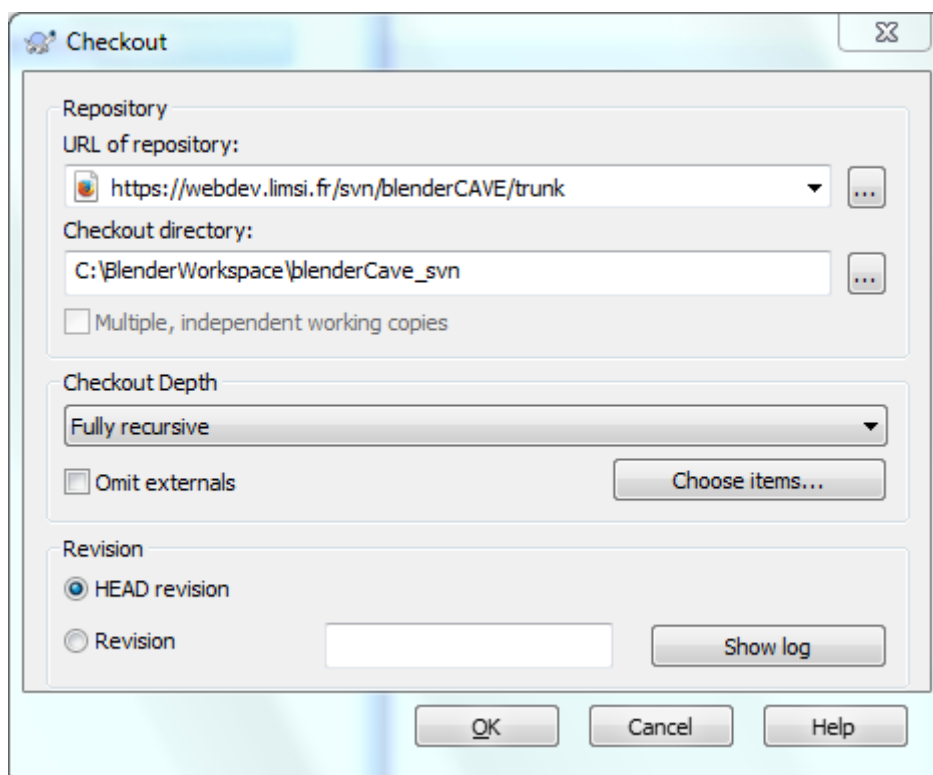
Download blenderCave sources

Next blenderCave version coming

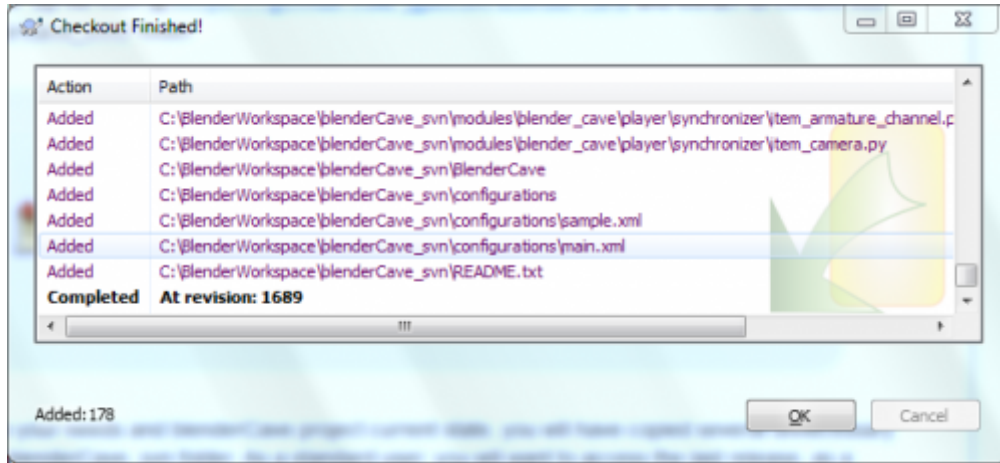
We are on the way of realeasing blenderCave 3.6 (latest is 3.2, intermediate releases are for internal use) - it will be downloadable when ready.

