# V-Ray Standalone Command Line Options

This page introduces command-line parameters for V-Ray Standalone.

## Overview

Below is a list of the command-line parameters for V-Ray Standalone. You can use the **VRAY_CMD_PREFIX** and **VRAY_CMD_SUFFIX** environment variables to specify additional command-line parameters – see the V-Ray Standalone Environment Variables page. If you run V-Ray Standalone without any parameters, it will also print a short help text with descriptions of each possible option. Portions of the options in square brackets [] are optional and portions in curly braces {} can be repeated zero or more times.

*See the Getting Started with Commands page for practical examples to get you started.*

## System Options

**-help** – Prints out a list of all the available command lines and their usage.

**-version** – Prints the V-Ray version.

**-credits** – Prints V-Ray and 3rd party copyright notices.

**-configFile="path/to/config/file"** – The path and file name to the V-Ray config file, which contains paths to the V-Ray plugins. Note that additional paths for V-Ray plugins can also be specified with the VRAY_PLUGINS_x86 and VRAY_PLUGINS_x64 environment variables. The default is vrayconfig.xml in the same folder as vray.exe.

**-numThreads=nnn** – Sets the number of rendering threads. The default is 0 - automatic number of threads.

**-profilerMode=0/1/2** – Specifies the mode of the V-Ray Profiler:

  0 - disabled
  1 - simplified output
  2 - full output

The default is 0 - disabled.

**-profilerOutputDir=\directoryPath\** - Specifies the output directory for the V-Ray Profiler results. If the parameter is empty or is omitted, the OS's default temp directory is used.

## Input Files

**-sceneFile="filename.vrscene"** – Specifies the scene file to render.

**-include="includePath{;includePath}"** – Specifies path(s) for include files. More than one **-include** options can be specified.

**-remapPath="fromPath=toPath{;fromPath=toPath}"** – Specifies path remapping pair. More than one **-remapPath** option can be specified.

**-remapPathFile="remapFile.xml"** – Specifies path to an XML file with path remapping data. Alternatively, the **VRAY_PATH_REMAP_FILEPATH** environment variable can be used to specify the path to the XML file (see the V-Ray Standalone Environment Variables page for more information). Example file:

```
<RemapPaths>
    <RemapItem>
        <From>Z:/export</From>
        <To>/mnt/export</To>
    </RemapItem>
</RemapPaths>
```

**-parameterOverride="<plugin-parameter-id>=<value>"** – Makes it possible to change values for plugin parameters without editing the vrscene file.

**<plugin-parameter-id>** – Can be of two formats - *PluginType::parameterName* or *pluginName.parameterName*.
**<value>** – Its format depends on the type of the parameter you want to override. Supported types are *bool*, *int*, *float*, *color*, *acolor*, *texture* and *float texture*. For bool, int and float a simple number is expected. For *color* - Color(1, 0.0, 1.0) is expected. For *acolor* - AColor(1.0, 0, 0, 0.3). For *texture* the format should be the same as for *acolor* or the name of a plugin connected to the parameter. For *float texture* parameters, either one float or a name of a plugin is expected. More than one **-parameterOverride** option can be specified and will be executed in the order specified in the command line. Examples:

```
-parameterOverride="SettingsImageSampler::type=3"
-parameterOverride="TexFresnel::fresnel_ior=1.3"
-parameterOverride="VRayMtl1@diffuse.color=Color(1.0,0.2,1.0)"
-parameterOverride="VRayMtl2@diffuse.color_tex=checker1"
```

**-camera="<sceneName>"** – Specifies which camera should be used for rendering with multiple cameras. The name should correspond to the camera settings scene_name parameter. Examples (where *InteriorCamera, ExteriorCamera*, and *VRayCam001* are names of existing cameras in the same vrscene file):

```
-camera="InteriorCamera"
-camera="ExteriorCamera"
-camera="VRayCam001"
```

Note that for the **-camera** option:

- For scenes with multiple Standard Physical cameras and Exposure control installed, the render is physically correct only if rendered with the active view.

## Render Output

**-imgFile="<path>/fileName.<extension>"** – Writes the resulting image to the given file. If the file name is empty, no image will be written. Alternatively, files can be saved without using this option in cmd if path (img_dir) and filename (img_file) are defined in the *.vrscene* output settings content.

