

mplayer & mencoder
NLLGG, 02.02.2008
Author: Marek Franke
E-Mail: mfranke@dulug.de

Typographic Conventions

Bold

is used for commands and their outputs to the commandline. A command that begins with a **\$** is a command that a normal user should do and a command that begins with a **#** is a command that root should do.

A user command:

\$ ls

A root command:

make install

Introduction

mplayer is a very powerful commandline based movieplayer with a lot of features and codecs. See that it has a lot of features the usage of mplayer is not that easy to understand. The actual release is mplayer-1.0rc2 and this document based on this release.

Building the source

The source of mplayer can be downloaded at <http://www.mplayerhq.hu> and there are several packages for most Linux distribution available, e.g. at <http://www.rpmseek.com>, <http://www.linuxpackages.net> etc. There are packages for other Unices (*BSD, OpenSolaris etc) available too. If you're out of luck and can't find a suitable package for your distribution, you need to compile mplayer by yourself. Depending on your distribution, you should first install several development packages. After that download a source package for mplayer and extract it:

\$ tar xjf MPlayer-1.0rc2.tar.bz2

Now change into the new created directory:

\$ cd Mplayer-1.0rc2

After that you should take a look at the compiling options:

\$./configure --help | less

...

Audio output:

--disable-alsa **disable ALSA audio output [autodetect]**

--disable-ossaudio **disable OSS audio output [autodetect]**

--disable-arts **disable aRts audio output [autodetect]**

...

As you can see above the most compiling options are autodetected. So the simplest example to finally configure mplayer is this:

\$./configure --codecsdir=/usr/local/lib/codecs

The codecsdir parameter will tell mplayer where to look for the codecs and in this

case it will be /usr/local/lib/codecs. If the configure skript didn't fail, you can build mplayer:

```
$ make
```

Depending on your hardware this could take a long time. When make compiled everything, you can install mplayer (keep in mind to do that as root):

```
# make install
```

Now you should look at /usr/local/bin:

```
$ ls /usr/local/bin
```

```
...
```

```
mencoder
```

```
mplayer
```

```
...
```

There are two new binaries, the installation of mplayer and mencoder is finished.

Installing the codecs

Installing the codecs is as easy as can be. At first you should download the essential codecs package from <http://www.mplayerhq.hu> (something like essential-20071007.tar.bz2). If you build mplayer by yourself then you should copy the extracted package to /usr/local/lib/codecs/ (or whatever you take as path for --codecsdir):

```
# tar xjf essential-20071007.tar.bz2
```

```
# mv essential-20071007/* /usr/local/lib/codecs/
```

```
# ls /usr/local/lib/codecs/
```

```
...
```

```
QuickTime.qts
```

```
QuickTimeEssentials.qtx
```

```
...
```

In most cases you should copy the codecs as root. If you use a precompiled package and need to install the codecs, then you should use the /lib/ dir near to your /bin/ dir. If you can find mplayer at /usr/bin/, you should copy the codecs to /usr/lib/codecs/. If you can find mplayer at /opt/anything/bin/, you should copy the codecs to /opt/anything/lib/codecs/.

mplayer

Now it's time to get started. The easiest way to play a file is this one:

```
$ mplayer file.avi
```

To play the movie in fullscreen press 'F' while playing or start mplayer with this command:

```
$ mplayer file.avi -fs
```

Video output

mplayer comes with a lot of output plugins and next we will take a look at the video output plugins:

```
$ mplayer -vo help
```

```
...
```

sdl SDL YUV/RGB/BGR renderer (SDL v1.1.7+ only!)
fbdev Framebuffer Device

...

You get a list with several plugins and you should try playing around a little with them. For example the fbdev plugin could be used to play movies in framebuffer without X. Or If your are an ascii art lover, try the aalib output:

\$ mplayer file.avi -vo aa

Audio output

You can do the same for the audio output plugins as for the video output plugins:

\$ mplayer -ao help

...

alsa ALSA-0.9.x-1.x audio output

arts aRts audio output

...

If you run KDE and can't hear a sound when playing a file with mplayer, you should try to tell mplayer to use the arts audio plugin (in most cases KDE starts up arts when being started):

\$ mplayer file.avi -ao arts

It is possible to combine audio output and video output plugins:

\$ mplayer file.avi -ao arts -vo aa

Identifying a file

Sometimes it is necessary to know what kind file you have and mplayer has a suitable option for this:

\$ mplayer file.avi -identify

...

