

## History

<b><u>Original author(s)</u></b>	<u>Fabrice Bellard</u>
<b><u>Developer(s)</u></b>	FFmpeg team
<b><u>Initial release</u></b>	December 20, 2000 <sup>[1]</sup>
<b><u>Stable release</u></b>	4.3.1 (July 11, 2020 <sup>[2]</sup> ) <span>[±]</span> ( <a href="https://en.wikipedia.org/w/index.php?title=Template:Latest_stable_software_release/FFmpeg&amp;action=edit">https://en.wikipedia.org/w/index.php?title=Template:Latest_stable_software_release/FFmpeg&amp;action=edit</a> )
<b><u>Preview release</u></b>	Git <span>[±]</span> ( <a href="https://en.wikipedia.org/w/index.php?title=Template:Latest_preview_software_release/FFmpeg&amp;action=edit">https://en.wikipedia.org/w/index.php?title=Template:Latest_preview_software_release/FFmpeg&amp;action=edit</a> )
<b><u>Repository</u></b>	<a href="https://git.ffmpeg.org/ffmpeg.git">git.ffmpeg.org/ffmpeg.git</a> ( <a href="https://git.ffmpeg.org/ffmpeg.git">https://git.ffmpeg.org/ffmpeg.git</a> )
<b><u>Written in</u></b>	<u>C</u> and <u>Assembly</u> <sup>[3]</sup>
<b><u>Operating system</u></b>	Various, including <u>Windows</u> , <u>macOS</u> , and <u>Linux</u> (executable programs are only available from third parties, as the project only distributes <u>source code</u> ) <sup>[4][5]</sup>
<b><u>Platform</u></b>	x86, ARM, PowerPC, MIPS, DEC Alpha, <u>Blackfin</u> , <u>AVR32</u> , <u>SH-4</u> , and <u>SPARC</u> ; may be <u>compiled for other desktop computers</u>
<b><u>Type</u></b>	<u>Multimedia framework</u>
<b><u>License</u></b>	LGPL 2.1+, GPL 2+ Unredistributable if compiled with any software with a license incompatible with the <u>GPL</u> <sup>[6]</sup>
<b><u>Website</u></b>	<a href="https://ffmpeg.org">ffmpeg.org</a> ( <a href="https://ffmpeg.org">https://ffmpeg.org</a> )

The project was started by [Fabrice Bellard](#)<sup>[10]</sup> (using the pseudonym "G rard Lantau") in 2000, and was led by Michael Niedermayer from 2004 until 2015.<sup>[11]</sup> Some FFmpeg developers were also part of the [MPlayer](#) project.

On January 10, 2014, two Google employees announced that over 1000 bugs had been fixed in FFmpeg during the previous two years by means of [fuzz testing](#).<sup>[12]</sup>

In January 2018, the *ffserver* command-line program – a long-time component of FFmpeg – was removed.<sup>[13]</sup> The developers had previously deprecated the program citing high maintenance efforts due to its use of internal [application programming interfaces](#).<sup>[14]</sup>

The project publishes a new release every three months on average. While release versions are available from the website for download, FFmpeg developers recommend that users compile the software from source using the latest build from their [source code](#) [Git version control system](#).<sup>[15]</sup>

## Codec history

Two [video coding formats](#) with corresponding codecs and one [container format](#) have been created within the FFmpeg project so far. The two video codecs are the lossless [FFV1](#), and the lossless and lossy [Snow](#) codec. Development of [Snow](#) has stalled, while its bit-stream format has not been finalized yet, making it experimental since 2011. The multimedia container format called [NUT](#) is no longer being actively developed, but still maintained.<sup>[16]</sup>

In summer 2010, Fiona Glaser, Ronald Bultje, and David Conrad of the FFmpeg Team announced the [ffvp8](#) decoder. Through testing, they determined that [ffvp8](#) was faster than Google's own [libvpx](#) decoder.<sup>[17][18]</sup> Starting with version 0.6, FFmpeg also supported [WebM](#) and [VP8](#).<sup>[19]</sup>