**-imgWidth=nnn** – Sets the output image width. The default is 640 or as specified in the *.vrscene* file.

**-imgHeight=nnn** – Sets the output image height. The default is 480 or as specified in the **.vrscene** file.

**-region=x0;y0;x1;y1** – Sets region to render. The integer values are in pixels, relative to the upper-left corner of the image. A region might be specified in the *.vrscene* file itself and it will be used in the rendering, unless the **-region** flag is used to override it or the **-region=none** flag is used to disable it.

**-region=none** – Ignores any region specified in the *.vrscene* file

**-crop=x0;y0;x1;y1** – Sets the crop region to render. The integer values are in pixels, relative to the upper-left corner of the image. The default is full image or as specified in the *.vrscene* file.

**-frames=b0[-e0[,s0]]{;bn[-en[,sn]]}** – Specifies the frames and/or frame intervals to render. More than one **-frames** option can be specified: bX is the start frame, eX is the end frame, and sX is the frame increment. If not specified, the frames from the scene description are rendered.

| Example | Description |
|---|---|
| -frames=1;3;5-10 | Render frames 1, 3 and 5 to 10 inclusively |
| -frames=10-100,5 | Render every 5-th frame starting from 10 to 100 (i.e. 10, 15, 20, 25, ..., 95, 100) |

**-noFrameNumbers=0/1** – Controls whether the **-frames** option will cause frame numbers to be automatically appended to the names of the rendered image files. The default value is 0 where the **-frames** option always causes frame numbers to be added to the output image files.

**-skipExistingFrames=0/1** – Controls whether V-Ray renders frames from the specified frame range that currently exist in the output folder.

**-resume=0/1** – V-Ray will resume rendering partially completed renders. This is only for bucket renders (non-deep images) when writing to *.vrimg*.

**-progressiveAutoSave=** – Specifies an interval in minutes for saving resumable files during rendering. This applies only for progressive sampling; bucket renderings are saved on every completed bucket. A value of 0 disables autosaving during rendering, and resumable files will be saved only at the end of the rendering.

## VFB (V-Ray Frame Buffer) Display Options

**-display=0/1** – Shows the rendered image in the V-Ray frame buffer. The default value is 1, which shows the rendered image.

**-autoClose=0/1** – Automatically closes the displayed image when rendering is complete. The default value is 0, which waits for user to close the VFB.

**-setFocus=0/1** – Sets the focus on the VFB window if displayed. The default value is 1 which sets the focus on the VFB window. This works reliably only on Windows.

**-displaySRGB=0/1/2** – Determines if the sRGB option of the V-Ray VFB will be turned on or off initially. A value of 0 means the default value for the VFB will be used (which usually has sRGB enabled). A value of 1 forces the sRGB button to be enabled. A value of 2 forces the sRGB button to be disabled. This can also be specified with the **VRAY_VFB_SRGB** environment variable. The command line option overrides the environment variable. The default is 0, where sRGB is the VFB default (enabled).

**-displayLUT=0/1** – Determines if the LUT option of the V-Ray VFB will be turned on or off initially. This can also be specified with the **VRAY_VFB_LUT** environment variable. The command line option overrides the environment variable. The LUT file itself can be specified with the **VRAY_VFB_LUT_FILE** environment variable. The default value is 0, where LUT is off.

**-displayOCIO=0/1** - Determines if the OCIO option of the V-Ray VFB will be turned on or off initially. This can also be specified with the **VRAY_VFB_OCIO** environment variable. The command line option overrides the environment variable. The OCIO configuration itself can be specified with the OCIO environment variable. The default value is 0, where the OCIO is off.

**-displayAspect=0/1** – Determines if the pixel aspect option of the V-Ray VFB will be turned on or off initially. This can also be specified with the **VRAY_VFB_PIXEL_ASPECT** environment variable. The command line option overrides the environment variable. The default value is 0, where pixel aspect is off.

**-deleteResumableFileOnSuccess=0/1** – Automatically deletes the resumable file if the frame is completed successfully. Applies to additional *.vrimg* or *.vrprog* files, and does not apply when the output image file is *.vrimg*. The default value is 0, which keeps the resumable file.


## Console Output

**-verboseLevel=n** – Specifies the verbose level of information printed to the standard output:

    0 – No information is printed
    1 – Only errors
    2 – Errors and warnings
    3 – Errors, warnings, and informational messages
    4 - All output

    The default is 3.

