



**Guide to Codecs:**  
The Most Commonly Used Codecs  
& Standard Use Cases



We designed this guide as a reference for the most commonly used codec you will run into in your work in motion picture post production. It can't serve as a universal encyclopedia of codecs; there are just too many to count, and new special purpose formats arrive seemingly every month.

Motion picture post production has luckily settled on a few commonly used codecs that have a large footprint in the industry, and a good working knowledge of each will help you tremendously as you go about your work.

## Apple ProRes

**Apple ProRes** is currently the most widely used codec in all of motion picture post production. If you work in a Mac shop, and Macs continue to dominate a lot of post, you'll likely run into ProRes on a daily basis. In fact, one of the factors that keep Macs on top in motion picture post is the functionality and ubiquity of ProRes.

ProRes is used all the way from image capture in major platforms like the Arri Alexa, through editing in any of the four major NLE platforms, all the way through delivery, with streamers and major networks accepting ProRes file for delivery. If you are worried about the drawbacks of transcoding from one format to another, ProRes avoids those issues with an end-to-end pipeline that stays in one codec throughout.

Though you shouldn't be afraid of transcoding for your edit workflow, you can always reconnect back to another format at the end in your online color session. Whatever format you shoot, you'll be happier with your edit if you take the time to transcode to ProRes, especially an edit-friendly flavor.

If you are going to be working with ProRes extensively, it's well worth a read of the **ProRes Whitepaper**, the technical sheet that Apple keeps alive spelling out the ins and outs of ProRes as a format.

The basics to understand is that Apple ProRes isn't just a single codec, but a family of codecs built around the same technology, available in multiple implementations. You can think of these as



“flavors” or “strengths” of ProRes. These flavors refer to both the method of encoding the image, 422 or 4444, and the data rate, how many Mb per second are allocated to creating the image. The higher the data rate, the higher quality the image reproduction will be, with fewer artifacts, but on the flip side, the larger the file will be.

The data rate scales with the image size, meaning that a given flavor of ProRes will be a much bigger file if you shoot a higher resolution and a smaller file if you shoot a smaller resolution. You can see the strengths in the following chart, combined with their data rate when working at 1080p 29.97.

This starts all the way at the smallest file sizes with ProRes Proxy and goes up to the currently largest file sizes of ProRes XQ. You might shoot your film and capture it in XQ, then transcode it to LT for editing, then reconnect back to the XQ’s for color grading, then deliver to your network in 4444.

4444 and XQ both support Alpha Channels (that’s the fourth four in 4444), which allows for passing transparency information back and forth with VFX platforms like After Effects, Fusion and Nuke. Most VFX houses work on PC platforms and prefer to get files delivered as image sequences (discussed below), but there is increasing use of 4444 and XQ for some motion graphics and VFX workflows.

For many years, ProRes support on Windows was relatively weak, but the last few years have seen an explosion of both approved and work-around versions of that support. You can currently work natively with ProRes in applications like Avid, Premiere and Resolve on a Windows machine, which is very useful for professional workflow. Where things break down is at the consumer level. If you are delivering a file to a client, there still isn’t an easy way to get a non-tech savvy Windows user who defaults to Windows Media Player to playback a ProRes file.

ProRes naming has sometimes been a little difficult, with “proxy” confusing some users since a lot of software can create “proxy” files, but in any format. You can use Premiere to make “proxies,” but they don’t have to be ProRes proxy; they could be in LT. 4444 is often difficult to say, so many

### **ProRes File Sizes**

422 Proxy 45 Mb/s

422 LT 102 Mb/s

422 147 Mb/s (often called prime)

422 HQ 220 Mb/s

4444 330 Mb/s (often called 4×4 or quatro)

4444 XQ 500 Mb/s



say four by four or quatro, with quatro being more common on the west coast. Plain old “prores” without any modifiers can be confusing since you might say to someone, “can I have it in ProRes,” meaning plain prores, and they’ll ask, “what flavor,” and you say, “ProRes,” and comedy ensues. Thus most use the term “prime,” as in, “let’s use ProRes prime for that workflow,” to mean the middle-level codec.