ID_VIDEO_WIDTH=320

ID_VIDEO_HEIGHT=240

ID_LENGTH=34.00

...

To get only the desired informations without playing the movie, try this command:

\$ mplayer file.avi -identify -frames 0 -ao null

The -frame option will only play the the number of frames you specify (in this case 0) and the -ao option will play no sound.

Playing a DVD

Before playing a DVD you need to know a little about the structure of a DVD. Each DVD comes with several titles and each title can have several chapters. As a simple example try this:

\$ mplayer dvd://1

This will play the complete first title of the DVD. To play the 1. title from the 2. chapter use this:

\$ mplayer dvd://1 -chapter 2

This will play the DVD from the 2. chapter till the end of the DVD. To play only the

2 chapter use this:

```
$ mplayer dvd://1 -chapter 2-2
```

And it is possible to change the language and use subtitles:

```
$ mplayer dvd://1 -chapter 2-2 -alang en -slang en
```

This will play the DVD with audio language (-alang) english and show the subtitle language english (-slang). If you have a music DVD with a concert from your favorite band and want to listen to the music without showing the concert (or else) then try this:

```
$ mplayer dvd://1 -chapter 2-2 -vo null
```

If you want to copy the audio output directly to harddisk then use the pcm audio output:

```
$ mplayer dvd://1 -chapter 2-2 -ao pcm:waveheader:file=02.wav
```

Or without watching the movie (even though it's funny):

```
$ mplayer dvd://1 -chapter 2-2 -ao pcm:waveheader:file=02.wav -vo null
```

After that you can play the new file again with mplayer:

```
$ mplayer 02.wav
```

Playing DVB-T

Before you can play a DVB-T signal you need a channel list (how to use your DVB-T hardware and how to create a channel list is far beyond this document but maybe you want to take a look at <http://www.linuxtv.org/>). A channel list should be stored at /home/user/.mplayer/channels.conf (where 'user' is your real username) and could look like this:

```
$ cat ~/.mplayer/channels.conf
```

```
...
```

```
WDR Duisburg:
```

```
67400000:INVERSION_AUTO:BANDWIDTH_8_MHZ:FEC_2_3:FEC_2_3:QAM_16:TRANSMISSION_MODE_8K:GUARD_INTERVAL_1_4:HIERARCHY_NONE:4881:4882:267
```

```
...
```

With the channel list you can now easily play a channel:

```
$ mplayer dvb://"WDR Duisburg"
```

Playing analogue composite TV signal

To play a signal that comes from the composite input of your TV card you need to specify the input device (in this case 1, but it depends on your TV card):

```
$ mplayer tv:// -tv driver=v4l2:input=1
```

That's a little confusing because of the two tv parameters. The first tv parameter (tv://) tells mplayer which kind device we want to use, just like dvd://, dvb:// etc. The second tv parameter (-tv) introduce the tv options, just like selecting a chapter from a DVD (-chapter 2-2).

When you start playing a TV signal with mplayer, you should get an output like this:

```
$ mplayer tv:// -tv driver=v4l2:input=1
```

```
...
```

```
inputs: 0 = Television; 1 = Composite1; 2 = Composite2; 3 = S-Video;
```

...

With this Information you can easily determine your input.

Playing analogue cable TV signal

To play a analogue cable TV signal we need a few more informations like TV norm, channellist and a channel:

```
$ mplayer tv:// -tv driver=v4l2:input=0:norm=pal:\  
chanlist=europe-west:channel=E1
```

First you see that we have changed our input from Composite1 (input=1) to Television (input=0). Next we specify our TV norm, in this case PAL (for Netherland and Germany). At last we need a channellist (europe-west) where we can choose a channel from (E1). You can zap down with the key 'h' and zap up with the key 'k'.

mencoder

Now it's time to convert a little with mencoder. The syntax is very easy and the basics are:

```
$ mencoder <audio options> <video options> <input file> <output file>
```

The audio options will be introduced with something like '-oac ...' and the video option with something like '-ovc ...'. The input file could be anything from above (a file, DVB-T signal, DVD...) and the output file must be a any creatable file. For example we want to convert the 2. chapter from the 1. title from a DVD:

```
$ mplayer -oac ... -ovc ... dvd://1 -chapter 2-2 -o chapter2.avi
```

As you can see the input file has the same syntax like the mplayer command:

```
$ mplayer dvd://1 -chapter 2-2
```

Now we only need to get rid of these dots for the output audio codec (-oac) and the output video codec (-ovc). And the easiest way to this is just to copy the codec and finally here is our first complete mencoder command:

```
$ mencoder -oac copy -ovc copy dvd:// 1 -chapter 2-2 -o chapter2.avi
```

Using lavc

To create a mpeg1, mpeg2 etc file you need to use the libavcodec that comes with the package ffmpeg.