From LIMSI

Using TortoiseSVN contextual menu in your BlenderWorkspace directory, check-out <https://webdev.limsi.fr/svn/blenderCAVE/trunk> (the trunk or a specific releases/X.Y release of blenderCave you need) into a blenderCave_svn/ directory:



Wait until it is completely downloaded.



Download blender sources

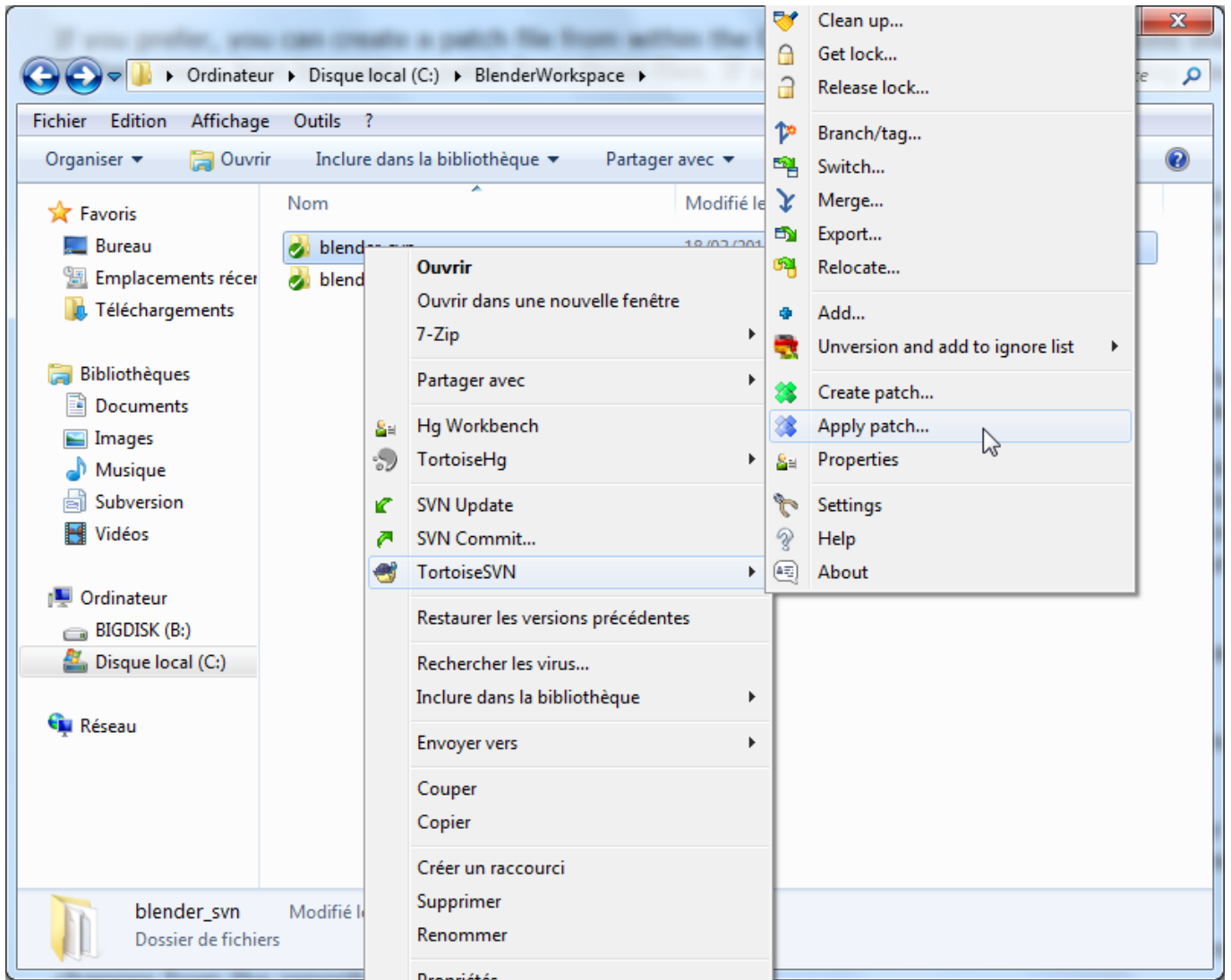
You must download the sources of **the blender version corresponding to the downloaded blenderCave patch version**. You can find that version number in the patch filename: blenderCave_svn/patch/blender_revision_xxxxx.patch.

