# Linux DVD HOWTO

# Table of Contents

# Linux DVD HOWTO

## http://www.linuxvideo.org/

v1.25, 2000−10−17

---

*A (hopefully) easy to follow explanation on how to get DVD movie playback in Linux.*

---



---

# 1. Introduction

## 1.1 Goal

The purpose of this is to provide step by step instructions on getting DVD movies to play in Linux. I will try to be as clear as possible, but if something is confusing or incorrect, please tell me about it and I'll fix it. If you have any questions, you can subscribe to the livid−user mailing list by sending a blank email to livid−user−subscribe@linuxvideo.org.

## 1.2 New Versions

This howto is available in many formats, including html and tex. The newest version can always be retrieved from http://www.linuxvideo.org/docs/Linux−DVD−HOWTO/. It is highly recommended that you get the most recent version, as this howto is being updated quite often.

## 1.3 Copyrights and Trademarks

This manual may be reproduced in whole or in part, without fee, subject to the following restrictions:

- The copyright notice above and this permission notice must be preserved complete on all complete or partial copies.
- Any translation or derived work must be approved by the author in writing before distribution.
- If you distribute this work in part, instructions for obtaining the complete version of this manual must be included, and a means for obtaining a complete version provided.
- Small portions may be reproduced as illustrations for reviews or quotes in other works without this permission notice if proper citation is given.

Exceptions to these rules may be granted for acedemic purposes: Write to the author and ask. These restrictions are here to protect us as authors, not to restrict you as learners and educators. Any souce code (aside from the SGML this document was written in) in this document is placed under the GNU General Public License, available via anonymous FTP from the GNU archive.

## 1.4 Thanks

Thanks to Nathan Rowlan nkr@helo.org for starting and maintaining this document in the early days. It has now been taken over by the LiViD support team.

Thanks to the authors of OMS and the rest of the LiViD developers, who have given the project much of their time and work, and also to those individuals who have given feedback to make this HOWTO better.

Much of the layout of this HOWTO was gotten from the HOWTO−HOWTO, written by Mark F. Komarinski. It can be found at http://www.linuxdoc.org/.

## 1.5 Translations

Translations of this document can be found at http://www.linuxvideo.org/docs/Linux−DVD−HOWTO/. If you would like to translate this howto, please contact the LiViD development mailing list.

## 2. Requirements

This HOWTO assumes you have:

- Linux kernel v2.2.x or greater. A kernel with DVD ioctls is prefered.
- XFree86 3.3.X or 4.x
- A DVD−ROM/RAM/RW drive supported in Linux (most are)
- A decent knowledge of bash and english

## 3. Files

First things first, you need to get a kernel with MTRR (more on this later) and DVD ioctl support. While MTRR's have been in the kernel since 2.2.11 (correct me if I'm wrong), you will probably need to update your kernel to get DVD ioctls. There are two ways to skin this cat, you can either download a development kernel (v2.3.x), or patch a stable release kernel (v2.2.x), the patches are at http://www.kernel.dk/, and the kernel can be found on your favorite mirror of ftp://ftp.kernel.org/.

Next, you need to get the LiViD utilities. This can be done two ways:

- Get the LiViD utilities from CVS, so you can update it as it is being updated by the developers. (recommended)

```
The following commands will retrieve them (you can use −d parameter
instead of setting CVSROOT if you want):

# mkdir ~/livid
# cd ~/livid
# export CVSROOT=:pserver:anonymous@cvs.linuxvideo.org:/cvs/livid
# cvs login
(Logging in to anonymous@cvs.linuxvideo.org)
CVS password:

There is no password for anonymous, just press enter.

# cvs −z3 co −P ac3dec mpeg2dec oms

They should download into their respective directories.
```

- If you don't like CVS or can't access it due to firewall issues, you can download the most recent nightly tarball of LiViD tools at http://www.linuxvideo.org/developer/dl.phtml

# 4. Installation

## 4.1 Kernel

Kernels at least as recent as 2.2.16 or 2.4 series should already have DVD ioctl support, so you just need to make sure you have MTRR support enabled in the kernel configuration, and then compile and install it as you normally would.

If you want to patch your exisiting older kernel, start by going to http://www.kernel.dk/ and downloading the correct patch for your kernel.

If don't know how to install or patch your kernel, you should go read the Kernel HOWTO at http://howto.tucows.com/LDP/HOWTO/Kernel−HOWTO.html

## 4.2 LiViD Utilities

To install the LiViD utilities, you need to make sure you have /usr/local/lib somewhere in /etc/ld.so.conf.

