



2. PRAKTIKA – AUDIO ETA BIDEO

MULTIMEDIA ZERBITZUAK

HARITZ BLANCO ETA ENEKO GONZÁLEZ

2020/2021 IKASTURTEA

AURKIBIDEA

SARRERA.....	3
PAKETEAK ETA SOFTWARE FUNTZIONALITATEAK	4
1) Instalatu ffmpeg paketea eta begiratu:.....	4
- Zer pakete gehiago instalatzen dituen, hau da, pakete horren mendeko paketeak (pista: apt-cache show). Azaldu zertarako diren mendeko pakete hauek.	4
- Zer fitxero exekutableak instalatu dituen (pista: /usr/bin direktorioan daudenak).	4
2) Begiratu hurrengo ahalmenak:.....	5
- Zer formatu (kontainer) kudea ditzakeen. Hautatu:	5
- Bakarrik mux egiteko bi formatu.	5
- Bakarrik demux egiteko bi formatu.	6
- Bi formatu, bai mux eta bai demux egiteko.	6
- Zer kodek erabil ditzakeen:	6
- Hautatu bi bakarrik dekodifikatzeko, beste bi bakarrik kodifikatzeko, eta beste bi biak egiteko.	7
- Hautatu azpтитuluetako bi kodek eta dokumenta itzazu haien arteko desberdintasunak (adibidez Wikipedian begiratzen).	7
- Zer protokoloak lan egin ditzakeen.	8
- Zer filtro dauzkan. Azaldu zer egiten duten Bideoa bi, eta Audioa bat.	8
- Zer bitstream filter dauzkan. Azaldu zertarakoa horietako bietan.	9
MULTIMEDIA EDUKIAREN IZAERA.....	10
3) Multimedia fitxategien ezaugarriak:	10
- Aplikatu ezazu ffprobe komandoa argazki batean, egin berdina identify -ekin (libmagick liburutegikoa), eta azaldu emaitzak.	10
- Egin ezazu bash script bat. Berak, ffprobe erabiliz, direktorio batean dauden audio edo bideo fitxategi guztiei buruzko informazioa erakutsiko du.	11
- Youtubetik kalitate ezberdineko fitxategiak deskarga itzazu youtube-dl paketearen tresnaren bidez, eta begiratu ffprobe komandoa aplikatzeko emaitzak.	11
Oharra: youtube-dl bi bertsioak: sistema eragilearen paketekoa eta snap-ekoa.	11
KONBERTSIOAK	13
4) Konbertsio erakusleen konparaketa	13
- Direktorio batean utzi audio fitxategiak bakarrik, eta egin bash script bat fitxategi guztiak konbertitzeko, gutxienez hiru formatuetan uzten. Konpara ezazu tamainak eta konbertsio bakoitzaren denborak (pista: time, date).	13
- Errepikatu aurrekoa bideo fitxategiekin.	13
5) Bideo fitxategietan eragiketak	13

- Deskarga ezazu Youtubetik bideo bat, eta egin 10 segunduko mozketak bat bigarren minututik aurrera ffmpeg komandoaz. Zati hau hurrengo galderetako lagin gisa erabiliko da.	13
- Lagin horretan aplikatu aldatu framerate beste beheko balore batera eta begira ezazu emaitza ffplay -ren bidez.	13
- Aurrekoa errepika ezazu, orain bitrate murrizten.	14
- Banandu frameak jpeg irudietan.	14
- Atera ezazu audioa wav formatura.	14
- Prestatu beste bideo fitxategi bat eta jar ezaiozu lehenengoko laginaren audioa.	14
- Gehitu azpitutuluak, onartzen den formatu bat erabiliz.	15
6) preset-ak	15
- Instala ezazu winff-qt eta azaldu parametro bakoitza hurrengo preset konfigurazioetan:...	15
- Audio—Ac3 DVD-384kbps Stereo	16
- Bideo—Google WebM generic Fullscreen	16
- Bideo—Google WebM for Android2.3.3+Widescreen	16
- Bideo—iPod Small WideScreen Anamorphic	16
- Bideo—MPEG-4H.264 video (super high quality)	17

SARRERA

Hurrengo dokumentuan “Multimedia Zerbitzuak” irakasgaiaren laborategiaren bigarren praktikaren ebazpena eta emaitzak azalduko dira. Praktika honetan FFmpeg liburutegiaren erabilera lantzen da.