The first thing we do is to change the copy parameter from above to lavc:

```
... -oac lavc -ovc lavc ...
```

Next we specify the output format:

```
... -oac lavc -ovc lavc -of mpeg ...
```

Select a video codec from the libavcodecs:

```
... -oac lavc -ovc lavc -of mpeg -lavcopts vcodec=mpeg2video ...
```

And a audio codec from the libavcodecs:

```
... -oac lavc -ovc-lavc -of mpeg -lavcopts vcodec=mpeg2video:acodec=mp2 ...
```

Now that we have selected a audio and video codec we can set a bitrate for audio and video:

```
... -lavcopts vcodec=mpeg2video:vbitrate=1152:acodec=mp2:abitrace=224 ...
```

The last audio option we set is the samplerate:

... -srate 44100 ...

And the last video option we set is the `harddup` video filter (this will force duplicate frames, without the `harddup` video filter the movie could look very fast but funny):

... -vf `harddup` ...

And finally the complete `mencoder` command looks like this:

```
$ mencoder -oac lavc -ovc lavc -of mpeg -lavcopts \  
vcodec=mpeg2video:vbitrate=1152:acodec=mp2:abitrage=224 -srate \  
44100 -vf harddup input.avi -o output.mpg
```

Setting output framerate

Sometimes it's necessary to set the output framerate and there's a single option to do it:

```
$ mencoder -oac lavc -ovc lavc -of mpeg -lavcopts \  
vcodec=mpeg2video:vbitrate=1152:acodec=mp2:abitrage=224 -srate \  
44100 -vf harddup -ofps 25 input.avi -o output.mpg
```

Resizing

To resize a video we need to append the video filter with this:

```
$ mencoder -oac lavc -ovc lavc -of mpeg -lavcopts \  
vcodec=mpeg2video:vbitrate=1152:acodec=mp2:abitrage=224 -srate \  
44100 -vf scale=320:240,harddup -ofps 25 input.avi -o output.mpg
```

Appendix A: Creating 3gp files in 3 steps

Step 1: Create an audiodump (where 'audiodump.wav' will be your audiodump and 'file.avi' will be your input file)

```
$ mplayer -vo null -ao pcm:file=audiodump.wav -af \  
resample=8000,volume=+4db:sc file.avi
```

Step 2: Create a videodump (where 'videodump.avi' will be your videodump and 'file.avi' will be your input file again)

```
$ mencoder file.avi -nosound -ovc lavc -lavcopts vcodec=mpeg4 -vop \  
expand=176:144,scale=176:-2 -o videodump.avi -ofps 12
```

Step 3: Merge audiodump and videodump to one 3gp file

```
$ ffmpeg -i videodump.avi -i audiodump.wav -b 48 -ac 1 -ab 12 -map 0.0 - \  
map 1.0 file.3gp
```

Appendix B: Using subtitle files (*.srt)

When you have a movie file (for example called `movie.avi`) and a subtitle file (called `subtitle.srt`) then you can play them both with `mplayer` this way:

\$ mplayer -sub subtitle.srt movie.avi

This will load the subtitle.srt file as subtitle and display them. If you can't see any subtitles again then look out for a message like this during startup:

\$ mplayer -sub subtitle.srt movie.avi

...

Please supply the text font file (~/.mplayer/subfont.ttf).

...

This means that the subtitle font is missing. So you need to copy any (of course good readable like Arial or Sans) ttf to /home/user/.mplayer/subfont.ttf, e.g.:

\$ cp /usr/share/fonts/TTF/DejaVuSans.ttf /home/user/.mplayer/subfont.ttf

Where user is your own username.

Appendix C: Manpages and links

To get more informations about mplayer and mencoder read the man page:

\$ man mplayer

or

\$ man mencoder

They are both the same.

Mplayer homepage: <http://www.mplayerhq.hu>

Linux TV (DVB, V4L,...): <http://www.linuxtv.org>