Music tools for the command line

ROCK AROUND THE SHELL

You don't always need a GUI-based tool to play MP3 and Ogg Vorbis tracks. In this month's column, we look at tools for rocking from the shell. **BY HEIKE JURZIK**

www.sxc.hu

If you have a fast machine, players such as amaroK or JuK are not only a treat for your ears – they also give you lots of eye candy. But even if you have an older machine, you don't need to do without music while you work. Thanks to small footprint command line players for MP3 and Ogg Vorbis, a text-based console is all you need.

The other advantage these programs offer is that they integrate nicely with other shell commands. You can search your disk for sound files and automatically feed the results into a playlist. And if you are converting to WAV format – to put a collection of tracks on an audio CD, for example – the command line is definitely preferable to any GUI.

Light Music with mpg123 and mpg321

mpg123 [1] is a command line tool that goes back quite a long way, having served for years on both Linux and Unix platforms. As mpg123 was not released under the GPL, Joe Drew developed a completely free alternative – the mpg321

[2] player. There is no difference between the major features of these tools, so the options we will be looking at apply equally to both.

The simplest way of calling the program is to type *mpg123 file.mp3* or *mpg321 file.mp3*. While happily outputting sound through your speakers, the program writes additional information on the console (Listing 1). Besides program version information, the output includes information such as the title, artist, and album (assuming the **ID3** tags are set correctly), as well as the MP3 file quality. If this is too much information for your liking, you can prevent this output by setting the *-q* flag.

To interrupt the player, press [Ctrl] + [Z] to send the process to the background. Typing *fg* moves the process back to the foreground and continues where you left off. To quit the command line player, press [Ctrl] + [C].

Talking about Ogg

Ogg is an alternative to the MP3 format. As a completely open, non-patented for-

mat for streaming applications, Ogg provides a container for codecs such as Vorbis and FLAC. The Ogg Vorbis codec for lossy audio compression [3] also offers superior quality to MP3. The *vorbis-tools* program package, which most distributions now include by default, has a useful selection of tools:

- ogg123: A command line player, which works in a similar way to mpg123 or mpg321.
- oggdec: A simple decoder that converts Ogg Vorbis files to WAV.
- oggenc: Converts WAV and AIFF files to Ogg Vorbis streams.
- ogginfo: Displays information about Ogg files at the command line.
- vcut: Cuts Ogg Vorbis files.
- vorbiscomment: Supports editing of Vorbiscomment tags.

The ogg123 player is all you need to play an Ogg Vorbis file. Listing 2 shows sample output. Again, the normal process control commands apply: [Ctrl] + [Z] allows you to pause the output, *fg* continues playing, and [Ctrl] + [C] quits the player. Just like mpg123 and mpg321,

you can suppress the informational output by specifying the *-q* parameter.

At One Fell Swoop

To play multiple files back to back with mpg123, mpg321 or ogg123, simply specify the filenames when calling the command:

```
mpg321 ⤶
05_Lily_Of_The_Valley.mp3 ⤶
08_Stone_Cold_Crazy.mp3 ⤶
09_Dear_Friends.mp3
```

or

```
ogg123 ⤶
01_Barenaked_Ladies_-⤶
_Celebrity.ogg 02_⤶
Barenaked_Ladies_-⤶
_Maybe_Katie.ogg ...
```

Of course you can use typical command line shortcuts and specify things like *mpg321 *.mp3* to play all MP3 files (or *ogg123 *.ogg* for all Ogg Vorbis files) below the current directory. Pressing [Ctrl] + [C] interrupts the current track and allows you to skip to the next track. To quit the program in this case, you need to press [Ctrl] + [C] twice in quick succession.

If the audio files on your system are not neatly organized below your home directory but are instead spread across subdirectories all over your disk, you

might like to call *find* to locate those audio files:

```
mpg321 $(find ~ -name "*.mp3")
```

This *find* command starts in your home directory (indicated by the tilde), and searches all sub-directories for files with the *.mp3* extension. The dollar sign and the parentheses tell bash to pass the output from this command as an argument to mpg321.