Horretarako, hasiera batean liburutegi honen instalakuntza eta oinarrizko funtzionalitateak aztertuko dira. Behin oinarriak argi izanda liburutegiaren ahalmena aztertuko da multimedia fitxategiekin lan egiten, konbertsioak eta eragiketak egiten. Amaitzeko, liburutegiaren funtzionalitateak ikusiko dira interfaze grafiko baten bidez.

PAKETEAK ETA SOFTWARE FUNTZIONALITATEAK

1) INSTALATU **FFMPEG** PAKETEA ETA BEGIRATU:

Ezer egin baino lehen liburutegiaren instalakuntza egin behar da. Horretarako, lehenengo urratsa gure sistema eragilearen egoera optimoa dela ziurtatzea, hau da, egin gabe dauden eguneraketa guztiak egin.

Behin hori eginda, instalazio berriak egiteko prest dago gure sistema. Beraz, gure liburutegi berria instalatu dezakegu.

- ZER PAKETE GEHIAGO INSTALATZEN DITUEN, HAU DA, PAKETE HORREN MENDEKO PAKETEAK (PISTA: APT-CACHE SHOW). AZALDU ZERTARAKO DIREN MENDEKO PAKETE HAUEK.

Honelako liburutegiak instalatzerakoan beste hainbat pakete instalatzen dira. Pakete horiek ezinbestekoak dira liburutegiaren funtzionamendurako. Hala ere, oso erabilgarria izan daiteke ezkutuan instalatzen denaren berri izatea. Horregatik, instalatzen diren pakete horiek ikusteko hurrengoko komandoa exekutatuko dugu:

\$ sudo apt-cache show ffmpeg | grep Depends

Hauek dira instalatzen diren paketeak eta bere funtzionalitateak:

- libavcodec58 (= 7:4.2.4-1ubuntu0.1): Audio eta bideo kodetu eta dekodetzeko.
- libavdevice58 (= 7:4.2.4-1ubuntu0.1): Sarrera eta irteera gailuak kudeatzen ditu.
- libavfilter7 (= 7:4.2.4-1ubuntu0.1): Multimedia iragazkiak. Fotogramak editatzeko liburutegia, grafikoetan oinarritua.
- libavformat58 (= 7:4.2.4-1ubuntu0.1): Multimedia kontainerrak multiplexatzeko eta desmultiplexatzeko.
- libavresample4 (= 7:4.2.4-1ubuntu0.1): Libavresample (lavr) audioaren birlaginketa, lagin-formatuen bihurtzea eta nahasketa maneiatzen dituen liburutegia da.
- libavutil56 (= 7:4.2.4-1ubuntu0.1): FFmpeg liburutegi guztiek partekatzen dute kode komuna.
- libc6 (>= 2.29): GNU C liburutegia.
- libpostproc55 (= 7:4.2.4-1ubuntu0.1): Bideo-prozesamendu ondoko liburutegia.
- libsdl2-2.0-0 (>= 2.0.10): DirectMediaren geruza sinplea.
- libswresample3 (= 7:4.2.4-1ubuntu0.1): Audioa berriz lagintzeko, lagin-formatuak bihurtzeko eta nahasteko liburutegia.
- libswscale5 (= 7:4.2.4-1ubuntu0.1): Koloreak eta eskalatzea bihurtzeko liburutegia.

Atal honen emaitza

- ZER FITXERO EXEKUTABLEAK INSTALATU DITUEN (PISTA: /USR/BIN DIREKTORIOAN DAUDENAK).

Horrekin batera, ffmpeg liburutegiarekin batera hainbat exekutagarri instalatzen dira. Exekutagarri horiek ikusteko /usr/bin direktorian begiratu behar da. Hala ere, direktorio

horretan exekutagarri asko eta jatorri desberdinekoak daude. Beraz, bilaketa ondo egiteko iragazketa bat egin beharko dugu. Horretarako hurrengo komandoa erabiliko dugu:

```
$ dpkg -L ffmpeg | grep bin/
```

Hori erabilita, hurrengo exekutagarriak daudela ikusi dezakegu:

- **/usr/bin/ffmpeg:** Multimedia-fitxategiak transkodetzeko, streaming egiteko eta erreproduzitzeko tresnak.
- **/usr/bin/ffplay:** FFplay multimedia erreproduzigailu sinple eta eramangarri bat da, FFmpeg eta SDL liburutegiak erabiltzen dituena. FFmpeg HJAetarako proba-banku gisa erabiltzen da batez ere.
- **/usr/bin/ffprobe:** Ffprobek multimedia-fluxuen informazioa biltzen du eta gizakientzat eta makinentzat irakurtzeko moduan inprimatzen du.
- **/usr/bin/qt-faststart:** QuickTime fitxategiak berrantolatzeke erabilgarria.

