Name

recordMyDesktop − record desktop sessions to an Ogg Theora video file with Vorbis audio

Synopsis

recordmydesktop

recordmydesktop *output-filename*

recordmydesktop −−rescue *path-to-data*

recordmydesktop {*image-options*|*sound-options*|*encoding-options*|*miscellaneous-options*| ... }

recordmydesktop {**−h**|**−−help**}

recordmydesktop −−version

recordmydesktop −−print−config

Description

recordMyDesktop produces a file (*out.ogv* by default) containing a video and audio recording of an X Window System session.

For a typical scenario, recording your session is as simple as

$ **recordmydesktop**

which will produce a full-screen recording named *out.ogv* in the current working directory, while the command

$ **recordmydesktop ../foo**

will save the recording in *foo.o gv* in the parent of the current working directory.

To end a recording, press **Control+C**. (This action sends a**SIGINT** to **r ecordMyDesktop**.)

To designate a region of the screen for recording you can type this:

$ **recordmydesktop −x** *x-pos* **−y** *y-pos* **−−width** *w* **−−height** *h* **−o foo.ogv**

where *x-pos* and *y-pos* specify the offset in pixels from the upper-left corner of your screen and *w* and *h* the size of the region to be recorded, again in pixels. If the area extends beyond your current resolution, you will be notified appropriately and nothing will happen.

Note: If *any* options are used, and you wish to use an output filename other than the default, you must specify it with the **−o** option.

Since version 0.3 of **recordMyDesktop**, encoding is done immediately after recording finishes. While this behavior saves a lot of CPU, you can revert to the old one by specifying the **−−on−the−fly−encoding** option.

The default behavior of recording is to mark areas that have changed (as determined by the X Damage extension) and update the frame. This behavior can be changed through the option**−−full−shots** to produce a more accurate result or capture windows that do not generate X events on change, such as windows with accelerated 3D context, but this will notably increase the workload.

recordMyDesktop doesn't have a user interface in the conventional sense. After startup, it can be controlled only through the following signals:

SIGUSR1
: causes the program to pause if it's currently recording, and vice-versa.

SIGTERM
SIGINT
: cause normal termination of the recording.

SIGABRT
: terminates the program and removes the output file.

These signals can also be delivered to the application through the configuration of keyboard shortcuts. See **−−pause−shortcut** and **−−stop−shortcut** in the **Miscellaneous** subsection of the **Options** section below.

For further manipulation of the end result, consult the **Options** and **Notes** sections below.

Options

If *and only if* no options are specified, **recordMyDesktop** accepts a single operand which it interprets as an output filename.

recordMyDesktop accepts many command-line options. The following discussion groups them into families.

General

−h or **−−help**
Print a summary of usage and exit.

−−version
Print program version and exit.

−−print−config
Print information about options selected during compilation and exit. Currently indicates whether Jack capture is enabled and if ALSA or OSS is used; may report more information in the future.

Image

−−windowid *wid*
Record the region of the screen corresponding to the X window *wid* at the time **recordMyDesktop** starts. Recording a window doesn't track the window itself, but instead the region that it covers; if the window is moved, **recordMyDesktop** does not follow it. If the window is obscured by another, the obscuring window's contents within the recording region are captured. An X client's window ID can be obtained with the **xwininfo**(1) program; see the **Notes** section below.

−−display *x-display*
Connect to the X Window System display at *x-display*. By default, the value of the **DISPLAY** environment variable are used to the determine the display.

−x *x-offset*
Restrict the recording region to start at *x-offset* pixels from the left edge of the screen or the X window ID designated with **−−windowid.** The default is 0.

−y *y-offset*
Restrict the recording region to start at *y-offset* pixels from the top edge of the screen or the X window ID designated with **−−windowid.** The default is 0.

−−width *w*
Restrict the recording region to *w* pixels in width. The default is the full width of the screen or the X window ID designated with **−−windowid.**

−−height *h*
Restrict the recording region to *h* pixels in height. The default is the full height of the screen or the X window ID designated with **−−windowid.**

−−dummy−cursor *color*
Draw a dummy cursor instead of the normal one. The given *color* must be a string, either "**black**" or "**white**".

−−no−cursor
Disable drawing of the cursor.