Things start to get more tricky if your files include blanks or non-standard characters; the typical result is a storm of protest in the form of error messages. But a simple shell trick will help set things right:

```
find ~ -name "*.ogg" ⤶
| ogg123 -@ - -z
```

The *find* command runs first; any files it finds are piped to the *ogg123* call, which reads from standard input (-) and edits the list of files as a playlist (as specified by the -@ option.) The final parameter, *-z*, makes sure that ogg123 does not sim-

Listing 1: mpg123 Output

```
01 $ mpg321 01_Keep_
     Yourself_Alive.mp3
02 High Performance MPEG 1.0/2.0/
     2.5 Audio Player for Layer 1,
     2, and 3.
03 Version 0.59q (2002/03/
     23). Written and copyrights by
     Joe Drew.
04 Uses code from various people.
     See 'README' for more!
05 THIS SOFTWARE COMES WITH ABSOL
     UTELY NO WARRANTY! USE AT YOUR
     OWN RISK!
06 Title  : Keep Yourself Alive
            Artist: Queen
07 Album  : Queen
            Year  : 1973
08 Comment:
            Genre : Rock
09
10 Playing MPEG stream from 01_
     Keep_Yourself_Alive.mp3 ...
11 MPEG 1.0 layer III, 128 kbit/
     s, 44100 Hz joint-stereo
```

ply play the tracks in sequence but uses a random play (shuffle) mode instead. As mpg123 and mpg321 understand the same parameters, this trick also works for MP3 files.

Converted

To store your MP3 collection on a normal audio CD, which you can then play on your stereo, for example, you first need to convert the compressed sound files to WAV format. mpg123 and mpg321 give you the *-w* option for this. Note that you need to specify the output file before the input file:

```
$ mpg321 -w file.wav file.mp3
...
[3:47] Decoding of ⤶
file.mp3 finished.
```

As you can only process a single file in this way, it is a good idea to call mpg321 within a *for* loop, thus automating the conversion procedure for multiple files. To convert all the MP3s in a directory to WAV, you would type:

```
for i in *.mp3; do mpg321 -w ⤶
`basename $i .mp3`.wav $i; done
```

This call assigns every file with the *.mp3* extension to the *i* variable, and then runs *mpg321 -w* against the variable. The first parameter is again the output file (made up of the base name without the *.mp3* extension, and with the *.wav* extension instead); the second parameter is the output file *$i*.

The command for converting Ogg Vorbis files to WAV format is not as long-winded: the *oggdec* program automatically replaces file extensions, and simply expects the file to be decoded:

```
$ oggdec sunrise.ogg
OggDec 1.0
Decoding "sunrise.ogg" ⤶
to "sunrise.wav"
     [100.0%]
```

To convert all files at a single pass, simply type *oggdec *.ogg*.

Terminal Ghetto Blaster

You may have noticed that the command line tools we have looked at thus far lack interactive controls. If you need an application that gives you these controls

without the overhead of a GUI, mp3blaster [4] may be just what you are looking for. As the name suggests, the player can handle MP3s, WAV, and Ogg Vorbis files; the player is included with most distributions. mp3blaster does not require a GUI environment; thanks to **Ncurses**, it is quite happy to run in a terminal window and gives you push button controls.

To launch the player, type mp3blaster at the command line. The available keyboard shortcuts are displayed at the top right of the window; [+] and [-] browse to the right or left in the list to allow you to view the other commands. mp3blaster gives you really informative output when playing audio files, including technical details about the file itself, the play sequence, and the next song on the playlist. You can type a question mark (?) to display or hide the integrated help (Figure 1).