[Atal honen emaitza](#)

2) BEGIRATU HURRENGO AHALMENAK:

FFMpeg komandoak aukera asko eskaintzen ditu. Aukera horiek zeintzuk diren ikusteko “man ffmpeg” exekutatzera ikus daiteke. Horretan oinarrituta, atal honetan, aukera desberdin horien artean kasuren bat aztertuko dugu.

- ZER FORMATU (KONTAINER) KUDEA DITZAKEEN. HAUTATU:

Adibidez, aukera desberdin horien artean eskaintzen diren formatu edo kontainerak ikusi daiteke. Horretarako:

```
$ffmpeg -formats
```

Horien artean talde desberdinak daude:

- D: Demux egiten duen formatua.
- E: Mux egiteko formatua.
- DE: Mux eta Demux egiteko formatu.

Jarraian talde bakoitzeko formatuak zeintzuk diren jakiteko egin beharrekoa ikusiko dugu eta horien adibide batzuk aipatuko dugu.

[Atal honen emaitza](#)

- BAKARRIK MUX EGITEKO BI FORMATU.

Talde honetako formatuak ikusteko hurrengokoa exekutatu beharko genuke:

```
$ffmpeg -formats | grep "E"
```

Horrekin mux egiteko balio duten formatu guztiak ikusiko dira. Horien artean, adibidez, hurrengo bi formatu agertzen dira:

- MOV (QuickTime / MOV)
- DVD (MPEG-2 PS (DVD VOB))

- BAKARRIK DEMUX EGITEKO BI FORMATU.

Talde honetako formatuak ikusteko hurrengokoa exekutatu beharko genuke:

\$ffmpeg -formats | grep "D"

Horrekin demux egiteko balio duten formatu guztiak ikusiko dira. Horien artean, adibidez, hurrengo bi formatu agertzen dira:

- TEDCAPTIONS (TED Talks captions)
- PSXSTR (Sony Playstation STR)

- BI FORMATU, BAI MUX ETA BAI DEMUX EGITEKO.

Talde honetako formatuak ikusteko hurrengokoa exekutatu beharko genuke:

\$ffmpeg -formats | grep "DE"

Horrekin demux eta mux egiteko balio duten formatu guztiak ikusiko dira. Horien artean, adibidez, hurrengo bi formatu agertzen dira:

- MP3 (MP3 (MPEG audio layer 3))
- h264 (raw H.264 video)

- ZER KODEK ERABIL DITZAKEEN:

FFMpeg komandoaren beste ahalmen bat liburutegiak lantzen dituen kodek-ak dira. Horiek ikusteko hurrengokoa exekutatu dugu:

\$ ffmpeg -codecs

Bertan kodek mota desberdinak aurkituko ditugu:

Codecs:

D..... = Decoding supported.

.E.... = Encoding supported.

..V... = Video codec.

..A... = Audio codec.

..S... = Subtitle codec.

...I.. = Intra frame-only codec.

....L. = Lossy compression.

.....S = Lossless compression.

Atal honen emaitza

-HAUTATU BI BAKARRIK DEKODIFIKATZEKO, BESTE BI BAKARRIK KODIFIKATZEKO, ETA BESTE BI BIAK EGITEKO.

Kodek mota desberdinen artean bat bakarrik bistaratu daiteke. Ikus dezagun adibide batzuk:

D motakoak:

Hauek ikusteko hurrengokoa exekutatu beharko zen:

```
$ ffmpeg -codecs | grep "D\."
```

Horrela dekodifikatzeko bakarrik balio duten kodek-ak ikusiko genituen. Haien artean:

- (D.V..S) VMNC - (VMware Screen Codec / VMware Video).
- (D.V.L.) VP3 - (On2 VP3).