−−no−shared
Do not use the MIT shared memory extension (MIT-SHM) ⟨https://www.x.org/releases/X11R7.7/doc/xextproto/shm.html⟩. *Use of this option is not recommended.*

−−full−shots
Take a full screenshot at every frame. *Use of this option is not r ecommended.*

−−follow−mouse
When this option is enabled, the capture area follows the mouse cursor. This is meaningful only when the selected area is a subset of the full screen. This option implies **−−full-shots**.

−−quick−subsampling
Sub-sample chroma planes by discarding extra pixels.

−−fps *n*
Attempt to capture *n* frames per second, where *n* is positive and may be an integral or floating-point number. The achievable frame rate is limited by your system's hardware and workload. The default is 15.

Sound

−−channels *n*
Record up to *n* channels, where *n* is a positive integer. The default is 2 (stereo).

−−freq *n*
Record at a sample rate of *n* hertz, where *n* is a positive integer. The default is 22050 (22.05 kHz).

−−buffer−size *n*
Buffer *n* bytes of sound data when using the ALSA or OSS sound systems, where *n* is a positive integer. The default is 4096 (4 kiB).

−−ring−buffer−size *n*
Set the size of the JACK ring buffer in seconds, where *n* is a positive floating-point number. The default is 3.0.

The total size of the buffer in bytes can be expressed as a C language expression:
`ring_buffer_size * sample_rate * number_of_ports * sizeof(jack_default_audio_sample_t),`
where `sizeof(jack_default_audio_sample_t)` is normally 4.

−−device *sound-device*
Capture audio from *sound-device*. The default is **hw:0,0** or */dev/dsp*, depending on whether ALSA or OSS is used, respectively.

−−use−jack *port ...*
Record audio from the specified list of space-separated JACK ports. When using this option, all other audio-related options (except **−−no-sound**) are ignored.

−−no−sound
Do not record sound.

Encoding

−−on−the−fly−encoding
Encode the audio/video data while recording. The default is to defer encoding until after recording is complete.

−−v_quality *n*
Set the desired video quality to *n*, an integer between 0 and 63 inclusive; higher is better. The default is 63. See the **Notes** section below.

−−v_bitrate *n*
Set the desired bit rate of the encoded video to *n* bits per second. The accepted range of *n* is between 45,000 and 2,000,000 inclusive; the default is 45,000. (The thousands separator is shown here for clarity; do not use it in the option argument.)

−−s_quality *n*
Set the desired audio quality to *n*, an integer between -1 and 10 inclusive; higher is better. The default is 10.

Miscellaneous

−−rescue *path-to-data*
Encode cache data from a previous session into an Ogg Theora file. The filename will be the one that was chosen initially. Any other option specified with this one will be implicitly ignored and **recordMyDesktop** will exit after the end of the encoding. This option was added in **recordMyDesktop** 0.3.7 and it will not work with cache files produced from earlier versions. When using this option, note that **recordMyDesktop**'s cache is not safe with respect to data type sizes and endianness. To locate the cached data, see the **−−workdir** option below.

−−no−wm−check
When a 3D compositing window manager is detected, the program will function as if the **−−full−shots** option had been specified. This option disables the check.

−−no−frame
Normally, a frame is drawn around the recording region to assist the user to identify what does and doesn't get captured. When the **−−follow−mouse** is given, this frame will also follow the pointer around. However, the frame might cause problems for drag-and-drop operations. If you do not wish this frame to appear, use this option.

−−pause−shortcut *mod+key*
Designate a key combination to toggle the pause state of the recording. *mod* can be any combination of the following modifier keys: Control, Shift, and Mod1 to Mod5. The modifiers can be separated by any character (*except a space*) or none at all. *k ey* can be any key defined in */usr/include/X11/keysymdef.h*. Omit the XK_ prefixes to the key names found in that file. The list of modifiers must be separated from the key with a plus sign "+". The default is **Control+Mod1+p**. (Mod1 usually corresponds to the left Alt key).

−−stop−shortcut *mod+key*
Designate a key combination to stop the recording. For details, see **−−pause−shortcut** above. The default is **Control+Mod1+s**.

−−compress−cache
Image data are cached with a light compression.

−−workdir *dir*
recordMyDesktop creates a temporary directiory in *dir* to cache intermediate files. The default is */tmp*.

−−delay *n*[**H**|**h**|**M**|**m**]
Wait *n* units of time (seconds, if not specified) before starting the recording, where *n* is a non-negative integral or floating-point value. An optional suffix of **H** or **h**, **M** or **m**; indicates time units of hours or minutes, respectively. The default is no delay.