While you might think “bigger is always better,” bigger files take up more storage, take longer to move around and are more taxing on the system to work with, so you often choose the flavor that works for your workflow. ProRes proxy is rarely used anymore since the image quality is visibly degraded, and storage is less expensive than it used to be. Most projects use LT for “offline” work like editing, then a bigger flavor for finishing and VFX.

But if your camera doesn’t shoot enough data, it’s likely not worth going to a huge format like XQ since those files are large, and the extra data rate isn’t going to magically create quality that isn’t there in the source file. XQ is really for cameras that are capable of shooting high bit depths natively (like an Alexa, for instance). If your camera shot in 10bit 4:2:2 video, transcoding it to 4444XQ doesn’t magically add extra quality. Typically most productions render out to plain old 4444 for their final master file.

While originally built primarily for the .mov video wrapper, Apple ProRes is officially supported in the .mxf wrapper, which is widely used in broadcast applications and has some features that can make it more useful, including better implementation of timecode.

## Avid DNx

While Apple ProRes has become far more ubiquitous in post-production workflows, **Avid DNx** as a codec family actually launched first and has a few key features that make it more useful in a few key situations that should keep it on your radar.

DNx, like ProRes, is actually a family of codecs available at a variety of data rates and encoding for a variety of workflows. You can shoot straight to it in cameras like the Alexa, the RED lineup and more, and you can edit it and deliver it to networks.



DNx is most comfortable in the .mxf (media exchange format) wrapper, which is a robust format with a lot of professional features, though you can also write DNx into a .mov wrapper if, for some reason, your workflow requires that.

DNx is widely supported on both PC and Mac machines, meaning it can be a great codec to use if your facility has mixed platforms or you are collaborating with others working in a variety of different formats. This has been its greatest strength. However, it's not particularly

easy to install for the less technically savvy, so it again doesn't make a great format for delivering cuts to clients since it requires installing a professional application for support.

DNx originally launched as DNxHD in a series of flavors that baked their data rate right into the name of the codec; you had DNx36 for editing and DNx175 for masters. DNx36 was a 36 Mb/s codec, designed to work well with 1080p 23.98 footage, and somewhat equivalent to ProRes proxy though ever so slightly smaller.

The problems came when formats started exploding. When the vast majority of work was 1080p, having the codec name and implementation built around a data rate made sense. While a 1080p 23.98 codec might look fine at 36Mb/s (not great, but fine), a 4k 60fps file would look terrible at that format. The larger resolution and framerate need more data to still look good.

Users, of course, could and should use a different flavor of DNx for 4k files than 1080p files, but many users were accustomed to using 36 for their edit. Avid revised the rollout of the DNx codecs to a new platform, which you would commonly work from today as the **DNxHR format of codecs**. These shift their data rate depending on the resolution and framerate of the source footage, making them work more like how ProRes works and more how users expect them to work.

So, to compare with ProRes, the new DNxHR HQ format at 1080p 29.97 is 25.99 MB/s, while ProRes HQ is 220 Mb/s. That might seem like a big difference until you note that the Avid number



is MB, while the Mac number is Mb. MB is megabyte, and Mb is megabit. Putting them both in Mb, the DNxHR HQ is around 207 Mb/s, roughly equivalent to ProRes HQ.

DNxHR is a very common format in all houses running **Avid Media composer**, and its cross-platform compatibility makes it useful when dealing with moving from PC and Mac.

## H.264/H.265

These codecs are consumer-facing codecs that post-production professionals need to be aware of and work with on a daily basis, but have some huge drawbacks. It's important to understand and master to keep your workflow performing optimally. The main place you will want to actively use these codecs is in delivery, especially on web platforms. You aren't going to send an H.265 file to Netflix or HBO, but if delivering to IG, YT, Vimeo or a work-in-progress review platform, you are going to be using H.265 all day long to get a file that is both small enough to quickly upload but still looks good enough to share with the world.

H.264 has been around longer, and H.265 is an update of the technology that offers similar image quality with about half the file size. You'll sometimes see H.265 referred to as "HEVC," an abbreviation for "high efficient video codec."