E motakoak:

Hauek ikusteko hurrengokoa exekutatu beharko zen:

```
$ ffmpeg -codecs | grep "\.E"
```

Horrela kodifikatzeko bakarrik balio duten kodek-ak ikusiko genituen. Haien artean:

- (.EVI.S) LIPEG - (Lossless JPEG).
- (.EA...) SONICLS - (Sonic lossless).

DE motakoak:

Hauek ikusteko hurrengokoa exekutatu beharko zen:

```
$ ffmpeg -codecs | grep "DE"
```

Horrela kodifikatzeko eta dekodifikatzeko balio duten kodek-ak ikusiko genituen. Haien artean:

- (DEV.L.) MPEG4 - (MPEG-4 part 2 (decoders: mpeg4 mpeg4_v4l2m2m mpeg4_cuvd) (encoders: mpeg4 libxvid mpeg4_v4l2m2m)).
- (DEA.L.) MP3 - (MP3 (MPEG audio layer 3) (decoders: mp3float mp3) (encoders: libmp3lame libshine)).

-HAUTATU AZPITITULUETAKO BI KODEK ETA DOKUMENTA ITZAZU HAIEN ARTEKO DESBERDINTASUNAK (ADIBIDEZ WIKIPEDIAN BEGIRATZEN).

Azpitituluetakoa kodek-ak ikusteko hurrengokoa exekutatu behar da:

```
$ ffmpeg -codecs | grep "S\."
```


Bertan agertzen diren guztien artean bi hauek aurki dezakegu:

- (DES...) ASS - (ASS(Advanced SSA)subtitle(decoders: ssa ass)(encoders: ssa ass)).
- (...S...) TTML - (Timed Text Markup Language).

Hauen arteko ezberdintasuna aztertuko dugu.

- ASS: Substation Alpha (edo Sub Station Alpha), laburki SSA, SubRip eta beste formatu batzuk baino azpititulu formatu aurreratuagoa da. Jatorrian, Kotusek garatu zuen, bere programa ezagunarekin editatua izateko, gaur egun ez zena, azpitituluak bezala deitua.
- Kodetzeko eta dekodetzeko balio duen azpitituluen kodeketarako kodek-a da.
- TTML: Touned Text Markup Language (TTML), lehen banaketa-formatua (DFXP) bezala aipatua, online hedabideetako testu tenporizatueterako XML-oinarritutako W3C estandarra da, eta diseinatua izan zen testu-informazioa baimentzeko, aldatzeko edo trukatzeko, lehentasunez erabiltzeko.
- TTML kodek-a azpitituluen kodeketarako baino ez du balio.

- ZER PROTOKOLOAK LAN EGIN DITZAKEEN.

FFmpeg zenbait protokolo lan egin dezake. Aurreko kasuetan gertatu den moduan hauek ikusteko hurrengokoa exekutatu behar da:

\$ ffmpeg -protocols

Bertan bi mota bereizten dira:

- Sarrerakoak: async, bluray, cache, concat, crypto, data, ffrtmphttp, file, ftp, gopher, hls, http, httpproxy, https, mmsh, mmst, pipe, rtmp, rtmps, rtmpt, rtmps, rtp, sctp, srtp, subfile, tcp, tls, udp, udplite, unix, sftp.
- Irteerakoak: crypto, ffrtmphttp, file, ftp, gopher, http, httpproxy, https, icecast, md5, pipe, prompeg, rtmp, rtmps, rtmpt, rtmps, rtp, sctp, srtp, tee, tcp, tls, udp, udplite, unix, sftp.

[Atal honen emaitza](#)

- ZER FILTRO DAUZKAN. AZALDU ZER EGITEN DUTEN BIDEOKO BI, ETA AUDIOKO BAT.

Filtroei dagokionez, ffmpeg liburutegiak hainbat ditu. Horiek ikusteko hurrengokoa exekutatu beharko genuke:

\$ ffmpeg -filters

Bertan mota desberdinekoak aurkitu dezakegu:

Filters:

T.. = Timeline support

.S. = Slice threading

..C = Command support

A = Audio input/output

V = Video input/output

N = Dynamic number and/or type of input/output

| = Source or sink filter

Kasu honetan bideoa bi adibide eta audioko bat aukeratuko dugu.

- Video:
 - (TS.) LUT - V->V Compute and apply a lookup table to the RGB/YUV input video.
 - (T.C) HUE - V->V Adjust the hue and saturation of the input video.
- Audio:
 - (T..) ADELAY - A->A Delay one or more audio channels.