Pressing [F1] takes you to file manager mode, where you can select files to play. This also changes the keyboard shortcuts at the top of the window. The middle of the window shows the files and directories in the folder where you launched the player, and you can use the arrow keys or [Pg Up] and [Pg Dn] to navigate. To change to a directory, simply press the enter key; selecting the ../ entry takes you up one level.
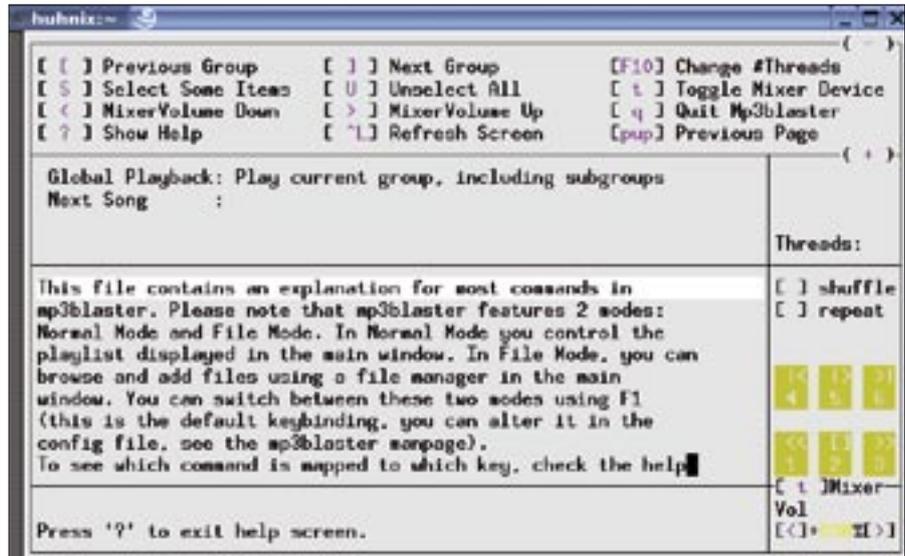

Figure 1: The integrated help feature describes the major functions.

To play a file, use the arrow keys to navigate to that file and then press Enter. If you would like to select multiple tracks and place them in a playlist instead, press the space key to select a track (the track entry is highlighted), and then press [F1] to add your selections to the playlist. You can select a whole directory quickly by inverting the selection ([F2]); this option automatically selects all the tracks in that directory, allowing you to press [F1] to add them to your playlist.

Press the [5] key to start playing. mp3blaster displays the current track and the next entry on the list (Figure 2). The other keyboard shortcuts are [5] (pause), [4] (previous track), [6] (next track), [2] (stop), [1] (rewind) and [3] (fast forward). The [F6] key enables the repeat function; [F7] selects a random play mode (shuffle). The integrated mixer is hidden in the bottom right-hand corner of the window; pressing [T] toggles between the various devices. You can press [<] (quieter) and [Shift]+[<] (louder) to set the volume – a percentage display gives you more information on the current volume. And finally, pressing [Q] quits the program.

Sorting Magic

If you press [F1] to change to selection mode, the various audio formats are displayed in green by default. And mp3blaster displays playlists, which typically have extensions such as *.m3u* or *.lst*, in yellow. If you select a playlist file by pressing the Enter key, mp3blaster

automatically adds all the songs on the list to your current playlist.

It is just as easy to set up a new collection. To store the current selection of files as a playlist, first press [F4], then type a name for the playlist (adding the file extension *.lst* or *.m3u*) and press [Enter] to confirm. If you need to tidy up or sort the list before you start, you can press [D] to delete a song from the list. Pressing [M] or [Shift]+[M] changes the order of the tracks on the list. And you can completely clear a playlist by pressing [C].

