

Conversiones multimedia

- [How do I convert 10-bit H.265 videos to H.264 without quality loss?](#)
- [Linux : convertir une vidéo H265 \(HEVC\) en H264 classique](#)
- [Converting Video Files to H.264 MP4 Format Using HandBrake on Linux](#)
- [Recortar un vídeo en partes con ffmpeg y sin transcodificar en Linux](#) 😎
- [Cómo unir vídeos en Linux por medio de FFmpeg](#) 😎
- [Cómo concatenar varios videos en un archivo en Linux en GUI sin volver a codificar \(HEVC y otros formatos de video\)](#) 😎

Convertir vídeo AVI en MKV sin pérdida de calidad

- Con **ffmpeg** podemos pasar de AVI a MKV sin recodificar, así que no hay pérdida de calidad porque solo cambiamos el contenedor.
- La conversión se hace copiando las pistas de vídeo y audio tal cual, lo que mantiene la calidad original y acelera mucho el proceso.
- **Comando básico**

```
ffmpeg -i entrada.avi -c copy salida.mkv
```

donde:

- `-i entrada.avi`: indica el archivo origen.
- `-c copy`: copia vídeo y audio sin volver a codificar.
- `salida.mkv`: archivo final en contenedor MKV.

- **Ejemplo práctico**

```
ffmpeg -i video.avi -c copy video.mkv
```

Si el AVI contiene códecs compatibles con MKV, esta es la forma ideal porque conserva la calidad original y termina rápido.

- **Comprobar el resultado.** Podemos verificar las pistas con:

```
ffprobe video.mkv
```

- **En caso en que no funcione.** Si alguna pista del AVI no es compatible con MKV, ffmpeg puede dar error o requerir recodificación de esa pista concreta. En ese caso ya no sería sin pérdida, porque habría que transcodificar solo lo necesario.

Comando para convertir en lote todos los AVI de una carpeta

- Desde dentro de la carpeta con los AVI:

```
for f in *.avi; do
  ffmpeg -i "$f" -c:v copy -c:a copy "${f%.avi}.mkv"
done
```

Ese bucle procesa cada `.avi` y genera un `.mkv` con el mismo nombre base.

- Variante más robusta. Si no queremos sobrescribir archivos ya existentes:

```
for f in *.avi; do
  out="${f%.avi}.mkv"
```

```
[ -e "$out" ] && continue
ffmpeg -i "$f" -c:v copy -c:a copy "$out"
done
```

- Si algún AVI tiene audio o vídeo en un códec poco compatible dentro de MKV, el contenedor puede aceptar la copia igual, pero si un reproductor falla entonces habría que recodificar solo esa pista concreta.

Convertir vídeo AVI en MP4 sin pérdida de calidad

- Para convertir un vídeo de formato AVI a MP4 desde la consola en Linux sin pérdida de calidad, la forma más efectiva es realizar una **copia directa de los *streams* (flujos) de datos**.
- Dado que el formato MP4 es un contenedor compatible con la mayoría de los códecs que suelen utilizar los archivos AVI (como H.264 o MPEG-4), podemos simplemente “reempaquetarlos” sin necesidad de volver a codificar el vídeo o el audio.
- **Comando para conversión sin pérdida.** Utilizaremos la herramienta ffmpeg con el parámetro copy

```
ffmpeg -i nombre_de_entrada.avi -c:v copy -c:a copy nombre_de_salida.mp4
```

Explicación de los parámetros:

- `-i nombre_de_entrada.avi`: Especifica el archivo de origen.
- `-c:v copy`: Indica a ffmpeg que copie el flujo de vídeo tal cual está, sin recodificarlo.
- `-c:a copy`: Indica que copie el flujo de audio directamente.
- `nombre_de_salida.mp4`: Es el nombre del nuevo archivo resultante.

Consideraciones importantes

- **Compatibilidad.** Este método funciona instantáneamente porque no procesa los datos, solo cambia el “contenedor”. Si el códec de vídeo original dentro del AVI fuera incompatible con el formato MP4 (algo poco común en archivos modernos), este comando fallaría o el archivo resultante podría no reproducirse correctamente.
- **Reinstalación.** Si no tenemos ffmpeg instalado, podemos instalarlo en sistemas basados en Debian/Ubuntu con `sudo apt install ffmpeg`.
- **Conversión masiva:** Si deseamos convertir todos los archivos `.avi` de una carpeta a `.mp4` de una sola vez, podemos usar este bucle en la terminal:

```
for i in *.avi; do ffmpeg -i "$i" -c:v copy -c:a copy "${i%.avi}.mp4"; done
```

- **Transcodificación**

- Si el archivo original utiliza códecs muy antiguos que no son compatibles con tus dispositivos finales (como reproductores DLNA o TV), el comando anterior no los hará compatibles, solo cambiará el contenedor.
- Si el archivo resultante no se reproduce correctamente tras la copia, necesitaremos recodificar el vídeo, por ejemplo, utilizando el códec H.264 (más compatible):

```
ffmpeg -i archivo_entrada.avi -c:v libx264 -crf 18 -c:a aac archivo_salida.mp4
```

Donde el valor `-crf 18` ofrece una calidad visualmente sin pérdida; valores más altos reducen el tamaño pero bajan la calidad.

From:

<https://euloxio.myds.me/dokuwiki/> - **Euloxio wiki**

Permanent link:

https://euloxio.myds.me/dokuwiki/doku.php/doc:tec:sis:gnu:media_convert:inicio?rev=1777887112

Last update: **2026/05/04 11:31**