You can find this version on [blender web site](#)

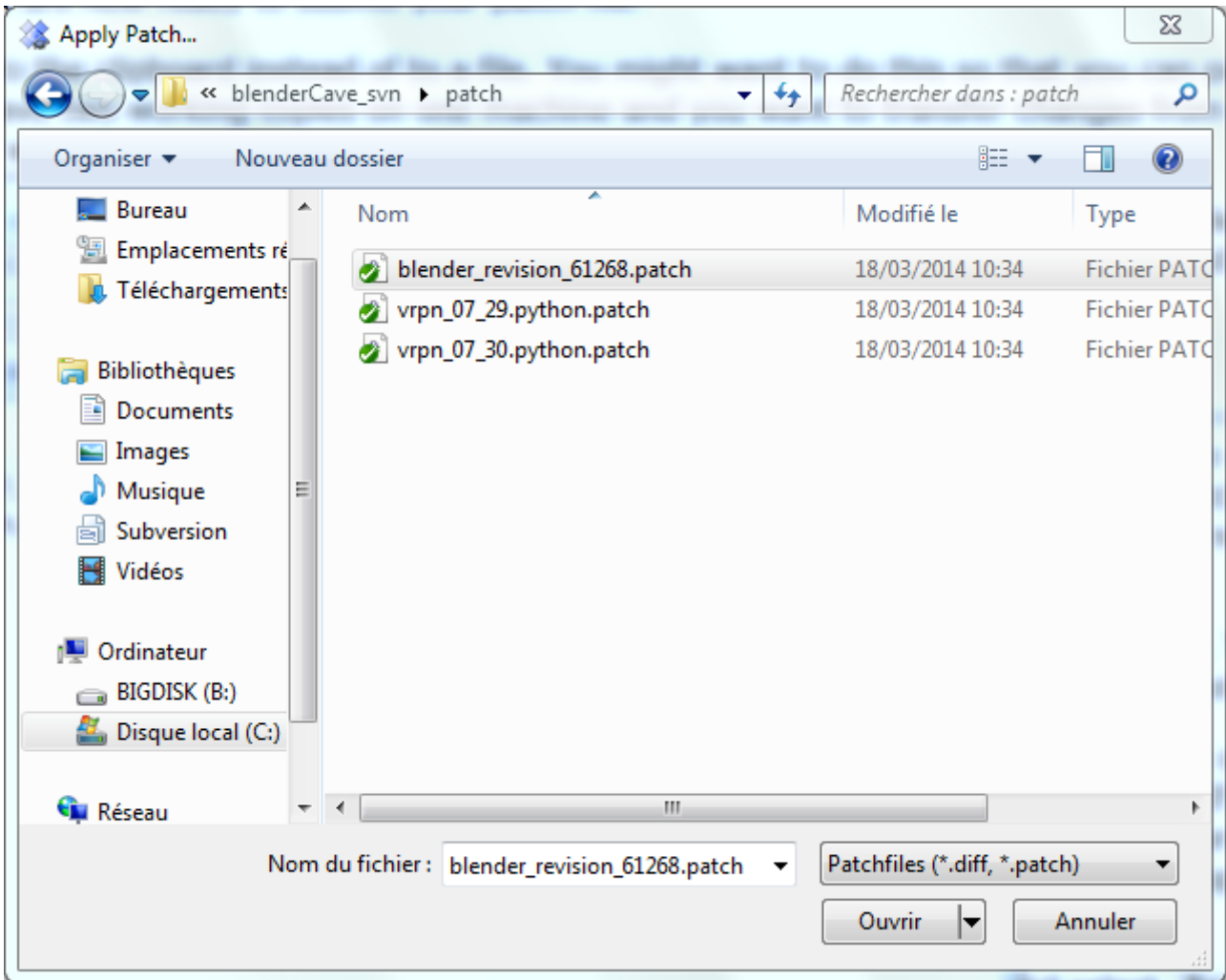
Then unzip it inside the C:\BlenderWorkspace\blender-src\.

