



Adobe® Premiere™ Technical Guides

An explanation of drop-frame vs. non-drop-frame timecode

Mathematics of 29.97 video

Frame rate is the rate at which video plays back frames. Black and white video ran at a true 30 frames per second (fps). When the color portion of the signal was added, video engineers were forced—for various technical reasons related to the physical circuits—to slow the rate down to 29.97 fps. This slight slowdown of video playback leads to distortions in the measurement of video vs. real time. Video is measured in indivisible units called frames. Real time is measured in hours, minutes, and seconds. Unfortunately, a second is not evenly divisible by 29.97 fps. Let's look at the mathematical relationships involved here:

A frame rate of 29.97 fps is 99.9% as fast as 30 fps. In other words, it is 0.1% (or one-thousandth) slower:

$$29.97 \text{ fps} / 30 \text{ fps} = .999 \text{ (or 99.9\%)}$$

$$100 - .999 = 0.1\% \text{ slower}$$

Conversely, a frame rate of 30 fps is 0.1% (or one-thousandth) faster than 29.97:

$$30 \text{ fps} / 29.97 \text{ fps} = 1.001 \text{ (or 100.1\%)}$$

(The actual value is 1.001,001,001, ..., 001 repeating infinitely. 1.001 is enough precision for our calculation, given that the next significant digit is the one-millionths place. No video program is long enough that the stray millionths of a second per hour will add up enough to throw the frame count off again.)

One hour's worth of "true 30 fps" video contains exactly 108,000 frames:

$$(30 \text{ frames/sec}) * (3600 \text{ sec/hour}) = 108,000 \text{ frames}$$

However, if you play back 108,000 frames at 29.97 fps, it will take longer than 1 hour to play:

$$(108,000 \text{ frames}) / (29.97 \text{ frames/sec}) = 3,603.6 \text{ seconds} = 1 \text{ hour and } 3.6 \text{ seconds}$$

(Actual value is 3,603.603,603, ..., 603 repeating infinitely. Again, 3,603.6 is sufficient for video timecode, given that the next loss of precision is three one-thousandths of a second per hour. You would have to make a video over 11 hours long before you were off again by a single frame.)

This is notated in timecode as 01:00:03:18. Thus, after an hour it is 108 frames too long. Once again, we see the relationship of 108 frames out of 108,000, or one-thousandth.



Now let's apply that discrepancy to 1 minute of video. One minute, or 60 seconds, of 30 fps video contains 1800 frames. One-thousandth of that is 1.8. Therefore, by the end of 1 minute you are off by 1.8 frames.

Remember, however, that frames are indivisible; you cannot adjust by a fraction of a frame. You cannot adjust by 1.8 frames per minute, but you can adjust by 18 full frames per 10 minutes.

Because 10 minutes is not evenly divisible by 18 frames, we use drop-frame timecode and drop two frame numbers every minute; by the ninth minute, you have dropped all 18 frame numbers. No frames need to be dropped the tenth minute. That is how drop-frame timecode works. When you use drop-frame timecode, Premiere 5.x renumbers the first two frames of every minute, except for every tenth minute.

NTSC and the drop-frame numbering system

There are three fundamentally important things to remember about NTSC and drop-frame timecode:

- NTSC video **always** runs at 29.97 frames/second.
- 29.97 video can be notated in either drop-frame or non-drop-frame format.
- Drop-frame timecode only drops numbers that refer to the frames, and not the actual frames.

We will examine the ramifications of these rules below.

NTSC video always runs at 29.97 frames/second

Unlike "true 30 fps" video, an hour's worth of NTSC video does not have 108,000 frames in it. It has 99.9% as many frames, or 107,892 frames, as described earlier. Again, at the rate of 1.8 less per minute, an hour of NTSC video has 108 frames less than an hour of "true 30 fps" video:

$$108,000 * 99.9\% = 107,892 \text{ frames in an hour of NTSC video}$$

$$108,000 - 107,892 = 108 \text{ frames difference}$$

If we were to sequentially number each of these frames using the SMPTE Timecode format, the last frame of the video would be numbered 00:59:26:12:

$$108 \text{ frames} = 00:00:03:18 \text{ in timecode format}$$

$$01:00:00:00 - 00:00:03:18 = 00:59:26:12$$

That is 3 seconds and 18 frames shorter than an hour-long video. Drop-frame timecode is a SMPTE standard that maintains time accuracy by eliminating the fractional difference between the 29.97 fps frame rate and the 30 fps numbering. When you use drop-frame timecode, Premiere 5.x adjusts the frame numbering so that an hour-long video has its last frame labeled 01:00:00:00.