−−overwrite
If the output filename already exists, delete it. The default is to refuse to overwrite, interpolating a numeric discriminator into the filename instead (see the **Files** section below).

−o *filename*
Write the encoded video to *filename*. The default is *out.ogv*.

Exit Status

An exit status of 0 means success per the usual Unix conventions.

A non-zero exit status means that an error occurred; these are accompanied by diagnostic messages to the standard error. Furthermore, several exceptional conditions are distinguished by exit status.

1 Argument parsing failed or argument input was nonsensical.

2 Encoder initialization failed (Vorbis or Theora).

3 Could not open or configure sound device.

4 XDamage extension is not present.

5 MIT-SHM extension is not present.

6 XFixes extension is not present.

7 **XInitThreads**(3) failed.

8 The **DISPLAY** environment variable is not set and no **−−display** option was specified.

9 Could not connect to the X server.

10 The color depth of the root window is not 32, 24, or 16 bits per pixel.

11 The recording window specification was invalid.

12 Could not attach shared memory to process.

13 Could not open file for writing.

Environment

DISPLAY

specifies the X server to connect to. For typical use cases, this variable is already set correctly in the environment and corresponds to the X display the user is already interacting with (in other words, **recordMyDesktop** "does the right thing"). See section **Display Names** in **X**(7) for more information.

Files

recordMyDesktop adds the suffix *.ogv* to any output filename that does not already possess one (in any letter case).

If you try to save to a filename that already exists, a numeric discriminator of the form *−n* is interpolated into the filename before the *.ogv* suffix. The number *n* starts at 1 and is incremented if the filename with that discriminator exists already.

For example, when not specifying a name, if *out.ogv* exists, the new file will be *out−1.ogv*, and if that exists too, *out−2.ogv* is used, and so on *ad infinitum* (not really, more like *ad unsigned short*).

Notes

Recoding using an X window ID

When using the **−−windowid** option, read carefully its description in the **Options** section above, as well as those of the **−x**, **−y**, **−−width**, and **−−height** options.

An easy way to find out the ID of a window is by using the **xwininfo**(1) program. Running a command like

$ **xwininfo | awk '/Window id:/ {print $4}'**

will report only the window ID, which will be a hexadecimal integer in C literal format, like "0x4800005". More conveniently, you can put all that in the command that launches **recordMyDesktop** like this:

$ **recordmydesktop −−windowid $(xwininfo | awk '/Window id:/ {print $4}')**

Tuning the video quality

The lower the quality you select on a video recording (by using the **−−v_quality** option), the more CPU performance you will require. So if you are also using the **−−on−the−fly−encoding** option, it's better to start with default values and manipulate the end result with another program. An excellent converter is the **vlc**(1) media player, which can perform a variety of transcoding operations, using either a graphical interface or the command line. **vlc** is a comple x piece of software; you should consult its documentation before using it. An example follows which will resize a recording named *out.ogv* to 512×384.

$ **vlc −I dummy out.ogv vlc:quit −−sout "#transcode{ vcodec = theo, width = 512, height = 384 }:duplicate{ dst = std{ access = file, mux=ogg, dst = \"out_512x384.ogv\" }}"**

If you wish to change the video quality you can append the option **−−sout−theora−quality=***n*, with *n* in the range [1,10]; for example,

$ **vlc −I dummy out.ogv vlc:quit −−sout "#transcode{ vcodec = theo, width = 512, height = 384 }:duplicate{ dst = std{ access = file, mux=ogg, dst = \"out_512x384.ogv\" }}" −−sout−theora−quality=3**

produces a file of a video quality 18 (in a range of 0–63), appropriate for web publishing.

Another option is **ffmpeg2theora**(1), which—despite its name—is also a Theora-to-Theora converter. Changing the quality of a recordng with it can be as simple as:

$ **ffmpeg2theora infile.ogv −v 3 −a 4 −o outfile.ogv**

It can even perform resizing on the geometry of the recording, or change the overall duration.

Bugs

- Does not record 3D windows if **−−full−shots** isn't specified.
- Saving 65536 files with the same name will result in unpredictable behavior, which might manifest as an endless loop or a segmentation fault.

Authors

John Varouhakis ⟨johnvarouhakis@gmail.com⟩

See Also

X(7), **ffmpeg2theora**(1), **jack_lsp**(1), **vlc**(1), **xwininfo**(1)