DVD playback only requires OMS to be built. It builds ac3 and mpeg2 codecs by itself. However, feel free to play with the standalone codecs if you wish. The following commands should compile and install the LiViD utilities on your system. If you have problems, please see the "Problems" section.

```
ac3dec: (optional)

  # cd ~/livid/ac3dec
  # ./autogen.sh
  # make
  # make install

mpeg2dec: (optional)

  # cd ~/livid/mpeg2dec
  # ./autogen.sh
  # make
  # make install

oms:

  # cd ~/livid/oms
  # ./autogen.sh
  # ./configure
  # make
  # make install
```

There are some extra options that can be used. They are placed as arguments to the ./configure script. OMS has −−enable−devel which will enable some development and experimental features in the code. Standard autoconf parameters such as −−prefix can be used if you wish to install somehwere other than /usr/local. For instance to install in a subdir of the source dir you could do:

```
  # ./configure --prefix=`pwd`/inst
```

Run ./configure −−help to get a complete listing of options. If you install X in an odd location (for instance you have XFree86 3.3.x and 4.x installed at the same time) you may need the −−x−includes and −−x−libraries options.

Now the necessary LiViD utilities should be installed. The next section is not required, but if you use the "pipes" plugin, you will need to make some special fifo pipes for the DVD data to travel over. These can be made with the following commands (if they don't already exist):

```
# mkfifo /tmp/video
# mkfifo /tmp/audio
```

Another method of controlling the data available in the oms_devel version is to use raw I/O, which is available in the latest linux kernels. If you don't have these, data will be read from the standard device. Using raw I/O is recommended, but not necessary. If they do not exist, create two devices as follows:

```
# mknod /dev/rawctl c 162 0
# mknod /dev/raw1 c 162 1
```

One last thing to do. OMS needs /dev/dvd to be a symlink to where your DVD drive resides, such as /dev/hdb1 or /dev/scd0. If it is at /dev/cdrom, you would create the link by typing:

```
# ln -s /dev/cdrom /dev/dvd
```

It is generally a good idea not to make a symlink to a symlink, because it adds unnecessary I/O. Replace /dev/cdrom with the correct device that your dvd drive is on.

# 5. Playing

First you need a config file. Copy doc/config.sample from oms source directory to .oms/config in your home directory. Edit the values to reflect the drivers, skin, device, etc you want to use.

Expect lots of text to scroll by. This is still development code. If you see lots of text, but no video window you may have a mistake in your config file. Check to make sure you are using the right plugins and devices.

To play a DVD, you need to be running X in 16−bit color mode. With the DVD in the drive, type:

```
# oms
```

There is a pipe method available in older OMS versions that uses the fifo's created earlier. Most people can ignore this. To use pipes type:

```
# ac3dec /tmp/audio& mpeg2dec /tmp/video& oms
```

ac3dec and mpeg2dec may need to be killed by hand using this method.

If all has gone well, some text should fly by, followed by it asking you if you want it to look up the name of the disk in the DVDDB. It is safe to say yes. Then, the oms panel should pop up. Click the playlist button, then the scan dvd button, then play.

# 6. Additional Stuff

## 6.1 Setting up MTRR's

```
** Note to Matrox video card owners **

If you have matroxfb compiled into the kernel,
chances are that your MTRR's are already set
up.  You can probably just skip this section.
```

Setting up your MTRR's can increase video performance quite a bit in some cases, so it is a good idea to do it. First, you need to make sure you have MTRR support in your kernel by typing:

```
# ls /proc/mtrr
```

If it tells you that there is no such thing as /proc/mtrr, you need to recompile your kernel with MTRR support (it is under "Processor type and features" in menuconfig).

Once you know MTRR is working, you need to know the base memory address of your video card, and how much video ram it has. The easiest way to do this is to look at the output of X as it starts up. Because the output usually scrolls off the screen and is lost when it switches to a different tty, you need to redirect the output of X to a file (xoutput) so you can go back and look at it to get the needed values. This can be done by typing:

```
# startx 2> xoutput
```

The line having the needed information is probably somewhere towards the middle of xoutput, and should look something like:

```
(--) SVGA: PCI: NVidia Riva TNT2 rev 17, Memory @ 0xee000000, 0xe2000000
```

Once you have located that, write down the last memory address, in this case 0xe2000000. Depending on your hardware, you may or may not have multiple memory addresses shown, so don't worry if yours looks a little different. With this information recorded, you can delete xoutput.