[Atal honen emaitza](#)

- ZER BITSTREAM FILTER DAUZKAN. AZALDU ZERTARAKOA HORIETAKO BIETAN.

Hainbat bitstream filtroak daude. Horiek ikusteko hurrengokoa exekutatu beharko genuke>

\$ ffmpeg -bsfs

Horrekin hurrengoko bitstream iragazkiak daudela ikusiko genuke:

Bitstream filters: aac_adtstoasc, av1_frame_split, av1_metadata, chomp, dump_extra, dca_core, eac3_core, extract_extradata, filter_units, h264_metadata, h264_mp4toannexb, h264_redundant_pps, hapqa_extract, hevc_metadata, hevc_mp4toannexb, imxdump, mjpeg2jpeg, mjpegadump, mp3decomp, mpeg2_metadata, mpeg4_unpack_bframes, mov2textsub, noise, null, prores_metadata, remove_extra, text2movsub, trace_headers, truehd_core, vp9_metadata, vp9_raw_reorder, vp9_superframe, vp9_superframe_split.

Horietako bi azalduko ditugu:

- prores_metadata: Koloretako metadata aldatu prores korrontean txertatuta.
- chomp: Kendu zero zulaketa pakete baten amaieran.

[Atal honen emaitza](#)

MULTIMEDIA EDUKIAREN IZAERA

3) MULTIMEDIA FITXATEGIEN EZAUGARRIAK:

Praktikarekin jarraitzeko, multimedia fitxategien ezaugarriak aztertzeari ekingo diogu.

- APLIKA EZAZU **FFPROBE** KOMANDOAK ARGAZKI BATEAN, EGIN BERDINA **IDENTIFY**-REKIN (LIBMAGICK LIBURUTEGIKOA), ETA AZALDU EMAITZAK.

Atal hau burutzeko, pasaden praktikan erabili genuen argazkia erabiliko dugu. Argazki hau hurrengokoa da:



Irudia 1. Erabilitakoirudia.

FFPROBE

Irudiaren ezaugarriak ikusteko ffprobe erabiliz hurrengoko komandoa erabili dugu:

```
$ ffprobe imagen400x300.jpg
```

Komando honen emaitza:

```
Input #0, image2, from 'imagen400x300.jpg':  
Duration: 00:00:00.04, start: 0.000000, bitrate: 4131 kb/s  
Stream #0:0: Video: mjpeg (Baseline), yuvj420p(pc, bt470bg/unknown/unknown), 400x300 [SAR 1:1 DAR 4:3], 25 tbr, 25 tbn, 25 tbc
```

Irudia 2. Ffprobe komandoaren emaitza.

IDENTIFY

Orain berdina egin dugu baina ffprobe erabili beharrean identify erabiliz:

```
$ identify imagen400x300.jpg
```

Honen emaitza:

```
imagen400x300.jpg JPEG 400x300 400x300+0+0 8-bit sRGB 20656B 0.000u 0:00.000
```

Irudia 3. Identify komandoaren emaitza.

Emaitzei erreparatuz, ikusi ahal da identify komandoak ffprobe komandoak baino informazio gehiago ematen duela. Identify komandoak irudiaren formatu eta ezaugarrien informazioa ematen du. Beste aldetik, ffprobe-k multimedia fluxuetako informazioa biltzen du eta modu ulergarriago batean erakusten digu pantailan, hau da, irudiari buruzko informazioa ematen du modu ulergarrian.

- EGIN EZAZU BASH SCRIPT BAT. BERAK, **FFPROBE** ERABILIZ, DIREKTORIO BATEAN DAUDEN AUDIO EDO BIDEO FITXATEGI GUZTIEI BURUZKO INFORMAZIOA ERAKUTSIKO DU.

Atal honetan garatu dugun script-a hurrengokoa da:

```
#!/bin/bash
for fitxategia in /home/haritzblanco/Desktop/2.Praktika/AudioEtaBideoak/*
do
    ffprobe "$fitxategia"
done
```

Irudia 4. Script-3_2.sh-ren edukia.

Bash fitxategi hau exekutatzeko hurrengoko komandoa erabili dugu:

```
$ sh script-3_2.sh
```

[Atal honen emaitzak](#)

- YOUTUBETIK KALITATE EZBERDINEKO FITXATEGIAK DESKARGA ITZAZU **YOUTUBE-DL** PAKETEAREN TRESNAREN BIDEZ, ETA BEGIRATU **FFPROBE** KOMANDOAK APLIKATZEKO EMAITZAK.

