Compile Bforartists for Windows with Cmake and Visual Studio 2017

Table of content

Overview................................................................................................................................................1
Introduction.........................................................................................................................................1
Outline.................................................................................................................................................1
Tool Requirements.................................................................................................................................2
  Required Tools................................................................................................................................2
    Building Tools...............................................................................................................................2
    Sourcing Tools..............................................................................................................................2
    Other Tools....................................................................................................................................2
  Optional Tools....................................................................................................................................2
Downloading the Repository..............................................................................................................3
Creating the Binary Folder................................................................................................................3
Installing the Dependencies................................................................................................................4
Updating the Repository....................................................................................................................7
Running CMake..................................................................................................................................7
  CMake Preparation..........................................................................................................................8
  Activate all needed features...........................................................................................................8
  Generate.........................................................................................................................................9
Troubleshooting Cmake Errors..........................................................................................................10
  Cuda................................................................................................................................................10
  Cmake Warning at intern/cycles/kernel/CmakeLists......................................................................11
  Adding System variables................................................................................................................12
  Missing Language Paths in Blender...............................................................................................12
  Open EXR.....................................................................................................................................13
Compiling............................................................................................................................................14
Congratulations!................................................................................................................................16

Overview

Introduction
This is a step-by-step tutorial packed with tips and tricks to help you compile your first Bforartists build – opening doors to develop your artistic ideas with the source code.

Outline
To compile, we first need to gather the correct tools, materials, and blueprints, and then we can build the “house” - which is Bforartists. The following tutorial will explain each process step by step with tips and tricks to make sure you can troubleshoot and build without any errors.

Once you have no errors in the building process, you will be free to work in a Github branch and commit and push new improvements to Bforartists – with set tasks that can be worked on collaboratively and approved to be included in future official releases.

So let’s begin by getting the right tools, then the correct materials and blueprints – then we can use the tools to build our house.
**Tool Requirements**

**Required Tools**

In order to build Bforartists, you need “building tools” like **compilers**, **“sourcing tools”** for the materials, “blueprints” like the **repository** and **libraries**, and other third party “resources” that are classified as dependencies.

**Building Tools**

- **Microsoft Visual Studio 2017 Community**: [https://visualstudio.microsoft.com/de/downloads/](https://visualstudio.microsoft.com/de/downloads/)

**Sourcing Tools**

- **Git**: [https://git-scm.com/](https://git-scm.com/)
- **Tortoisegit**: [https://tortoisegit.org/](https://tortoisegit.org/)
  
  **Quick Tip**: You can of course also do everything from the Git command line. But Tortoisegit makes things much easier with a GUI.

- **TortoiseSVN** for the dependencies: [http://tortoisesvn.net/index.de.html](http://tortoisesvn.net/index.de.html)

**Other Tools**

- **Python 2.7**: [https://www.python.org/downloads/](https://www.python.org/downloads/)
- **Python 3.x**: [https://visualstudio.microsoft.com/de/downloads/](https://visualstudio.microsoft.com/de/downloads/)
- **Nvidia Optix SDK** (requires registration for download)  
  
  **Quick Tip**: Hitting “Next” “Next” “Next” to install the above is generally ok and risk free.

**Optional Tools**

**We need an installed version of Blender if you want to modify the icons.** You need the Windows Installer version, the one with the *.msi ending. Make sure that the path to the Blender.exe is added to the Windows system variables. You can find all our Windows Installer versions here: [http://download.blender.org/release/](http://download.blender.org/release/)

**Quick tip:** Check out Page 9 to learn how to add System Variables

- **7Z or similar zip software** that can handle tar.gz archives: [http://www.7-zip.org/](http://www.7-zip.org/)

**Nice to have but not necessary is Grepwin to search the source code:**  
[http://stefanstools.sourceforge.net/grepWin.html](http://stefanstools.sourceforge.net/grepWin.html)

**Meld**: [http://meldmerge.org/](http://meldmerge.org/) or **Atom**: [https://atom.io/](https://atom.io/) are also beneficial but not required. They allow you to
use an overview, syntax highlights and document comparisons.