H.264 is far more ubiquitous since it's been around longer and is easier to license. H.265 has relatively high license pricing, so while you'll find it supported natively in all the major editing platforms and all the major web delivery platforms (Youtube, Instagram, Vimeo, etc.) you're still going to run into the occasional weird platform that doesn't fully support H.265. If you are having trouble delivering to a strange client portal or obscure streaming software the client uses, the issue might be that that platform doesn't support H.265, and you should try making an H.264 instead.

These codecs are built around Long GOP technology, in which a group of frames is compressed together to save space in the file. This is a wonderful technology for when you are viewing something linearly forward in time, making this a great codec for delivering video over the web. However, Long GOP can be very awkward in the editing room, since it requires your video software to recreate individual frames by looking at the group of frames. If you are scrubbing around, it can be laggy,



and if you cut in the middle of a GOP group, the software has to recreate the missing picture information by holding those other frames in memory.

While some software platforms like to market that they can natively cut in H.264 or H.265, it is highly recommended you transcode footage into an editing codec like ProRes or DNxHR for an easier post-workflow experience. Running an overnight dailies render will make the rest of your post pipeline so much easier.

H.264/H.265 can also be used for capture, though that is generally something to be avoided if you can, as the image quality drawbacks can be very frustrating. Even cameras like the iPhone now shoot straight to ProRes, so the arguments for capturing into H.265 are less pressing than they were a few years ago. If you have to shoot to H.265, choose the highest bitrate you can and choose “All-I” if it is an option, which will make every frame an “I” frame instead of compressing groups of frames together for compression.

H.264/H.265 formats can support whatever data rate you want to encode; generally, your encoder will let you change the data rate of your compression when you make the file. It is highly encouraged you test your specific encoder and project at a variety of data rates to find one that works for your projects and deliveries. For some reason, most encoders (like Resolve, Adobe Media Encoder, Compressor, etc.) have relatively small data rates as their “high-quality” file size. If you aren’t happy with how your images look when compressing to these codecs, try testing at higher quality file sizes to see when you start to like the image.

## **DPX and Similar Image Sequences**

These aren’t technically “codecs,” but you should be aware of image sequences as a tool in post production. DPX is the most common image sequence (and the one we’ll focus most on here), though EXR and Cineon are other common image sequences.

Image sequences are literally just a folder with a series of still images, numbered sequentially, saved into it. That’s pretty much the totality of it. Software dealing with image sequences (like Resolve and most VFX platforms like Nuke) will look at that folder full of still images and see it as



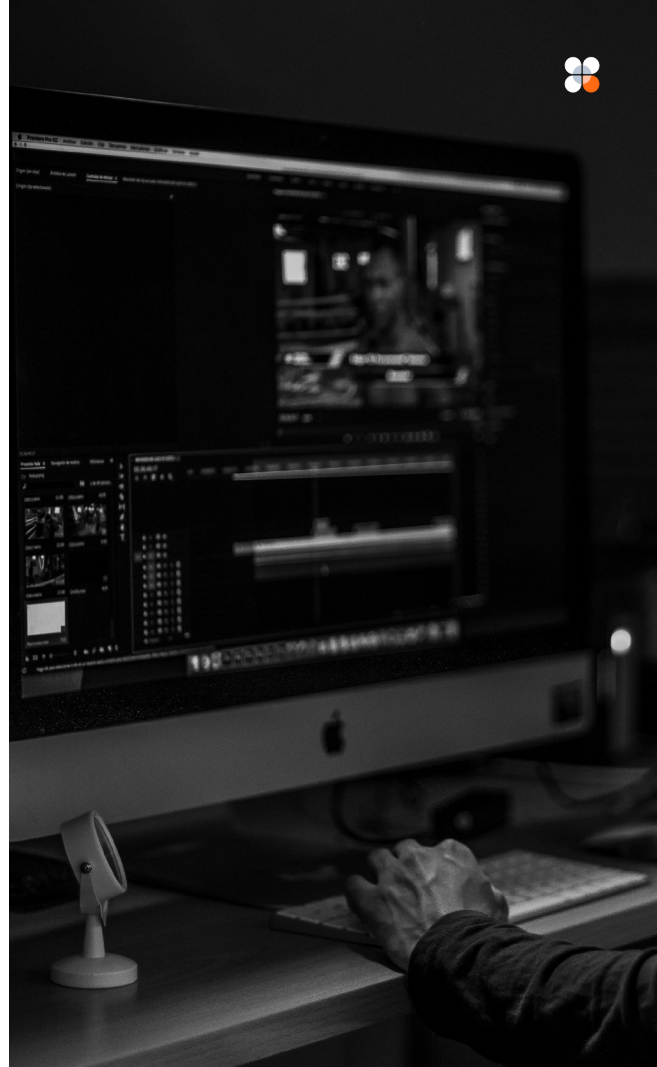
a single image file that you can manipulate just like a video file. Image sequences are incredibly popular in the VFX world for a few main reasons. First off, they are easier to move around. If you have a 40GB file, and your file transfer crashes halfway through your upload to the web, you have to start over from the beginning. Not with an image sequence; you can just start over with the last frame copied.