Next you need to create a new MTRR. In order to do that, you have know how much ram your video card has in hex. Here are some common values:

```
4MB  -- 0x400000
8MB  -- 0x800000
16MB -- 0x1000000
32MB -- 0x2000000
```

To add the MTRR, type:

```
# echo "base=0xe2000000 size=0x2000000 type=write-combining" >| /proc/mtrr
```

Substituting "0xe2000000" and "0x2000000" with the base address and amount of video ram specific to your system.

Now you should have MTRR set up, and just to make sure, type:

```
# cat /proc/mtrr
```

And you should get output that looks remotely similar to:

```
reg00: base=0x00000000 (   0MB), size= 128MB: write-back, count=1
reg01: base=0xe2000000 (3616MB), size=  32MB: write-combining, count=1
```

Again, the number of entries will probably be different from these, don't sweat it.

---

# 7. Problems

## 7.1 Run time errors

This is a (small) list of known run–time errors.

### Illegal Instruction Error

If you use a non–Intel chip (K6 especially), and you are getting this error when you try to run mpeg2video, try editing nist/configure.in, lines 129 and 130:

```
CFLAGS="$CFLAGS -DHAVE_MMX -DLINUX -march=i686 -fschedule-insns2 -malign-doub
CXXFLAGS="$CXXFLAGS -DHAVE_MMX -DLINUX -march=i686 -fschedule-insns2 -malign-
```

replace "–march=i686" in both with "–march=i586", and then recompile and reinstall.

## 7.2 Compilation errors

This is a running list of common compilation errors that have known fixes.

### `dvd_struct' undeclared...

The most frequent problem people have when trying to compile these utilities stems from oms looking in the wrong place for the kernel headers. By default, it uses /usr/include/[linux|asm], but those headers are from a stable kernel so that when you compile a normal program, it will be using older headers. OMS needs to use headers with dvd ioctl support. If things are set up wrong many things are undeclared and compilation fails. The best way to fix this is to use the configure option ––with–kernel–headers=(path to headers). Another way is to adjust which headers are in the default location.

```
These commands should do it:

# mkdir /usr/include/old
# mv /usr/include/linux /usr/include/old/linux
# mv /usr/include/asm /usr/include/old/asm
# mv /usr/include/scsi /usr/include/old/scsi
# ln -s /usr/src/linux/include/linux /usr/include/linux
```

```
# ln -s /usr/src/linux/include/scsi /usr/include/scsi
# ln -s /usr/src/linux/include/asm /usr/include/asm
```

### Can't determine absolute dir of '../../../../src/plugin/codec/mpeg2dec/.libs'

Another common error is with the oms/src/plugin/codec/mpeg2dec/.libs directory. For some reason or another, this directory does not exist, and it needs to for compilation. To fix it, just make the directory:

```
# mkdir src/plugin/codec/mpeg2dec/.libs
```

### Can't find libXv.so or libXxf86dga.so

Xfree86 4.x doesn't build shared libs for Xv and Xxf86dga. However, the shared lib setup OMS uses needs them. Build them as follows in your favorite lib dir:

```
# ld --whole-archive -shared -o libXv.so libXv.a
# ld --whole-archive -shared -o libXxf86dga.so libXxf86dga.a
```

### Failure via segfault for no apparent reason

Sometime you have old libraries laying around that are used by mistake. This will cause unwanted behavior such as crashing. Get rid of old libraries from /usr/local/lib or wherever you put them and things should work.

## 7.3 Broken stuff

Features that don't currently work as intended:

- The chapter seek buttons may fail.
- The play/pause/stop buttons may fail to work.
- Audio/video may skip or sound "choppy". This is due to proper syncronization and framedropping not being implemented yet.

## 7.4 Other errors

If your problem isn't listed here, then likely the current CVS code has a bug in it or is missing a feature. You can either hope it will be fixed and update CVS every so often or subscribe to the mailing lists and report the bug or request a feature. To subscribe send a blank email to livid−user−subscribe@linuxvideo.org.

---

# 8. To Do

- Makefile options (HAVE_MMX, HAVE_3DNOW, etc)
- proper architecture detection (compile time and runtime)
- add easy MTRR stuff (framebuffer)
- add need for region coding to be set on most drives: www.linuxtv.org/dvd

---