OHARRA: YOUTUBE-DL BI BERTSIOAK: SISTEMA ERAGILEAREN PAKETEKOA ETA SNAP-EKOA.

Atal honekin amaitzeko, lehenengo youtube-dl erabilgarritasuna deskargatu dugu hurrengoko komandoa erabiliz.

```
$ sudo snap install youtube-dl
```

Jarraian dauden kalitate aukerak ikusteko hurrengo komandoa erabili dugu, horren aurretik aldagai batean deskargatu nahi dugun bideoaren URL-a adieraziz.

Aldagaia sortu:

```
$URL="https://www.youtube.com/watch?v=GX8Hg6kWQYI&ab_channel=XXXTENTACION"
```

Kalitate aukerak zeintzuk diren ikusteko komandoa:

```
$ /snap/bin/youtube-dl -F $URL
```

Eta orain hauetako bi aukeratu eta deskargatu ditugu:

```
$ /snap/bin/youtube-dl -f 18 $URL
```

```
$ /snap/bin/youtube-dl -f 399 $URL
```

Behin bideoak deskargatuta, ffprobe komandoa erabili dugu emaitzak alderatzeko. Aldaketak ikusi ditugun eremuak hurrengokoak izan dira:

bideoa18_ffprobe.txt

major_brand: mp42

compatible_brands: isommp42

bitrate: 369 kb/s

Video: h264 (Constrained Baseline) (avc1 / 0x31637661), yuv420p(tv, bt709), 640x360 [SAR 1:1 DAR 16:9], 271 kb/s, 23.98 fps, 23.98 tbr, 24k tbn, 47.95 tbc (default)

Bideoa399_ffprobe.txt

major_brand: dash

compatible_brands: iso6av01mp41

bitrate: 817 kb/s

Video: av1 (Main) (av01 / 0x31307661), yuv420p(tv, bt709), 1920x1080, 12 kb/s, 23.98 fps, 23.98 tbr, 24k tbn, 24k tbc (default)

[Atal honen emaitzak](#)

KONBERTSIOAK

4) KONBERTSIO ERAKUSLEEN KONPARAKETA

Atal honetan ffmpeg liburutegiak eskaintzen dituen konbertsio aukerak aztertuko ditugu.

- DIREKTORIO BATEAN UTZI AUDIO FITXATEGIAK BAKARRIK, ETA EGIN BASH SCRIPT BAT FITXATEGI GUZTIAK KONBERTITZEKO, GUTXIENEZ HIRU FORMATUETAN UZTEN. KONPARA EZAZU TAMAINAK ETA KONBERTSIO BAKOITZAREN DENBORAK (PISTA: TIME, DATE).

Lehenengo urratsa, audio fitxategien konbertsioak aztertuko dira. Horretarako, bash script bat sortuko dugu.

Script horretan, konbertsioak egin nahi ditugun direktorian for bat egingo dugu banan banan bertan dauden audioak konbertitzeko. Prozesua audioaren izena hartu eta ffmpeg -i komandoa erabilia konbertsioa egitean datza.

Script-a .sh fitxategia izango da eta exekutatzeko hurrengokoa egingo dugu:

```
$ chmod +x p2-4.sh
```

```
$ ./p2-4.sh
```

[Atal honen emaitza](#)

- ERREPIKATU AURREKOA BIDEO FITXATEGIEKIN.

Bideoekin egin beharreko prozesua berdina da. Hala ere, formatu desberdinak erabili behar dira.

[Atal honen emaitza](#)

5) BIDEO FITXATEGIETAN ERAGIKETAK

Atal honetan erabilitako bideoa hirugarren atalean deskargatutako bat aukeratu dugu

- DESKARGA EZAZU YOUTUBETIK BIDEO BAT, ETA EGIN 10 SEGUNDUKO MOZKETA BAT BIGARREN MINUTUTIK AURRERA **FFMPEG** KOMANDOAZ. ZATI HAU HURRENGO GALDERETAKO LAGIN GISA ERABILIKO DA.

