Encoding Videos on Unix Systems

Reduce Video size with *FFMPEG*

We can encode any video into **H.265** because, * Better compression rate without losing quality.

Step 1

ffmpeg -i input.mp4 -vcodec libx265 -crf 28 output.mp4

Thats it done….

- **crf 24** would be a better balanced rate.
- *Note that lower CRF values correspond to higher bitrates, and hence produce higher quality videos.*

FFMPEG with GPU Encoding

- *ffmpeg -vsync 0 –hwaccel cuvid -c:v h265_cuvid -i input.mp4 -c:a copy -c:v h265_nvenc output.mkv*

Mixing CPU and GPU processing

- *ffmpeg -vsync 0 -c:v h264_cuvid -i input.264 -vf "fade,hwupload_cuda,scale_npp=1280:720" -c:v h264_nvenc output.264*

Multi-GPU

- *ffmpeg -vsync 0 -hwaccel cuvid -hwaccel_device 1 -c:v h264_cuvid -i input.mp4 -c:a copy -c:v h264_nvenc -b:v 5M output.mkv*

Best Settings

- *ffmpeg -strict 2 -hwaccel auto -i "inputfile.mp4" -c:v hevc_nvenc -rc vbr -cq 24 -qmin 24 -qmax 24 -profile:v main10 -pix_fmt p010le -b:v 0K -c:a aac -map 0 "outputfile.mp4"*

Othertools

1. handbrake
2. mkvmerge

Handbrak

HandBrakeCLI -i /file/input.mp4 -o /file/out.mp4 -E fdk_faac -B 96k -6 stereo -R 44.1 -e x264 -q 27 -x cabac=1:ref=5:analyse=0x133:me=umh:subme=9:chroma-me=1:deadzone-inter=21:deadzone-intra=11:b-adapt=2:rc-lookahead=60:vbv-maxrate=10000:vbv-bufsize=10000:qpmax=69:bframes=5:b-adapt=2:direct=auto:crf-max=51:weightp=2:merange=24:chroma-qp-offset=-1:sync-lookahead=2:psy-rd=1.00,0.15:trellis=2:min-keyint=23:partitions=all