Apply the patch

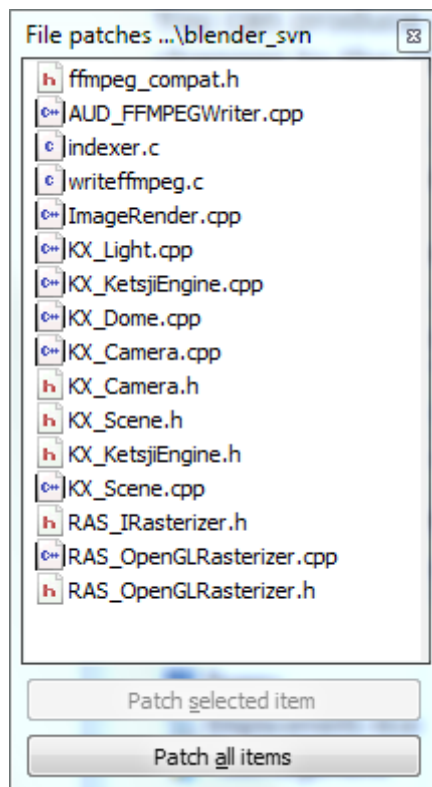
Using TortoiseSVN contextual menu on blender_src/ directory, select submenu **TortoiseSVN** then submenu **Apply patch...**



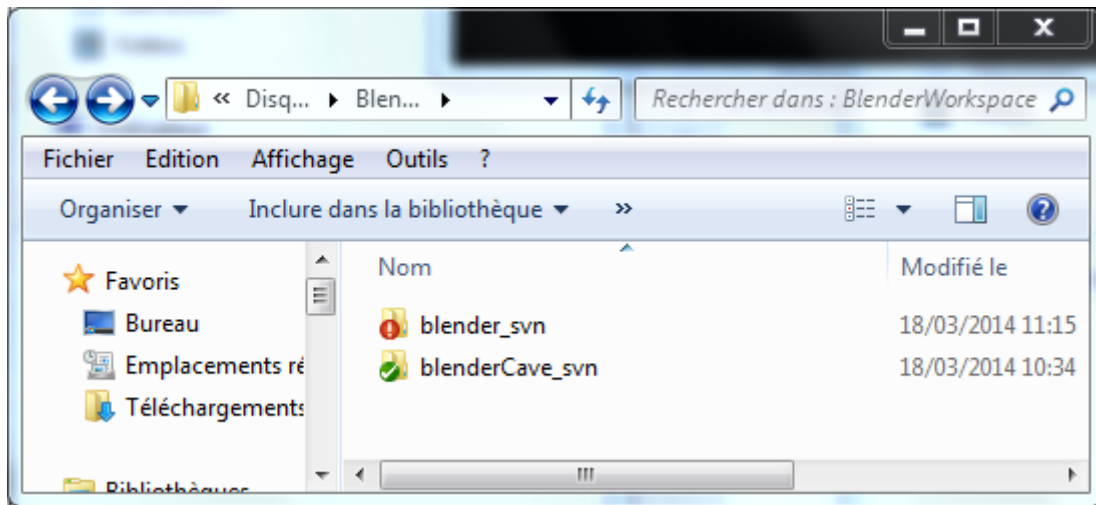
Select the patch file from blenderCave_svn/patch/ directory.



When the list of patched files is opened, click "Patch all items".



When finished, if you update your Windows Explorer, you can see that blender_src/ directory is marked dirty by TortoiseSVN (because its patched).