In October 2013, a native [VP9](#)<sup>[20]</sup> and the [OpenHEVC](#) decoder, an open source [High Efficiency Video Coding](#) (HEVC) decoder, were added to FFmpeg.<sup>[21]</sup> In 2016 the native [AAC](#) encoder was considered stable, removing support for the two external AAC encoders from [VisualOn](#) and [FAAC](#). FFmpeg 3.0 (nicknamed "*Einstein*") retained build support for the [Fraunhofer FDK AAC](#) encoder.<sup>[22]</sup> Since version 3.4 "*Cantor*" FFmpeg supported the [FITS](#) image format.<sup>[23]</sup> Since November 2018 in version 4.1 "*al-Khwarizmi*" [AV1](#) can be muxed in [MP4](#) and [Matroska](#) incl. [WebM](#).<sup>[24][25]</sup>

## Forks

On March 13, 2011, a group of FFmpeg developers decided to fork the project under the name "[Libav](#)".<sup>[26][27][28]</sup> The event was related to an issue in project management, in which developers disagreed with the leadership of FFmpeg.<sup>[29][30][31]</sup>

## Components

---

### Command line tools

- *ffmpeg* is a command-line tool that converts audio or video formats. It can also capture and encode in real-time from various hardware and software sources<sup>[32]</sup> such as a TV capture card.
- *ffplay* is a simple media player utilizing [SDL](#) and the FFmpeg libraries.
- *ffprobe* is a command-line tool to display media information (text, [CSV](#), [XML](#), [JSON](#)), see also [Mediainfo](#).

### Libraries

- *libswresample* is a library containing audio [resampling](#) routines.
- *libavresample* is a library containing audio resampling routines from the [Libav](#) project, similar to *libswresample* from *ffmpeg*.
- *libavcodec* is a library containing all of the native FFmpeg audio/video encoders and decoders. Most codecs were developed from scratch to ensure best performance and high code reusability.
- *libavformat* ([Lavf](#))<sup>[7]</sup> is a library containing demuxers and muxers for audio/video container formats.
- *libavutil* is a helper library containing routines common to different parts of FFmpeg. This library includes hash functions ([Adler-32](#), [CRC](#), [MD5](#), [RIPEMD](#), [SHA-1](#), [SHA-2](#), [MurmurHash3](#), [HMAC MD-5](#), [HMAC SHA-1](#) and [HMAC SHA-2](#)), ciphers ([DES](#), [RC4](#), [AES](#), [AES-CTR](#), [TEA](#), [XTEA](#), [Blowfish](#), [CAST-128](#), [Twofish](#) and [Camellia](#)), [LZO](#) decompressor and [Base64](#) encoder/decoder.
- *libpostproc* is a library containing older h263 based [video postprocessing](#) routines.
- *libswscale* is a library containing video [image scaling](#) and [colorspace/pixelformat](#) conversion routines.
- *libavfilter* is the substitute for [vhook](#) which allows the video/audio to be modified or examined between the decoder and the encoder. Filters have been ported from many projects including [MPlayer](#) and [avisynth](#).

## Supported hardware

---

### CPUs

FFmpeg encompasses software implementations of video and audio compressing and decompressing algorithms. These can be compiled and run on diverse instruction sets.

Many widespread instruction sets are supported by FFmpeg, including x86 (IA-32 and x86-64), PPC (PowerPC), ARM, DEC Alpha, SPARC, and MIPS.<sup>[33]</sup>

### Special purpose hardware

Various application-specific integrated circuits (ASICs) related to video and audio compression and decompression exist. Such ASICs can perform the computation for audio/video decompression or compression partly or fully to offload these from the host CPU. To make use of such an ASIC, instead of a complete implementation of some algorithm, only the API is required. There are numerous ASICs and APIs available, of which several are supported by FFmpeg.<sup>[34]</sup>