Atal hau burutzeko erabili dugun komandoa honako hau da:

```
$ ffmpeg -i bideoa18.mp4 -ss 00:02:00 -to 00:02:10 -c copy bideoaMoztuta.mp4
```

[Atal honen emaitza](#)

- LAGIN HORRETAN APLIKA ALDATU FRAMERATE BESTE BEHEKO BALORE BATERA ETA BEGIRA EZAZU EMAITZA **FFPLAY**-REN BIDEZ.

Atal hau burutzeko lehenik eta behin bere balioa zein den begiratu dugu hurrengoko komandoarekin:

```
$ ffmpeg -i bideoaMoztuta.mp4
```

Ondoren, hurrengoko komandoak erabili ditugu framerate-a txikitzeko eta emaitza bistaratzeko:

```
$ ffmpeg -i bideoaMoztuta.mp4 -filter:v fps=fps=4  
bideoaMoztutaFramerate.mp4
```

```
$ ffplay bideoaMoztutaFramerate.mp4
```

[Atal honen emaitza](#)

- AURREKOA ERREPIKA EZAZU, ORAIN BITRATE MURRIZTEN.

Orain bitrate-a murrizteko hurrengoko komandoa erabili dugu:

```
$ ffmpeg -i bideoaMoztuta.mp4 -b:v 144k bideoaMoztutaBitrate.mp4
```

```
$ ffplay bideoaMoztutaBitrate.mp4
```

[Atal honen emaitza](#)

- BANANDU FRAMEAK JPEG IRUDIETAN.

Atal honetan erabili dugun komandoa hurrengokoa izan da:

```
$ ffmpeg -i bideoaMoztuta.mp4 %dframe.jpg
```

[Atal honen emaitza](#)

- ATERA EZAZU AUDIOA WAV FORMATURA.

Orain bideotik audio ateratzeko hurrengoko komandoa erabili dugu:

```
$ ffmpeg -i bideoaMoztuta.mp4 bideoarenAudioa.wav
```

[Atal honen emaitza](#)

- PRESTATU BESTE BIDEO FITXATEGI BAT ETA JAR EZAIOZU LEHENENGOKO LAGINAREN AUDIOA.

Atal hau gauzatzeko aurreko ataletan gauzatutako komandoekin beste bideo bat deskargatu eta prestatu dugu hurrengoko komandoekin:

```
$ URL2="https://www.youtube.com/watch?v=UVv1K3bx4ak&ab_channel=Wa  
rnerMusicSpain"
```

```
$ /snap/bin/youtube-dl -F $URL2
```

```
$ ffmpeg -i bideoa2.mp4 -ss 00:02:00 -to 00:02:10 -c copy bideoa2Moztuta.mp4
```

Hau eginda prest gaude bideoari audio berria jartzeko. Hau egiteko hurrengoko komandoa erabili dugu:

```
$ ffmpeg -i bideoa2Moztuta.mp4 -i bideoarenAudioa.wav -map 0:v -map 1:a -strict 2 -vcodec copy bideoa2bideo1renAudioarekin.mp4
```

[Atal honen emaitza](#)

- GEHITU AZPITUTULUAK, ONARTZEN DEN FORMATU BAT ERABILIZ.

Azkenik, azpitutuluak gehituko diogu bideoari. Lehenik eta behin nahi ditugun azpitutuluak .srt fitxategi batekin sartu ditugu. Fitxategi horren edukia:

```
1
00:00:01,001 --> 00:02:30,007
<font color="#EA899A">Haritz Blanco eta Eneko Gonzalez</font>
```

[Irudia 5. Azpitutuluak.srt fitxategiaren edukia.](#)

Orain fitxategi hau .ass fitxategi batera bihurtu behar izan dugu. Hau egiteko hurrengoko komandoa erabili dugu:

```
$ ffmpeg -i azpitutuluak.srt azpitutuluak.ass
```

```
[Script Info]
; Script generated by FFmpeg/Lavc58.54.100
ScriptType: v4.00+
PlayResX: 384
PlayResY: 288

[V4+ Styles]
Format: Name, Fontname, Fontsize, PrimaryColour, SecondaryColour, OutlineColour, BackColour, Bold, Italic, Underline, StrikeOut, ScaleX, ScaleY, Spacing, Angle, BorderStyle, Outline, Shadow, Alignment, MarginL, MarginR, MarginV, Encoding
Style: Default,Arial,-16,&Hffffff,&Hffffff,&H0,&H0,0,0,0,0,100,100,0,0,1,1,0,2,10,10,10,0

[Events]
Format: Layer, Start, End, Style, Name, MarginL, MarginR, MarginV, Effect, Text
Dialogue: 0,0:00:01.00,0:02:30.01,Default,,0,0,0,,{\c&H9A89EA&}Haritz Blanco eta Eneko Gonzalez{\c}
```