**Downloading the Repository**

So you have downloaded and installed the tools – one after the other. Now it’s time to download the repository (the blueprints) and get the libs (the materials).

1. Choose a location.

   **Quick Tip:** *When possible choose or create a folder directly at the drive level.*

   I have chosen drive H:\, and have created the folder **bforartists**, which is our target folder.

2. Open your browser and navigate to [https://github.com/Bforartists/Bforartists](https://github.com/Bforartists/Bforartists). This brings you to the repository URL recognized by Git.

3. Right click your target folder and choose “git clone” in the right mouse button (rmb) menu.

4. You navigated to the Bforartists Github repository in step 2, so Git should have copied the right link into the URL field already. If not, then type in the URL manually: [https://github.com/Bforartists/Bforartists](https://github.com/Bforartists/Bforartists).

5. Click okay. Git will download the Bforartists repository now.
6. If everything has gone well then you will get the following result (notice the green tickbox):

![Image of file structure]

**Creating the Binary Folder**

On with some more preparations.

You need a folder where **CMake** builds the Binaries and where **Visual Studio** compiles the result. In order to do so, create another folder alongside the repository.

You can name this folder whatever you like. I will call it **bforartists_build**.

![Image of file structure]

**Installing the Dependencies**

What is missing now are the dependencies, or “materials”. Bforartists is a fork of Blender, and uses the same dependencies. It doesn't host and update this dependencies though. We use the official dependencies from Blender.

1. Create a folder called **lib** alongside the repository. This name is important, it must be **lib**!
2. Inside the lib folder create a folder called `windows_vc14` if you want to build a 32 Bit version of Bforartists, or a folder called `win64_vc14` if you want to build a 64 Bit version. Or both. These names are important!

Quick Tip: the 32 Bit version of Bforartists is quickly becoming outdated and is no longer supported officially. It is recommended you build the 64 Bit version.

3. Now, right click on the `win64_vc12` folder, and choose SVN Checkout.

   You could also do this by command line (CMD):

   For 64 Bit:
   
   ```
   svn checkout https://svn.blender.org/svnroot/bf-blender/trunk/lib/win64_vc14
   ```

   For 32 Bit:
   
   ```
   svn checkout https://svn.blender.org/svnroot/bf-blender/trunk/lib/windows_vc14
   ```

Because you have SVN installed, you can use SVN Checkout via the right click GUI instead of the command line.
4. Once the Checkout panel is open, add the following link to get the correct lib:

   **For the 32 Bit version we need the url:**
   https://svn.blender.org/svnroot/bf-blender/trunk/lib/windows_vc14

   **For the 64 Bit version we need the url:**
   https://svn.blender.org/svnroot/bf-blender/trunk/lib/win64_vc14

5. Click okay, and the SVN repository with the dependencies should download.
6. After these steps, the downloaded results in the target folder should now look like this:

Note that the creation of the folder structure for the libs is required so that Cmake finds the libraries automatically.

*Quick tip:* Periodically update these SVN repositories, as they often update, sometimes weekly. If you don’t update these, you may get build errors.

### Updating the Repository

Before you compile you might want to update the repository (the blueprints) to download the latest changes.

1. First “Fetch”. This updates the repository index with branches, changes etc. This command only updates the index, not the repository itself. To update the repository you must do a “Pull”.

2. So first **Fetch**, then **Pull**. Your repository should be up to date again.

### Running CMake

The Materials (Repository and Libraries) and Tools (Compilers) preparation is done using the Sourcing tools
You have downloaded and installed everything. The repository is up to date. Time to fire up Cmake for the first time and create the solution files.

Cmake Preparation

1. Enter the paths to the repository source code and to the build folder then click on Configure.

2. A window will pop up asking you what Visual Studio version to use. Choose Visual Studio 15 2017 in the “Specify the Generator for this project” dropdown box to compile the 64 bit version.

3. Let Cmake run through once. Then click Configure again. The second run will turn the upper window back to normal – where the option toggles are displayed.

4. Sometimes, if there are incomplete libraries, missing or corrupt dependencies, and/or previous damaged builds, you might get errors in the bottom window. If you have errors to resolve, skip ahead to the next chapter.