Timecode measures time in Hours:Minutes:Seconds:Fractions-of-seconds called frames. However, in NTSC video, a frame is not an even fraction of a second! Thus, NTSC timecode is always subtly off from real time—by exactly 1.8 frames per minute. Drop-frame timecode numbering attempts to adjust for this discrepancy by dropping two numbers in the numbering sequence, once every minute except for every tenth minute (see the preceding section, Mathematics of 29.97 video, for details). The numbers that are dropped are frames 00 and 01 of each minute; thus, drop-frame numbering across the minute boundary looks like this:

..., 00:00:59:27, 00:00:59:28, 00:00:59:29, 00:01:00:02, 00:01:00:03, ...

Note, however, that you are off by only 1.8 frames per minute. If you adjust by two full frames every minute, you are still off by a little. Let's go through a sequence of minutes, to see how far off we are each minute, and where each adjustment leaves us:

Minute	Start Position	Frames Lost	Drop Frame	Adjusted Position
01		1.8 lost this minute	drop 2 to correct	0.2 ahead
02	0.2 ahead	1.8 lost this minute	drop 2 to correct	0.4 ahead
03	0.4 ahead	1.8 lost this minute	drop 2 to correct	0.6 ahead
04	0.6 ahead	1.8 lost this minute	drop 2 to correct	0.8 ahead
05	0.8 ahead	1.8 lost this minute	drop 2 to correct	1.0 ahead
06	1.0 ahead	1.8 lost this minute	drop 2 to correct	1.2 ahead
07	1.2 ahead	1.8 lost this minute	drop 2 to correct	1.4 ahead
08	1.4 ahead	1.8 lost this minute	drop 2 to correct	1.6 ahead
09	1.6 ahead	1.8 lost this minute	drop 2 to correct	1.8 ahead
10	1.8 ahead	1.8 lost this minute	drop 0	

Thus, 00:10:00:00 in drop-frame is the same as 00:10:00:00 in real time! Also, 10 minutes of NTSC video contains an exact number of frames (17,982 frames), so every tenth minute ends on an exact frame boundary. This is how we can get exactly 1 hour of video to read as exactly 1 hour of timecode.

29.97 Video can be notated in either drop-frame or non-drop-frame format

You can notate 29.97 video using drop-frame or non-drop-frame format. The difference between the two is that with drop-frame format the frame address is periodically adjusted (once every minute) so that it exactly matches real time (at the 10 minute mark), while with non-drop-frame format the frame address is never adjusted and gets progressively further away from real time.



At the end of an hour-long video, the frame address for drop-frame format will be 01:00:00:00, while the frame address for non-drop-frame format will be 108 frames lower (remember, 108 frames out of 108,000, or 0.1%) at 00:59:56:12.

Conversely, at the point where the frame address for non-drop-frame format reads 01:00:00:00, the frame address for drop-frame format would be 01:00:03:18. Remember, this is longer than 1 hour of real time: 3.6 seconds out of 3600, or 0.1%.

Either numbering system could have been used for this theoretical video program. No matter which timecode format you use, the frame rate—29.97 fps—would be the same, and the total duration of the program—in real time—would be the same. The only difference is which address code gets stamped on what frame number.

Drop-frame timecode only drops numbers that refer to the frames, and not the actual frames

This is nothing complicated; just remember to keep your terminology straight. Much analog video equipment uses drop-frame SMPTE timecode. Just imagine if analog video were to drop the actual frames! First, it would visually disturbing to literally drop two frames every minute. Second, and more importantly, analog video equipment is governed by a certain amount of tape moving past the heads at a certain speed. Even if the equipment didn't display two frames, there is no way for the physical mechanism to make up for the lost time. This is not the same as with digital video, where a capture or playback device will drop frames because it simply can't keep up with the amount of data being streamed through it.

Also, when we talk about being 1.8 frames ahead or behind, we are referring to the frame numbering scheme being ahead of real time. It does not refer to the video track being ahead or behind the audio track; audio that drifts away from its video is a different issue,

In summary, "dropped frames" refers to a playback or capture issue related to data rates and hardware capabilities; drop-frame timecode refers to a frame-numbering convention.