[Irudia 6. Azpitutuluak.ass fitxategiaren edukia.](#)

Azkenik, bideoari azpitutuluak jartzeko erabili dugun komandoa honako hau izan da:

```
$ ffmpeg -i bideoa18.mp4 -vf ass=azpitutuluak.ass bideoaAzpitutuluekin.mp4
```

[Atal honen emaitza](#)

6) PRESET-AK

Azken atalean preset-ak aplikatu eta aztertu ditugu.

- INSTALA EZAZU **WINFF-QT** ETA AZALDU PARAMETRO BAKOITZA HURRENGO PRESET CONFIGURAZIOETAN:

Winff-qt programa instalatzeko hurrengoko komandoa erabili dugu:

```
$ sudo apt-get install winff-qt
```


Atal honetan, winff-qt programa erabiliz fitxategiak bihurtuko ditugu preset desberdinak erabiliz eta gero, ffprobe-ren bitartez, hauek aztertuko ditugu.

- AUDIO–AC3 DVD-384KBPS STEREO

AC3 (Audio Coding 3) audio fitxategietarako luzapena da. “Dolby Laboratories”-ek sortu zuen 1987an DVDetan, Blu-ray erreproduktoreetan, HDTV programazioan eta etxeko entretenimendu sistemetan erabiltzeko. AC3 formatuak 6 soinu-kanal diskretu ditu. Normalean erabiltzen diren 5 audio kanalak maila normaleko bozgorailuei eskainiak daude (20-20,000Hz) eta kanal bat frekuentzia baxuko subbufer efakturako da (20-120Hz).

Bihurketa honetan, ffprobe egitean ikus daitekeen moduan, 384 kbps-eko bitrate-arekin sortzen da eta ac3 audio formatuarekin.

[Atal honen emaitza](#)

- BIDEO–GOOGLE WEBM GENERIC FULLSCREEN

Google WebM multimedia fitxategietarako Google-ek garatutako formatu libre bat da eta HTML5ekin erabiltzeko pentsatua dago. VP8 bideo kodek-arekin erabiltzeko pentsatua izan zen Vorbis audio kodek-arekin batera, guztia Matroska multimedia edukiontzi baten barruan sartuta.

Kasu honetan, ffprobe komandoari erreparatuz gero, ikus dezakegu 778 kbps-ko bitratearekin sortu dela, bideo VP8 kodek-arekin eta Vorbis audio kodek-arekin.

[Atal honen emaitza](#)

- BIDEO–GOOGLE WEBM FOR ANDROID2.3.3+WIDESCREEEN

Android 2.3.3an Google WebM edukiontziak erabiltzeko formatua da, widescreen pantailetan. Hau da, 16:9 formatuarekin.

Ffprobe eginez gero, ikus daiteke Matroska edukiontzia erabili dela eta webM formatua. Bideo eta audio kodek-ei erreparatuz gero ikus dezakegu bideorako VP8 erabili dela 621 kbps bitrate-arekin eta audiorako Vorbis.

[Atal honen emaitza](#)

- BIDEO–IPOD SMALL WIDESCREEEN ANAMORPHIC

Ipod Small widescreen Anamorphic-en pantaila panoramiko bat horizontalki konprimituko da memoria euskarri batean sartu eta ikusi ahal izateko.

Bihurketa honetan kusi dugu 272 kbps-ko bitrate bat erabiltzen dela h264 kodek-arekin bideorako eta aac audiorako.

[Atal honen emaitza](#)

- BIDEO-MPEG-4H.264 VIDEO (SUPER HIGH QUALITY)

MPEG-4 H.264 Video, FullHD eta audio seinaleak grabatu eta zabaltzeko bideo kodetze formatu bat da, ITU-T Video Coding Experts Group-ek sortutakoa.

Azkenik, ffmpeg bidez aztertuz behatu dugu izenak adierazten duen moduan bideorako H.264 kodek-a erabiltzen duela 2862 kbps-ko bitrate batekin eta audiorako aac kodek-a.

[Atal honen emaitza](#)