Compile patched blender

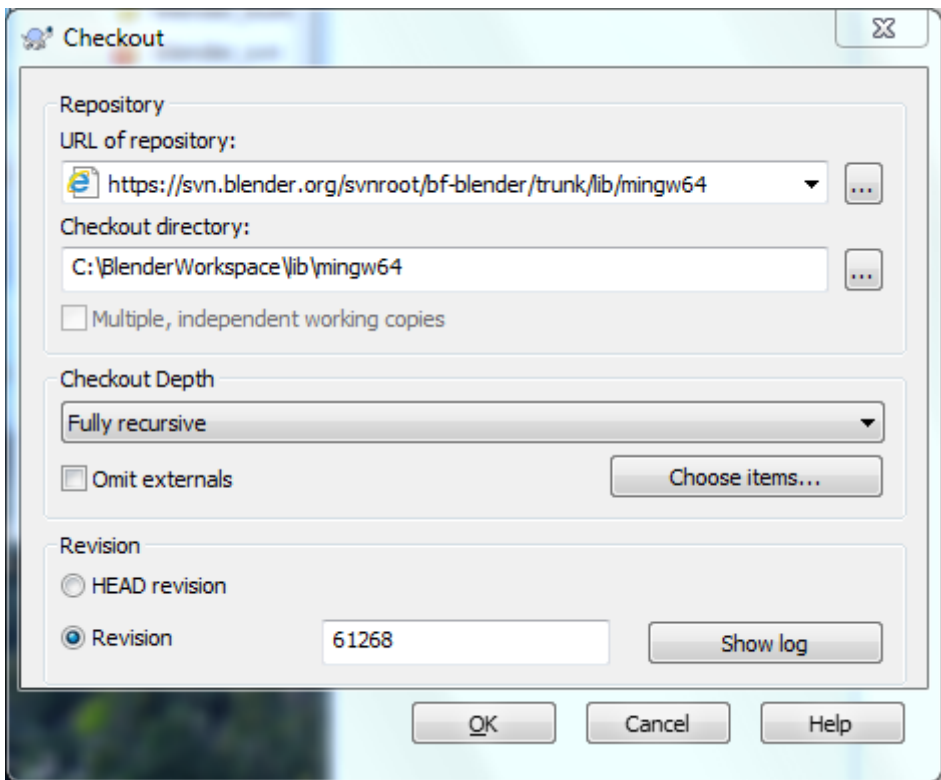
Install compiler and build tools

We use [Visual Studio Express 2013 for Windows Desktop](#).

Download precompiled libraries

Download ad-hoc revision of **precompiled libraries for Visual Studio 2013 64 bits** under Windows, from svn repository https://svn.blender.org/svnroot/bf-blender/trunk/lib/win64_vc12/ and install it into `BlenderWorkspace\lib\win64_vc12`.

Beware to select the revision 61323. The others include wrong libraries.



Prepare makefiles

Within CMake, select Unix makefile.

Follow the [build procedure](#) (adapted to our paths). Here are some notes from our build experience.

We used [mingw-get-setup](#) from [MinGW](#) site.

CMake failed to find numpy, we installed [Python 3.3 from Python site](#) (for win32) in standard C:\Python33\ directory (with PATH modification to find it on command line), and Numpy for Python 3.3, then we adapted CMakeLists.txt to target this Python installation:

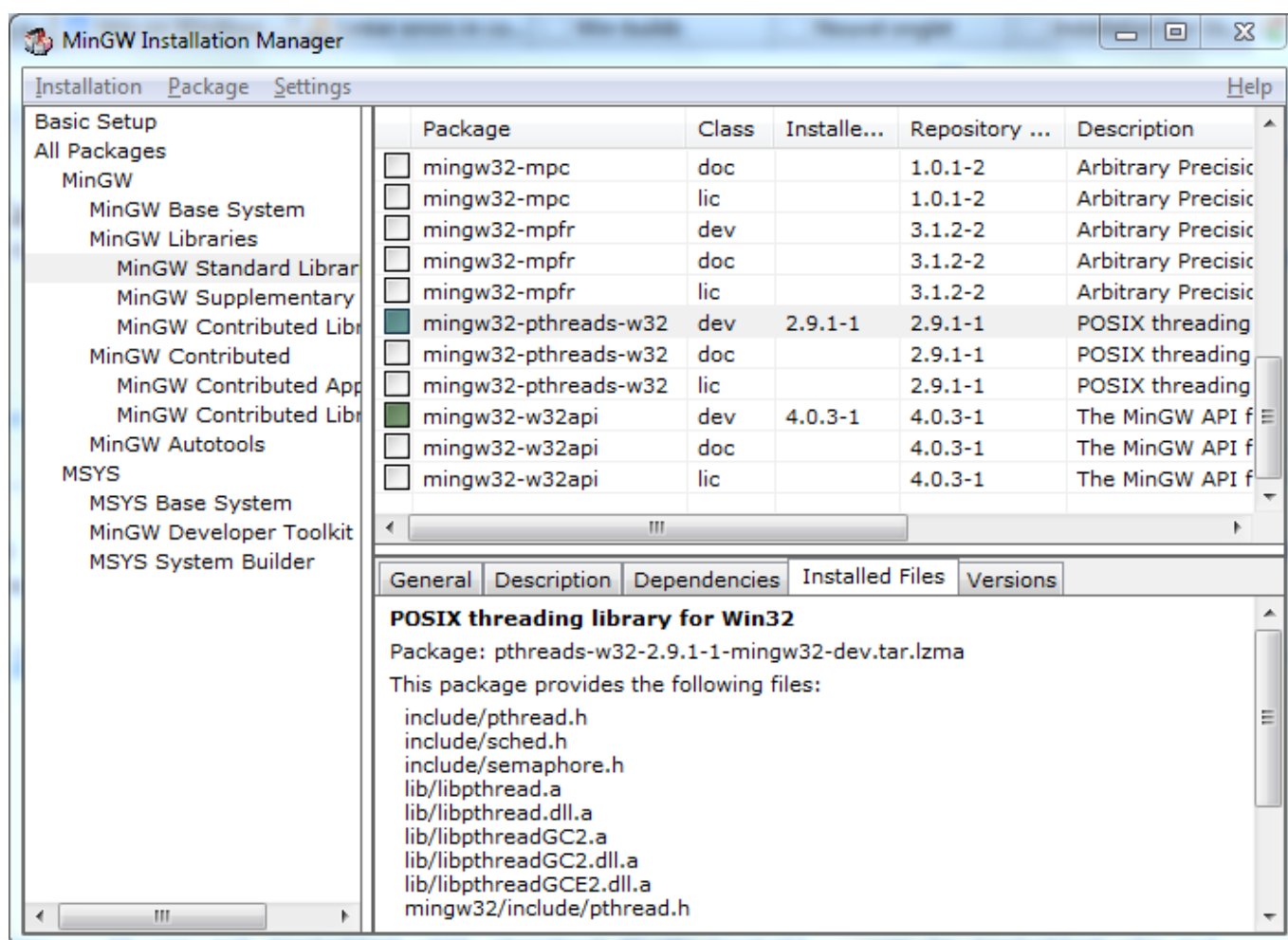
```
--- C:/Users/YOURLOGINHERE/AppData/Local/Temp/CMakeLists.txt-  
revBASE.svn000.tmp.txt mer. nov. 6 02:45:15 2013  
+++ C:/BlenderWorkspace/blender_src/CMakeLists.txt mer. avr. 2 14:04:36  
2014  
@@ -1387,8 +1387,10 @@ elseif(WIN32)  
    if(WITH_PYTHON)  
        # normally cached but not since we include them with blender  
        set(PYTHON_VERSION 3.3) # CACHE STRING)  
-        set(PYTHON_INCLUDE_DIR  
"${LIBDIR}/python/include/python${PYTHON_VERSION}") # CACHE PATH)  
-        set(PYTHON_LIBRARY "${LIBDIR}/python/lib/python33mw.lib") #  
CACHE FILEPATH)  
+        set(PYTHON_INCLUDE_DIR "C:/Python33/include")  
+        #set(PYTHON_INCLUDE_DIR  
"${LIBDIR}/python/include/python${PYTHON_VERSION}") # CACHE PATH)  
+        set(PYTHON_LIBRARY "C:/Python33/libs/python33.lib")  
+        #set(PYTHON_LIBRARY "${LIBDIR}/python/lib/python33mw.lib") #  
CACHE FILEPATH)
```

```

# uncached vars
set(PYTHON_INCLUDE_DIRS "${PYTHON_INCLUDE_DIR}")
@@ -2106,6 +2108,7 @@ if(WITH_PYTHON)
    HINTS
        "${PYTHON_LIBPATH}/python${PYTHON_VERSION}/"
        "${PYTHON_LIBPATH}/python${_PY_VER_MAJOR}/"
+
        "C:/Python33/Lib"
    PATH_SUFFIXES
        site-packages
        dist-packages

```

Missing pthread library - it was not installed with standard MinGW install - can be installed via gui interface, selecting mingw32-pthreads-w32 dev package:



Or via the command line with:

```
mingw-get install mingw32-pthreads-w32-dev
```

Command to build the project, within C:\BlenderWorkspace\blender_build\ directory:

```
mingw32-make
```



From:

<https://blendervr.limsi.fr/> - **BlenderVR**

Permanent link:

https://blendervr.limsi.fr/doku.php?id=doc:build_windows_old

Last update: **22/09/2014 15:52**

