

Conversión de formatos multimedia

(gnu/linux)

Una [conversión de formatos](#) consiste en transformar el contenido de los datos [multimedia](#) de un tipo de fichero a otro. En los casos de ficheros multimedia, esta conversión suele estar asociada con una [pérdida de datos](#) por lo que es importante asegurarse de que esta conversión es necesaria.

Un caso común es transformar un fichero [MPEG](#) a [3GP](#) para poder reproducir un vídeo en un dispositivo móvil. O un fichero de sonido [WAV](#) a [MP3](#) para reducir su tamaño.

Tabla de contenidos

- [1 Formatos de audio](#)
 - [1.1 WMA -> MP3](#)
 - [1.2 MP3 -> AMR](#)
 - [1.3 WAV -> AMR](#)
 - [1.4 MPEG -> MP3](#)
- [2 Formatos de vídeo](#)
 - [2.1 AVI -> FLV](#)
 - [2.2 3GP -> MPEG4](#)
 - [2.3 AVI -> VCD](#)
 - [2.4 MPEG -> 3GP](#)
 - [2.5 MPEG -> XviD](#)
 - [2.6 MPEG -> FLV](#)
 - [2.7 AVI -> DV](#)
 - [2.8 AVI -> PNG](#)
 - [2.9 JPG -> DivX](#)
 - [2.10 Incluir subtítulos en un vídeo](#)
- [3 Comprobar el resultado](#)
- [4 Otros conversores](#)

Formatos de audio

A continuación se enumeran algunas conversiones comunes en ficheros de sonido empleando [Ffmpeg](#).

WMA -> MP3

Tras el parametro *ab* especificaremos el [bitrate](#) del MP3 (192 en el ejemplo).

```
$ ffmpeg -i ficheroEntrada.wma -f mp3 -ab 192 ficheroSalida.mp3
```

MP3 -> AMR

```
$ ffmpeg -i music.mp3 -acodec amr_nb -ar 8000 -ac 1 -ab 32 music.amr
```

WAV -> AMR

```
$ ffmpeg -i music.wav -acodec amr_nb -ar 8000 -ac 1 -ab 32 music.amr
```

MPEG -> MP3

Extrae el audio de un fichero MPEG y lo pasa a MP3

```
$ ffmpeg -i video.mpg -f mp3 audio_track.mp3
```

Formatos de vídeo

A continuación se enumeran algunas conversiones comunes en ficheros de video empleando [Ffmpeg](#).

AVI -> FLV

```
$ ffmpeg -i movie.avi -acodec mp3 -ar 11025 movie.flv
```

3GP -> MPEG4

```
$ ffmpeg -i movie.3gp -vcodec mpeg4 -acodec mp3 movie.avi
```

AVI -> VCD

Añadiendo la opción `-hq` usa alta calidad.

```
$ ffmpeg -i myfile.avi -target pal-vcd myfile_vcd.mpg
```

MPEG -> 3GP

```
$ ffmpeg -i archivo.mpeg -s qcif -r 12 -ac 1 -ar 8000 -b 30 -ab 12 salida.3gp
```

O también con más calidad:

```
$ ffmpeg -i archivo.mpeg -s qcif -r 15 -ac 1 -ar 8000 -b 256000 -ab 15 salida.3gp
```

MPEG -> XviD

```
$ ffmpeg -i pelicula.mpg -acodec mp3 -vcodec xvid -b 687 pelicula.avi
```

MPEG -> FLV

```
$ ffmpeg -i pelicula.mpg -vcodec flv -y pelicula.flv
```

El fichero resultante no contiene correctamente la información de metadatos. Esto implica que cuando utilices el fichero en un visualizador de videos Flash como [Flash Video Player](#) el fichero se verá correctamente pero la barra de progreso no se actualizará. Para arreglar esto usa la utilidad **flvtool2** que la encontrarás en <http://inlet-media.de/flvtool2>. Está hecha en Ruby de modo que tendrás que instalar el paquete correspondiente. Se usa así:

```
$ flvtool2 -U pelicula.flv
```

También tenemos otra utilidad llamada [FLV MetaData Injector](#) que aunque sea para Windows, funciona perfectamente con [Wine](#) (por lo menos la versión de línea de comandos). Se usa así:

```
$ wine flvmdi.exe pelicula.flv
```

Otra forma de convertir a formato FLV es usando [Memcoder](#):

```
$ mencoder pelicula.avi -o pelicula.flv -of lavf -oac mp3lame -lameopts br=32 -af lavcresample=22050 -srate 22050 -ovc lavc -lavcopts vcodec=flv:vbitrate=340:autoaspect:mbd=2:trell:v4mv -vf scale=320:240 -lavfopts i_certify_that_my_video_stream_does_not_use_b_frames
```

AVI -> DV

Este es el formato Digital Video, el mismo que produce una cámara de vídeo digital y es el que se puede usar para editar con Kino.

```
$ ffmpeg -i pelicula.avi -target pal-dv pelicula.dv
```

Esto me da unos errores de sincronismo del audio que no veo que se noten. Si quieres evitarlos tendrás que hacerlo de esta forma:

```
$ mencoder pelicula.avi -ovc lavc -oac pcm -o pelicula-new.avi
$ ffmpeg -i pelicula-new.avi -s pal -r pal -ac 2 -ar 48000 pelicula.dv && rm pelicula-new.avi
```

Otra forma de hacerlo:

```
$ ffmpeg -i pelicula.avi -vcodec dvvideo -acodec copy -f dv pelicula.dv -hq
```

Kino también lee el formato AVI generado de esta forma (por ejemplo a partir de un FLV):

```
$ mencoder -oac mp3lame -ovc xvid -of avi -xvidencopts bitrate=1350 -o output.avi
input.flv
```

AVI -> PNG

```
$ ffmpeg -i swing.avi -vcodec png -vframes 1 -an -f rawvideo -s 320x240 swing1.png
```

JPG -> DivX

```
$ mencoder -mf on:w=800:h=600:fps=0.5 -ovc divx4 -o output.avi \*.jpg
```

Así te muestra una foto cada dos segundos, si quieres cada cuatro segundos deberías poner 0.25 en *fps*.

Incluir subtítulos en un vídeo

```
$ mencoder -ovc lavc -oac mp3lame pelicula.avi -o pelicula_con_subtitulos.avi -sub
subtitulos.srt
```

Comprobar el resultado

Si el formato de destino no lo soporta el reproductor multimedia [Totem](#) puedes usar la aplicación **ffplay** que viene en el paquete *ffmpeg*, reproducirá cualquier formato soportado por *ffmpeg*. Esto es útil, por ejemplo, para el formato de audio de móvil AMR.

Otros conversores

- [Convert for free video files direct to your Mobile phone, iPod, PSP, PC](#) Se trata de un servicio de conversión online, de modo que no tienes que tener nada instalado en tu PC.
- [Zamzar](#) Otro conversor de formatos online gratuito. Permite convertir tanto documentos como audio y video.
- [Mobile Media Converter](#) Es un conversor multiplataforma con un interfaz gráfico bastante sencillo. Funciona con una versión [Ffmpeg](#) compilada para soportar formatos adicionales.