Firm	ASIC	purpose	supported by FFmpeg	Details
AMD	<u>UVD</u>	decoding	✓	via <u>VDP AU API</u> and <u>VAAPI</u>
	<u>VCE</u>	encoding	✓	via <u>VAAPI</u> , considered experimental <sup>[35]</sup>
Amlogic	<u>Amlogic Video Engine</u>	decoding	?	
BlackMagic	<u>DeckLink</u>	encoding/decoding	✓	real-time ingest and playout
Broadcom	<u>Crystal HD</u>	decoding	✓	
Intel	<u>Intel Clear Video</u>	decoding	✓	
	<u>Intel Quick Sync Video</u>	encoding/decoding	✓	
Nvidia	<u>PureVideo / NVDEC</u>	decoding	✓	via the <u>VDP AU API</u> as of FFmpeg v1.2 (deprecated) via <u>CUVID API</u> as of FFmpeg v3.1 <sup>[36]</sup>
	<u>NVENC</u>	encoding	✓	as of FFmpeg v2.6

### Use with the FFmpeg Utility

Internal hardware acceleration decoding is enabled through the `-hwaccel` option. It starts decoding normally, but if a decodable stream is detected in hardware, then the decoder designates all significant processing to that hardware, thus accelerating the decoding process. Whereas if no decodable streams are detected (as happens on an unsupported codec or profile), hardware acceleration will be skipped and it will still be decoded in software. `-hwaccel_device` option is applied when the hardware requires a particular device to function especially there are several graphic cards are available.

## Supported codecs and formats

### Image formats

FFmpeg supports many common and some uncommon image formats.

The **PGMYUV** image format is a homebrewn variant of the binary (P5) PGM Netpbm format. FFmpeg also supports 16-bit depths of the PGM and PPM formats, and the binary (P7) PAM format with or without alpha channel, depth 8 bit or 16 bit for `pix_fmts` *monob*, *gray*, *gray16be*, *rgb24*, *rgb48be*, *ya8*, *rgba*, *rgb64be*.

### Supported formats

In addition to FFV1 and Snow formats, which were created and developed from within FFmpeg, the project also supports the following formats:

Group	Format type	Format name
<u>ISO/IEC/ITU-T</u>	Video	<u>MPEG-1 Part 2</u> , <u>H.261 (Px64)</u> , <sup>[37]</sup> <u>H.262/MPEG-2 Part 2</u> , <u>H.263</u> , <sup>[37]</sup> <u>MPEG-4 Part 2</u> , <u>H.264/MPEG-4 AVC</u> , <u>HEVC/H.265</u> <sup>[21]</sup> ( <u>MPEG-H Part 2</u> ), <u>MPEG-4 VCB</u> (a.k.a. <u>VP8</u> ), <u>Motion JPEG</u> , <u>IEC DV video</u> and <u>CD+G</u>
	Audio	<u>MP1</u> , <u>MP2</u> , <u>MP3</u> , <u>AAC</u> , <u>HE-AAC</u> , <u>MPEG-4 ALS</u> , <u>G.711 μ-law</u> , <u>G.711 A-law</u> , <u>G.721</u> (a.k.a. <u>G.726 32k</u> ), <u>G.722</u> , <u>G.722.2</u> (a.k.a. <u>AMR-WB</u> ), <u>G.723</u> (a.k.a. <u>G.726 24k</u> and <u>40k</u> ), <u>G.723.1</u> , <u>G.726</u> , <u>G.729</u> , <u>G.729D</u> , <u>IEC DV audio</u> and <u>Direct Stream Transfer</u>
	Subtitle	<u>MPEG-4 Timed Text</u> (a.k.a. <u>3GPP Timed Text</u> )
	Image	<u>JPEG</u> , <u>Lossless JPEG</u> , <u>JPEG-LS</u> , <u>JPEG 2000</u> , <u>PNG</u> , <u>CCITT G3</u> and <u>CCITT G4</u>
<u>Alliance for Open Media</u>	Video	<u>AV1</u> <sup>[38]</sup>
<u>EIA</u>	Subtitle	<u>EIA-608</u>
<u>CEA</u>	Subtitle	<u>CEA-708</u>
<u>SMPTE</u>	Video	<u>SMPTE 314M</u> (a.k.a. <u>DVCAM</u> and <u>DVCPRO</u> ), <u>SMPTE 370M</u> (a.k.a. <u>DVCPRO HD</u> ), <u>VC-1</u> (a.k.a. <u>WMV3</u> ), <u>VC-2</u> (a.k.a. <u>Dirac Pro</u> ), <u>VC-3</u> (a.k.a. <u>AVID DNxHD</u> )
	Audio	<u>SMPTE 302M</u>
	Audio	<u>Full Rate (GSM 06.10)</u> , <u>AC-3 (Dolby Digital)</u> , <u>Enhanced AC-3 (Dolby Digital Plus)</u> and <u>DTS Coherent Acoustics</u> (a.k.a. <u>DTS</u> or <u>DCA</u> )
	Image	<u>DPX</u>
<u>ATSC/ETSI/DVB</u>	Subtitle	<u>DVB Subtitling (ETSI 300 743)</u>
<u>DVD Forum/Dolby</u>	Audio	<u>MLP / Dolby TrueHD</u>
	Subtitle	<u>DVD-Video subtitles</u>
<u>DTS, Inc/QDesign</u>	Audio	<u>DTS Coherent Acoustics</u> (a.k.a. <u>DTS</u> or <u>DCA</u> ), <u>DTS Extended Surround</u> (a.k.a. <u>DTS-ES</u> ), <u>DTS 96/24</u> , <u>DTS-HD High Resolution Audio</u> , <u>DTS Express</u> (a.k.a. <u>DTS-HD LBR</u> ), <u>DTS-HD Master Audio</u> , <u>QDesign Music Codec 1</u> and <u>2</u>
<u>Blu-ray Disc Association</u>	Subtitle	<u>PGS (Presentation Graphics Stream)</u>
<u>3GPP</u>	Audio	<u>AMR-NB</u> , <u>AMR-WB</u> (a.k.a. <u>G.722.2</u> )
<u>3GPP2</u>	Audio	<u>QCELP-8</u> (a.k.a. <u>SmartRate</u> or <u>IS-96C</u> ), <u>QCELP-13</u> (a.k.a. <u>PureVoice</u> or <u>IS-733</u> ) and <u>Enhanced Variable Rate Codec (EVRC)</u> a.k.a. <u>IS-127</u>
<u>World Wide Web Consortium</u>	Video	<u>Animated GIF</u>
	Subtitle	<u>WebVTT</u>
	Image	<u>GIF</u> , and <u>SVG (via libsvg)</u>
<u>IETF</u>	Audio	<u>iLBC (via libilbc)</u> , <u>Opus</u> and <u>Comfort noise</u>
<u>International Voice Association</u>	Audio	<u>DSS-SP</u>
<u>SAC</u>	Video	<u>AVS video</u>
<u>Microsoft</u>	Video	<u>Microsoft RLE</u> , <u>Microsoft Video 1</u> , <u>Cinepak</u> , <u>Indeo (v2, v3, v4 and v5)</u> , <sup>[37]</sup> <u>Microsoft MPEG-4 v1, v2 and v3</u> , <u>Windows Media Video (WMV1, WMV2, WMV3/VC-1)</u> , <u>WMV Screen</u> and <u>Mimic codec</u>
	Audio	<u>Windows Media Audio (WMA1, WMA2, WMA Pro and WMA Lossless)</u> , <u>XMA (XMA1 and XMA2)</u> , <u>MS-GSM</u> and <u>MS-ADPCM</u>
	Subtitle	<u>SAMI</u>
	Image	<u>Windows Bitmap</u> , <u>WMV Image (WMV9 Image and WMV9 Image v2)</u> and <u>DirectDraw Surface</u>
<u>Interactive Multimedia Association</u>	Audio	<u>IMA ADPCM</u>
<u>Digital Video Interactive</u>	Video	<u>RTV 2.1 (Intel Indeo 2)</u>
	Audio	<u>DVI4 audio codec</u>
<u>RealNetworks</u>	Video	<u>RealVideo Fractal Codec</u> (a.k.a. <u>Iterated Systems ClearVideo</u> ), <u>1</u> , <u>2</u> , <u>3</u> and <u>4</u>
	Audio	<u>RealAudio v1 – v10</u>
	Subtitle	<u>RealText</u>
<u>Apple / Spruce Technologies</u>	Video	<u>Cinepak (Apple Compact Video)</u> , <u>ProRes</u> , <u>Sorenson 3 Codec</u> , <u>QuickTime Animation (Apple Animation)</u> , <u>QuickTime Graphics (Apple Graphics)</u> , <u>Apple Video</u> , <u>Apple Intermediate Codec</u> and <u>Pixlet</u>
	Audio	<u>ALAC</u>
	Image	<u>QuickDraw PICT</u>
	Subtitle	<u>Spruce subtitle (STL)</u>
<u>Adobe Flash Player (SWF)</u>	Video	<u>Screen video</u> , <u>Screen video 2</u> , <u>Sorenson Spark</u> and <u>VP6</u>
	Audio	<u>Adobe SWF ADPCM</u> and <u>Nellymoser Asao</u>
<u>Aldus / Adobe</u>	Image	<u>TIFF</u> , <u>PSD</u> , and <u>DNG</u>
<u>Xiph.Org</u>	Video	<u>Theora</u>

