Codec Corner

422, H.264, NotchLC, HAP, etc

- Video Codec Overview
- The HAP Codec

Video Codec Overview

> Codec Categories, Codec Types, Use-Cases, Performance, Color Depth

Codecs are video encode file types. Almost all of them support at least 2-channels (left and right) of audio. Some formats, like HAP, support up to 8-channels.

Codec Categories

1. Shooting Codecs
2. Edit (and proxy) Codecs
3. Delivery/Daily/Show Codecs

Codec Types

- Apple Pro Res (also called “Pro Rez” or “APR”) - all files used the “.mov” extension
  - 4444 (also known as “Quad 4”) - supports Alpha (transparent video) - 12-bit
  - 422 HQ - 10-bit (as are all other forms of APR)
  - 422 (this is usually the file type someone is referring to when they say “Pro Rez”)
  - 422 LT
  - 422 Proxy (lightest)
- WEBM
  - Video format for web, pain in the ass to encode in AME - better to use FFMPEG
- MP4
  - H265 aka HEVC (better than H264 but not always supported) - supports alpha and is 8-bit - still no method to encode 265+A in AME or FFMPEG. You have to right click the intermediary alpha export, only on a mac (dumb), right click something like “convert video”. Maybe you can do it in Compressor or FCPX but that’s not something I’ve done because RIP FCP 7.
  - H264 - web friendly pretty much everywhere - sometimes used for proxy (offline) editing
- AV1 - is a royalty-free open video codec that directly rivals HEVC/H.265. It is commonly found in YouTube video streams, and is not fully supported by Apple. It is not recommended for live video playback or fast scrubbing.
- RAW
  - R3D
  - Blackmagic Raw
- HAP - all versions support 8-channels of audio, Cam (me) has a bus map Premiere template that will work. You have to export to 422 first and then transcode using FFMPEG – here's the FFMPEG Cheat Sheet.

    - HAPQ
    - HAP Alpha
    - HAP
    - HAPr - a better HAP with gradients similar to NotchLC, but unfortunately, not as much application support.
- NotchLC or **NLC** - Like HAP, but way better with gradients. Zippier to transcode, as well. Apple Silicone version of the encoder for AME is great with some unfortunate crash bugs though. No FFMPEG support as the Notch wants to keep it 100% proprietary.

To audit what versions of codecs you have installed on a Mac OS system, go to System Information, then scroll down to “Installations” then scroll down to the codec name. This is useful in debugging when you have render gremlins on one system, but not another.

Shooting Codecs

Shooting codecs are what comes out of a camera natively. On a better camera, you are going to be shooting APR or RAW. RAW is usually proprietary to a camera manufacturer. On a lesser camera, you’ll be shooting h264 or h265. Some cameras have RAW recording only through an external recorder – for example, the Fuji XT3 can record APR 422 at 60, but only through something like an Atmos Ninja or BMD recorder. To make things even more confusing, RAW isn't necessarily a format – it can also mean Log footage within a different codec container. Fuji's RAW format is Log 422.

Edit and Proxy Codecs

If your camera codec shoots an edit codec that your computer can handle efficiently, you’re good to go without making proxies.

Proxies are lighter weight files used for editing. This is done when your computer can’t handle the original files or if you want to edit “offline”. (offline, meaning, not using the original files). Generally speaking, colorists and VFX/mograph folks do not / should not use offline files.

Delivery, Show, and Daily Codecs

Delivery codecs are file types used to deliver your final exports. These are also the “archival” copies.

To client, these are almost always Pro-Res files and/or web-optimized H264s.

For shows, like events or installations, the codec is almost always HAP, NLC, or Pro-Res

For dailies, the codec is almost always H264, H265, or 422 Proxy files.

For masters, these are most likely APR 4444.

Color Depths

Different codecs have different bit depths. In general:

- 8 BPC = 8 Bits Per Channel (Millions of Colors)
- 16 BPC = 16 Bits Per Channel (Trillions of Colors)
- 32 BPC = 32 Bits Per Channel (Floating Point)

Encoding / Transcoding Software

See this directory for info – but to name a few: Adobe Media Encoder, AfterCodecs, FFMPEG, Handbrake, Jokyo HAP Encoder, ShutterEncoder

The HAP Codec

Everyone's favorite open source codec developed by VIDVOX

HAP and it's original varieties (Hap, Hap Alpha, Hap Alpha-Only, Hap Q, Hap Q Alpha) can be encoded and decoded using many applications – like FFMPEG or ShutterEncoder.

HAP is performatively very good and it can be tuned to specific systems to take advantage of processor cores (chunk count = number of cores to encode / decode).

Supported Playback Applications Include : Isadora, MadMapper, TouchDesigner, QLab

HAP Oral History

VIDVOX built the original version of the codec, and if you wanted to encode to HAP you could install that version to your system and anywhere you can encode Quicktime, you could encode HAP. Simultaneously, the core versions of HAP were included as part of FFMPEG. Eventually, the original codec that you'd use for Adobe CC wasn't working anymore and Disguise took it over and made their own Adobe CC Integration (hap-encoder-adobe-cc git repo). Last checked, the Disguise version also didn't work when installed vanilla-style. You can probably manually install it, but it's no longer supported (FFMPEG continues to work).

HAP In 2025

If you're encoding HAP's original varieties, you can use FFMPEG or an FFMPEG top like ShutterEncoder. Works great.

If you're encoding to HAPr varieties (or want to encode original/any variety in CC), you need to use AfterCodecs or Jokyo. Both have CC integrations or stand-alone versions. Unfortunately, neither adds support for FFMPEG.

HAPr is supported more and more, but is not yet supported using MadMapper – Jokyo has a great breakdown on their main page showing which applications support which versions of HAP.

> Why HAPr? Faster encode. Better colors. Much better with gradients and banding. Cleaner alpha, etc.