

Doc:Reference ▾

- ...map
- + Buttons
- ...Color Picker
- + Command Line
- ...Doc edit in the UI
- + Glossary
- + Hotkeys
- + Menus
- + Panels
- ...QuickStart
- + Windows

Blender can also be run from the command line. In the following I will assume the you are in Blender's directory or that the Blender executable is in your PATH. This second option is the preferred one. Please refer to your friendly OS manual to learn what a PATH is, and how to have Blender in it. Plain *blender* runs the whole thing just as if you clicked the Blender icon.

## Basic Options

Adding command line arguments can force different behaviours. The general syntax is:

```
blender [options] [file]
```

The basic option to remember is *-h*, that is the help option... once you remember this Blender will tell you all the others... a sort of magic word.

```
blender -h
```

Yields:

```
Blender 2.45 (sub 17) Build
Usage: blender [args ...] [file] [args ...]

Render options:
-b <file> Render <file> in background (doesn't load the user defaults .B.blend file)
-a render frames from start to end (inclusive), only works when used after -b
-S <name> Set scene <name>
-f <frame> Render frame <frame> and save it
-s <frame> Set start to frame <frame> (use before the -a argument)
-e <frame> Set end to frame <frame> (use before the -a argument)
-o <path> Set the render path and file name.
    Use // at the start of the path to
    render relative to the blend file.
    The # characters are replaced by the frame number, and used to define zero padding.
    ani_##_test.png becomes ani_01_test.png
    test-#####.png becomes test-000001.png
    When the filename has no #, The suffix ##### is added to the filename.
    The frame number will be added at the end of the filename.
    eg: blender -b foobar.blend -o //render_ -F PNG -x 1 -a

Format options:
-F <format> Set the render format, Valid options are...
    TGA IRIS HAMX JPEG MOVIE IRIZ RAWTGA
    AVIRAW AVIJPEG PNG BMP FRAMESERVER
    (formats that can be compiled into blender, not available on all systems)
    HDR TIFF EXR MULTILAYER MPEG AVICODEC QUICKTIME CINEON DPX DDS
-x <bool> Set option to add the file extension to the end of the file.
-t <threads> Use amount of <threads> for rendering.
    [1-8], 0 for systems processor count.

Animation playback options:
-a <file(s)> Playback <file(s)>, only operates this way when -b is not used.
-p <sx> <sy> Open with lower left corner at <sx>, <sy>
-m Read from disk (Don't buffer)
-f <fps> <fps-base> Specify FPS to start with

Window options:
-w Force opening with borders (default)
-W Force opening without borders
-p <sx> <sy> <w> <h> Open with lower left corner at <sx>, <sy>
    and width and height <w>, <h>

Game Engine specific options:
-g fixedtime Run on 50 hertz without dropping frames
-g vertexarrays Use Vertex Arrays for rendering (usually faster)
-g noaudio No audio in Game Engine
-g nomipmap No Texture Mipmapping
-g linearmipmap Linear Texture Mipmapping instead of Nearest (default)

Misc options:
-d Turn debugging on
-noaudio Disable audio on systems that support audio
-h Print this help text
-y Disable script links, use -Y to find out why its -y
-P <filename> Run the given Python script (filename or Blender Text)
-v Print Blender version and exit
-- Ends option processing. Following arguments are
```

```
passed unchanged. Access via Python's sys.argv
```

#### Environment Variables:

```
$HOME    Store files such as .blender/ .B.blend .Bfs .Blog here.
$TMP or $TMPDIR Store temporary files here.
$SDL_AUDIODRIVER LibSDL audio driver - alsa, esd, alsa, dma.
$BF_TIFF_LIB Use an alternative libtiff.so for loading tiff image files.
$IMAGEEDITOR Image editor executable, launch with the IKey from the file selector.
$WINEDITOR Text editor executable, launch with the EKey from the file selector.
$PYTHONHOME Path to the python directory, eg. /usr/lib/python.
```

Note: Arguments must be separated by white space. eg:

```
"blender -ba test.blend"
...will ignore the 'a'
"blender -b test.blend -f8"
...will ignore 8 because there is no space between the -f and the frame value
```

Note: Arguments are executed in the order they are given. eg:

```
"blender -b test.blend -f 1 -o /tmp"
...may not render to /tmp because '-f 1' renders before the output path is set
"blender -b -o /tmp test.blend -f 1"
...may not render to /tmp because loading the blend file overwrites the output path that was set
"blender -b test.blend -o /tmp -f 1" works as expected.
```

## Render Options

The most important set of options. They allow you to do *background rendering*. This implies that you render an image without having to run the Blender GUI. This uses some less memory and, usually, much less CPU time. Most parameters for rendering need to have been set interactively before via the GUI, however some important parameters, such as file type and output path can now be set from the command line. However, note that the behavior of these options is somewhat unconventional. Options are treated as individual commands, executed in order, with the frame number option rendering the frame. Therefore, it is necessary to list your parameter setting options *before* the frame number option, otherwise the parameters will not be set when the image is rendered.

To render a still image (assuming the still is in frame number 1):

```
blender -b yourfile.blend -f 1
```

To render an animation going from frame 1 to frame 100:

```
blender -b yourfile.blend -s 1 -e 100 -a
```

To render a still image (frame 1) to a PNG file (overriding whatever file format is set in the .blend file itself):

```
blender -b yourfile.blend -F PNG -f 1
```

whereas the following case will not work; the image will be rendered to the file format listed in the .blend file:

```
blender -b yourfile.blend -f 1 -F PNG
```

You can also use an external python file to render files

```
blender -b yourfile.blend -P yoursript.py
```

the script should look something like

```
import Blender
from Blender import *
from Blender.Scene import Render

scn = Blender.Scene.Get("Scene")
context = scn.getRenderingContext()

context.extensions = True
context.renderPath = "c:\\temp"
context.sizePreset(Render.PAL)
context.imageType = Render.TARGA
context.fps = 25
context.sFrame = 1
context.eFrame = 100

context.renderAnim()
```

More RenderContext can be found here <http://www.blender.org/documentation/243PythonDoc/Render.RenderData-class.html>

## Animation Options

These allow you to play back an animation:

```
blender -a yourfile.avi
```

Useful if you used Blender's built in Jpeg Codec, and don't know how to play back the AVI.

## Window Options

---

These options force Blender to open its window with specified dimensions. *-W* forces the default full-screen opening. *-w* forces a non-full screen opening. If no other parameters are given the window occupies all the screen the same, but it is not 'maximized'. If the *-p* switch is given, the position and dimension of the window can be specified.

```
blender -w -p 128 128 1024 768
```

Forces Blender to open as a 1024x768 window with its lower left corner in (128,128).

## Other Options

---

The Game Engine options are not covered in this manual. The Miscellaneous Options turns on debugging features (*-d*), suppress audio (*-noaudio*), disable automatic running of 'On Load' scripts (*-γ*), register '.blend' extension in the Operating System (*-R*) and provides the list of all options (*-h*).

Category: [References](#)