Converting Standard Audio Formats
CD2MP3

HANS-GEORG ESSER

You may already have the hang of converting audio files to wav files and downloading mp3 files from the Net. But how is one format converted to the other?

If you'd like to listen to *mp3* files in your car, but you don't have a new-fangled in-car mp3 player, you've got a problem because your CD with the 180 best tracks from the Net cannot be played on a normal CD changer. The files have to be converted into the memory-guzzling *wav* format and then reprocessed into an ordinary audio CD. Conversion in the other direction requires a 50GB hard drive, and the advised format for conversion is MPEG 1 Layer 3, rather than *wav* files. However, if you're willing to work on the command line, you'll find that Linux provides fast and practical tools for this task.

mp3 to wav

One of the most popular console MP3 players is *mpg123* (*http://www.mpg123.de/*, latest version: 0.59r). Anyone taking a look at its options will also immediately find an easy way to convert from *mp3* to *wav*:

```
mpg123 -w song.wav song.mp3
```

does the necessary (see Figure 1). This is unsurprising since every mp3 player converts files into *wav* format, in order to play them.

On Freshmeat's *mpg123* project homepage (*http://freshmeat.net/projects/mpg123/*) there is script which burns a series of mp3 files directly onto an audio CD:

```
for var in `ls -1 $1`; do
 echo Burning $var ..........;
 mpg123 -s $var | cdrecord dev=imation -v -➥
nofix -audio -swab -
done
cdrecord dev=imation -fix
```

The script uses the option *-s* ("write to stdout"), and the output from *mpg123* is piped directly into the *cdrecord* process.

Those who prefer to take it easy can also use *xmms* (*http://www.xmms.org/*): To write *wav* files, open the settings via the menu item *Options/Preferences* (or [Ctrl-P]) and change the output plugin to *Disk Writer Plugin*. Then click on *Configure* and specify a destination directory for the *wav* files (see Figure 2).

You can now create an mp3 playlist for *xmms*. When your playlist is compiled, you can simply click with the mouse to start playback. Although you might not hear anything at first, *xmms* is working through your playlist systematically, placing the *wav* files in the correct directories.

wav to mp3

mpg123 is usually employed as an *mp3* decoder, so, going from wav to mp3 requires an encoder. One good candidate is *bladeenc* (*http://bladeenc.mp3.no*); an alternative is LAME (*http://www.mp3dev.org/mp3/*). The syntax of *bladeenc* is, not surprisingly:

```
bladeenc in.wav out.mp3
```

Since coding takes up considerably more resources than decoding, *bladeenc* cannot code the data in real time. This means that the procedure takers longer than it would to play the piece at normal speed. *bladeenc* creates *mp3* files at 128KB/sec by default. If you want better quality, you should use the *-br* (bit rate) option. The bit rates available are 32, 40, 48, 56, 64, 80, 96, 112, 128, 160, 192, 224, 256 and 320 KB/sec. So, for the 192 KB/sec frequently found on Napster servers, use the command:

```
bladeenc -br 192 in.wav out.mp3
```

Automation

One advantage of working on the command line is that you can automate procedures, making it is easy to build *mpg123* and *bladeenc* into shell scripts. These scripts can then be used to convert whole directories from one format into the other overnight. Whether by script or by hand, you can now have lots of fun converting and enjoy many hours of listening pleasure — on the computer or in the car. ■

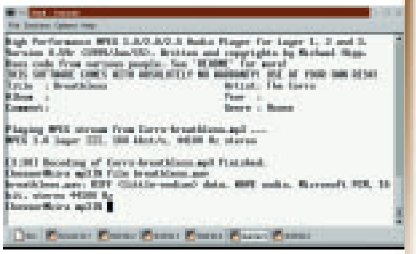

Figure 1: mpg123 effortlessly converts from mp3 to wav

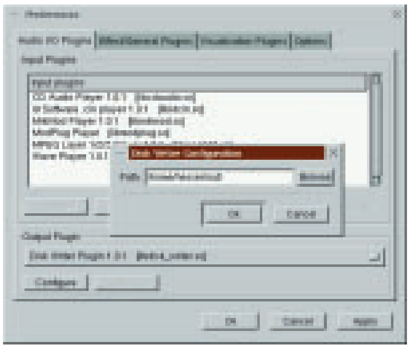

Figure 2: Anyone who doesn't want to work on the console can fall back on xmms

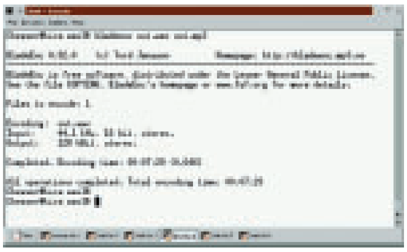

Figure 3: bladeenc is an mp3 encoder