Beyond that, if you have a render and 90% of the shoot looks perfect, but you need to fix part of something that looked off at the end, with an image sequence, you only need to re-render those final frames. With a video file, you need to re-render the whole shot. With render times sometimes being exceptionally long in the VFX world, this is a huge time savings.

VFX artists aren't going to give up image sequences any time soon for those benefits. If you are being asked to interface with a VFX artist and they are asking for an image sequence, you can and should ask them for a spec sheet on what they are looking for. Then you can deliver it with a tool like Resolve, which has full support for multiple image sequence formats built-in natively.

## RAW Formats

RAW capture formats aren't technically "codecs" since RAW happens to the video signal before it gets wrapped into a codec, but it's good to have a handle on the most common RAW formats and how they might present in your workflow. RAW isn't "video" in that it can't be played easily by a video player. RAW formats take the RAW camera signal from the sensor and compress it into a file before it gets processed into video and the menu settings like ISO, white balance, etc. get applied. This makes more processing required in post (since your editing station has to do all that work that the camera used to have to do), but they offer the benefit of more flexibility in post. If you want to change your mind about white balance or ISO, you can do it in the edit or color suite, which is helpful, especially if the settings were accidentally wrong on camera.







There are two major categories of RAW forms, open RAW formats and proprietary or closed RAW formats. Open RAW formats are designed for many different platforms to capture to or work with. Proprietary formats created by a camera company are often only supported by that one company, with varying support from post-production software. The major proprietary RAW formats are now natively supported in all the major software platforms, but if you run into a more obscure format, you'll often need to download software support from their website.

## Open RAW Formats

Before discussing the two open RAW formats, one issue needs to be discussed: the red RAW patent. **RED** introduced the RED ONE camera at NAB 2006, and it was a working model that captured compressed motion picture RAW footage into an internal recorder. They applied for and received a patent on that technology. Both Sony and Apple have challenged the patent in court, and even with their legal resources, both lost. The RED patent stands, and as far as we know (it's not always public), the other internal RAW proprietary formats are paying some sort of license fee to RED.

This led to two different strategies for how to implement a RAW video format that was accessible to all users without paying for the RED license fee, since whatever that fee is, it doesn't make sense within a mass-market-facing, consumer-focused video market.

## ProRes RAW

The first "open" RAW to market is ProRes Raw, a format co-developed by Apple and **Atomos**, which makes an external monitor/recorder platform. That is their method for getting around the limitations of the RED patent; ProRes RAW is something you record to an external recorder.

Currently, ProRes raw has native support in **Final Cut Pro**, Avid Media Composer and Adobe Premiere, but not in Blackmagic Resolve. There are no announced plans to bring it to Resolve. If you are planning on doing your final color grade in Resolve, ProRes Raw isn't going to be the format for you.

Interestingly, DJI has implemented ProRes RAW into some of their drones, since, technically, with a drone, the camera is actually dangling underneath the drone, and the recorder is up in the body of the drone, which is enough to make it an "external" recording. ProRes RAW was briefly available in



the DJI Ronin 4D but then disappeared, and the suspicion is that they weren't able to argue that it counted as "external" on that camera.

ProRes Raw is available in two data rates and offers substantial image quality benefits for shots that weren't exposed under proper settings, such as with the wrong white balance. However, for shots properly exposed and with correct menu settings, the benefits are not large, though they are there.