**-detailedTiming=0/1** – Specifies whether summary timing information would be printed to the standard output at the end of execution.

    0 – Disabled (the default value);
    1 – Enabled;

**-showProgress=n** – Specifies whether calculation progress should be printed to the standard output:

    0 – Do not display progress
    1 – Display progress only if **verboseLevel** is > 0
    2 – Always display progress

    The default is 1.

**-progressUpdateFreq=n** – Specifies the render progress update frequency in milliseconds. The default value is 200 ms. Only one of **-progressUpdateFreq** and **-progressIncrement** is used, depending on which is specified last on the command line.

**-progressIncrement=n** – Specifies the progress increment in percentage when a new progress message will be printed. Only one of **-progressUpdateFreq** and **-progressIncrement** is used, depending on which is specified last on the command line.

**-progressUseCR=0/1** – Controls how to use carriage return when printing render progress information.

**-progressUseColor=0/1** – Specifies whether to colorize the output:

    0 – Disabled
    1 – Enabled (the default value)


## RT Engine

**-rtEngine=0/1/3/5/7** – Specifies which rendering engine is going to be used:

    0 – The regular render engine is used.
    1 – The CPU RT engine is used.
    3 – (Deprecated) The GPU RT engine running on OpenCL is used.
    5 – The GPU RT engine running on CUDA is used.
    7 – The GPU RT engine running on RTX is used.

The default is 0.

**-rtTimeOut=fff** – Specifies a floating-point render time value (in minutes) when using the RT engine. The default is 0.0, no time limit.

**-rtNoise=fff** – Specifies floating-point noise threshold for a frame when using the RT engine. The default is 0.001.

**-rtSampleLevel=nnn** – Specifies maximum number of paths per pixel for a frame when using the RT engine. The default is 0, no limit.

# Distributed Render

## Server

**-server** – Start in server mode waiting for connections.

**-autoCloseIfIdle=n** – Specifies the minutes the server will wait in idle state before closing itself. A value of 0 means this feature is disabled, otherwise it is the idle time in minutes.

## Client

**-distributed=0/1/2** – Specifies whether and how to use distributed rendering:

> 0 – Distributed rendering is disabled.
> 1 – Distributed rendering is enabled, and the local machine is included in the rendering.
> 2 – Distributed rendering is enabled and the local machine is excluded from the rendering.
>
> The default is 0.

**-renderhost="host{;host}"** – A semicolon separated list of hosts to use for distributed rendering. Hosts can be specified either as computer names or by IPs. More than one -renderhost options can be specified.

**-transferAssets=0/1/2** – Specifies whether missing assets should be transferred to the DR servers:

> 0 – Do not transfer.
> 1 – Transfer missing assets.
> 2 – Check for assets in the asset cache folder before transferring them. The VRAY_ASSETS_CACHE_PATH environment variable can be used to specify the asset cache folder; if it is not specified, the assets are stored in a directory "vray_assets_cache" inside the temp folder for the current user.
>
> The default is 0.

**-cachedAssetsLimitType=0/1/2** – Type of limitation used for cached assets on DR server:

> 0 – No limits (the default value);
> 1 – Older than: assets older than specified period of hours will be deleted;
> 2 – Cache size: if the assets cache grow over the specified size (GBs), the oldest assets will be deleted from cache.

**-cachedAssetsLimitValue=fff** – Specifies limit value for a cached assets. *fff* is a floating-point number representing the value. The meaning of the value depends on the -cachedAssetsLimitType parameter. The default value is 0.0.

**-overwriteLocalCacheSettings=0/1** - Specifies which settings for asset cache limitations to be taken into account.

> 0 – Takes the server's settings. This is the default value. It is recommended to use the server's settings for assets cache limitations;
> 1 – Takes the client's settings.

**-limitHosts=N** – Limits the number of render hosts used for distributed rendering to the first *N* idle hosts specified in the -renderHost options. The default value is 0 - use all available hosts.

## Common

**-portNumber=n** – Specifies the port number to use for distributed rendering. The port numbers of the render servers and the render client must match for DR to work. The port can also be overridden using the VRAY_DR_CONTROLPORT environment variable. This command line option overrides the environment variable. The default is 20204.