Off to a Good Start

mp3blaster has a few command line options that allow you to control the way the player behaves when it launches. To automatically load a playlist on launching the player, and to start playing back immediately, specify the *-a*

Listing 2: ogg123 Output

```
01 $ ogg123 01_Barenaked_Ladies_
   -_Celebrity.ogg
02 Audio Device:   Advanced Linux
    Sound Architecture (ALSA) out
   put
03
04 Playing: 01_Barenaked_Ladies_
   -_Celebrity.ogg
05 Ogg Vorbis stream: 2 channel,
   44100 Hz
06 Title: Celebrity
07 Artist: Barenaked Ladies
08 Genre: Rock
09 Date: 2003
10 Album: Everything to Everyone
11 Track number: 01
12 Time: 02:00,12 [01:27,62] of 0
   3:27,73  (110,7 kbps)  Output
   Buffer  96,9%
```

GLOSSARY

Ncurses: A free C program library, which includes keyboard and mouse input handling, along with multiple windowing

INFO

[1] mpg123 homepage:
   http://www.mpg123.de/
[2] mpg321 website:
   http://mpg321.sourceforge.net/
[3] Ogg Vorbis website:
   http://www.vorbis.com/
[4] mp3blaster console player:
   wwww*http://www.stack.nl/~brama/*

option and supply the name of the play-list.

```
mp3blaster -a queen.lst
```

To load a list without starting to play, specify *-l* instead. To enable an infinite loop for the playlist, you can specify *-R* (for "repeat"):

```
mp3blaster -a queen.lst -R
```

Besides passing a playlist to the player, you can also specify multiple filenames. After playing these songs, mp3blaster will automatically quit. To prevent this from happening, you can add the *-q* parameter.

Fully Automatic

You can modify mp3blaster's appearance and behavior through the hidden config-uration file in your own home directory (*~/.mp3blasterrc*.) The program is sup-plied with a sample file that you can modify with any text editor. The *sample. mp3blasterrc* template file is normally
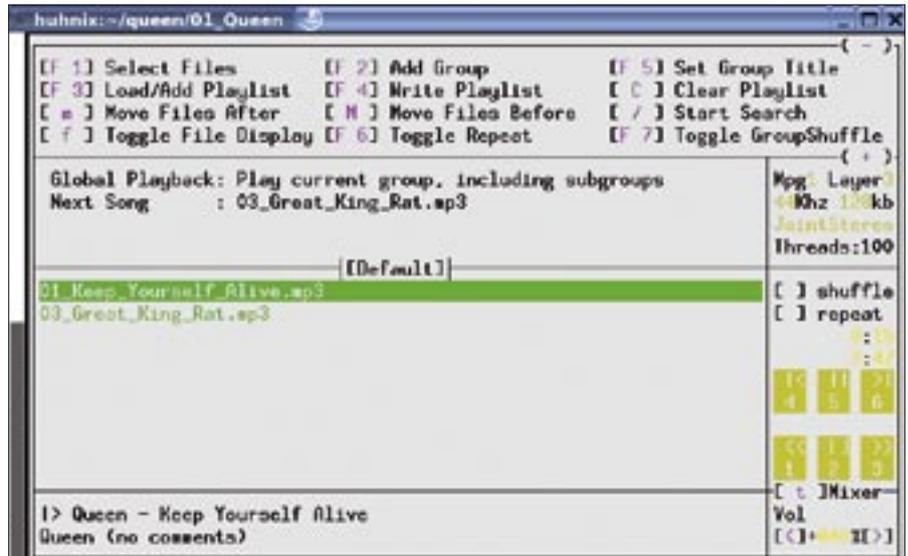

Figure 2: mp3blaster displaying the next track on the playlist.

located in */usr/share/mp3blaster/*, how-ever, Debian users will need to look for a packed file in */usr/share/doc/ mp3blaster/examples/*.

The text file has a list of commented entries, each line starting with a pound sign (*#*.) You can remove the pound sign

to enable a feature. Besides the pro-gram's general behavior, you can rede-fine the color scheme or even change the keyboard shortcuts. Check out the man-page for the player (*man mp3blaster*) for more tips and tricks on designing your own *~/.mp3blasterrc*. ■