## Blackmagic RAW

**Blackmagic** had an interesting challenge in building their RAW codec; they make external recorders, and editing software, but they also make cameras, and they wanted their RAW to work inside a Blackmagic Camera. However, they sell a lot of cameras, and outsiders suspect they wanted to avoid a RED license fee considering the sheer volume of units they ship. To get around it they designed the Blackmagic RAW format which is partially debayered. It's not a full debayer, which means there are still some of the benefits of RAW (you can change ISO and white balance in post), but also avoid the patent limitations of trying to record full RAW.



Blackmagic RAW is an open format, supported by all the major NLEs, and available in Blackmagic cameras and recorders, supported by several other camera manufacturers, including Fujifilm.

Blackmagic RAW is available in multiple bitrates but, interestingly, is also available in a variable bitrate format. This changes the bitrate based on the content of the shot so that a very static shot (an interview, for instance, where only the mouth of the speaker moves) can be a smaller file than a handheld shot out the moving window of a car in a busy street where there is a ton of movement. Variable bitrate shooting makes some users nervous, but some doc shooters have taken to the format for data rate savings in predictable environments.



## Proprietary or Closed RAW Formats

We can't cover every proprietary raw format here as there are too many, but there are two we need to discuss a bit. If you run into another format, you should go to the camera manufacturers website for more info.

### .r3d RED RAW

RED RAW, recorded in the .r3d wrapper, is the format that started the RAW video revolution. RED RAW takes the RAW camera data, applies a JPEG2000 compression to it, and wraps it up in a file that you can then process to your heart's content in post production.

RED RAW is currently supported basically everywhere. It's been around 15 years, and all the major software platforms have fully integrated it's technology into their systems.

RED files are surprisingly small, considering the quality of their imagery; because of the nature of their compression, many users are surprised to discover that the files can get larger when transcoding to an edit codec like ProRes, depending on your editing resolution and codec choice.

### .ARRIRAW

ARRIRAW is the other major file format to discuss, not just because **ARRI** is at the top of the industry but also because the files are just huge. For a long time, you needed to rent an additional external recorder from Codex to record ARRIRAW (to avoid the patent, most assume), though you can now record ARRIRAW internally to an ARRI camera. Either ARRI figured out some very tricky way to argue their internal recorder is actually external, or they are paying the license fee to RED.

The thing to know about ARRIRAW files is that they are big. If you are bidding your first ARRIRAW job after years of RED RAW, know that it's going to require more hardware resources than you are used to. These are massive files. Transcode them immediately to an edit-friendly codec, then deal with them again only at the end for color grading on a powerful machine. Their saving grace is that ARRI cameras only shoot up to 4k; an 8k or 12k ARRIRAW file would be a monster.



## Oddball Formats

While this guide can't go into detail on every possible format and codec you might encounter, we want to offer some general advice when a shot lands in your lap that might not immediately make sense to you.

Your first tip is to use the “get info” command, either in the finder, in QuickTime player or in an app like “Screen,” to get a better sense of what is going on with the codec. A quick Google search will often turn up more info on the codec, and it is usually available for download and install on your system for playback. If “get info” isn't helping, there is a great app called “MedialInfo” that might offer more information.

There are some limits to this (Apple ProRes still has issues with running on a Windows machine in certain players depending on the install), but for the most part, pretty much every codec you need is possible to download, and that will often lead to your software being able to decode the video.

If you run into a truly unplayable codec, there is a player you should know about called VLC. It's a video playback software that is often a “swiss army knife” in post when you've been given a strange video format to deal with. Maybe you are working on a documentary with a lot of archival home-video footage in an obscure format that didn't take off commercially. Or you are working on a film with footage coming in from primary sources from a variety of archives. Or you have a shot that has gremlins and just doesn't want to play. VLC is often the tool that will finally get that video open, and then you can export from VLC into a more traditional codec and format that will let you play it in your editing platform of choice.



# Create Amazing Together.

Our products help the biggest names in entertainment and advertising securely manage, present, and collaborate on their highest-value projects.

[LEARN MORE](#)