5. If you got no errors, only warnings, then you can proceed to the next steps:

Activate all needed features.

Some features are off by default. Cuda Binaries, Alembic, Open VDB and Open Subdiv needs to be manually activated. You must activate Cuda to have GPU acceleration in the software. And so on.

The current release settings are:

WITH_BUILDINFO OFF
WITH_CODEC_SNDFILE ON
WITH_CYCLES_CUDA_BINARIES ON
WITH_CYCLES_DEVICE_OPTIX ON
Quick tip: The list for a release version of Bforartists can be found in the build_files/cmake/config folder in the repository. The blender_release.cmake file contains all needed settings. Unfortunately using this with cmake just works on Linux. Even I haven't gotten it to work on Windows. When you are on Windows then you have to tick the items one by one.

Generate

When all errors are fixed Click on Generate. This will now create the solution files in the bforartists_build folder. You can now open the bforartists.sln file in Visual studio.

Quick Tip: Note that the sln file may be called called Blender.sln. Bforartists is a Blender fork. And merging the newest Blender features into Bforartists can always lead to some unwanted side effects ...
Troubleshooting Cmake Errors

You might get errors when you run Cmake for the first time: missing or corrupt dependencies for example. Make sure you have downloaded and installed all of them. You might also have missing path variables by which the installed software gets recognized by Cmake. These are just some possible errors. Feel free to drop a task in the Github tracker or Forum space for Bforartists asking for help if this step is over your head. We will do our best to help you out.

Cuda

You might get a CUDA_SDK_ROOT_DIR-NOTFOUND warning in the upper section. Bforartists can compile with this error present. If you want to fix this, search for nvcc in the C:\Program Files\Nvidia Corporation folder.
Cmake Warning at intern/cycles/kernel/CmakeLists

Sometimes you will get this warning (which can cause compiling errors) if you have the incorrect CUDA Developer Tools installed. Uninstall and reinstall the correct version. 9.1.85 is the last tested version for Bforartists 1.0.0
Adding System variables

Some things are required to be added to the system variables.

Here's an example where you add the path to the Blender exe in case it is missing. Windows System control, Advanced system settings -> System properties.

Missing Language Paths in Blender

When you compile Blender, and not Bforartists, then you might have the above errors with missing language paths. Bforartists does not have this problem anymore, we have united everything in one repository. Blender
still has this issue. Here's how to fix it:

Type in the following commands into the bash console, one after another. Wait until each single task is finished:

```
git submodule update --init --recursive
```
```
git submodule foreach --recursive git checkout master
```
```
git submodule foreach --recursive git pull --rebase origin master
```

This should download the needed submodules. Cmake shouldn't moan anymore.

Later, when you want to update the submodules, type in:

```
git pull --rebase
```
```
git submodule foreach --recursive git pull --rebase origin master
```

This just a sidenote for Blender itself. Bforartists does not need this step.

**Open EXR**

Often the Openexr libraries are not found on Windows, but you can fix them by hand whenever you want. It is not necessary to fix this error to successfully compile and run Bforartists. You can happily ignore the warning. It's just a cosmetic problem in the Cmake GUI. If you choose to fix them by hand, then don't mix the libs for the 32 bit version with the libs for the 64 bit version.
Finally you arrive at the compiling part, the best part. Cmake should have produced everything needed in the build folder.
1. Click on the Bforartists.sln file to open Visual Studio.

2. In VS, change the build type to **Release** (VS starts usually with **Debug**, but that's only of interest for development).

3. Select the **Install** item in the Solution Explorer, right click on it, and choose **Build**.

4. Visual Studio should now start the Build process.

5. When everything works as planned then you should have a result in the **bin/Release** folder.
Congratulations!

You have compiled Bforartists for the first time. You should be able to work with the source code now. Double click on the *.exe to run and see that everything is working fine. When it does, you are a winner.

**Quick Tip:** Now that it’s compiled, you can do Python code changes to the built data and test it almost live. Don’t forget to copy your python changes in the bin/Release folder back to the Repository or else when you rebuild, you will loose all your changes!

For C code changes, you will need to make the changes in the repository before building, then re-build to test. If you get errors on re-build, just delete the old build in the bin/Release folder.