	Audio	<a href="#">Speex</a> (via <a href="#">libspeex</a> ), <a href="#">Vorbis</a> , <a href="#">Opus</a> and <a href="#">FLAC</a>
	Subtitle	<a href="#">Ogg Writ</a>
<a href="#">Sony</a>	Audio	<a href="#">Adaptive Transform Acoustic Coding</a> (ATRAC1, ATRAC3, ATRAC3Plus and ATRAC9) <sup>[37][39]</sup> and <a href="#">PSX ADPCM</a>
<a href="#">NTT</a>	Audio	<a href="#">TwinVQ</a>
<a href="#">On2</a> / <a href="#">GIPS</a> / <a href="#">Google</a>	Video	<a href="#">Duck TrueMotion 1</a> , <a href="#">Duck TrueMotion 2</a> , <a href="#">Duck TrueMotion 2.0 Real Time</a> , <a href="#">VP3</a> , <a href="#">VP4</a> , <a href="#">VP5</a> , <sup>[37]</sup> <a href="#">VP6</a> , <sup>[37]</sup> <a href="#">VP7</a> , <a href="#">VP8</a> , <a href="#">VP9</a> <sup>[20]</sup> and <a href="#">animated WebP</a>
	Audio	<a href="#">DK ADPCM Audio 3/4</a> , <a href="#">On2 AVC</a> and <a href="#">iLBC</a> (via <a href="#">libilbc</a> )
	Image	<a href="#">WebP</a>
<a href="#">RAD Game Tools</a>	Video	<a href="#">Smacker video</a> and <a href="#">Bink video</a>
	Audio	<a href="#">Bink audio</a>
<a href="#">CRI Middleware</a>	Audio	<a href="#">ADX ADPCM</a> , and <a href="#">HCA</a>
<a href="#">Nintendo</a> / <a href="#">NERD</a>	Video	<a href="#">Mobiclip video</a>
	Audio	<a href="#">GCADPCM</a> (a.k.a. <a href="#">ADPCM THP</a> ), <a href="#">FastAudio</a> , and <a href="#">ADPCM IMA MOFLEX</a>
<a href="#">DSP Group</a>	Audio	<a href="#">Truespeech</a>
<a href="#">Electronic Arts</a> / <a href="#">Criterion Games</a> / <a href="#">Black Box Games</a>	Video	<a href="#">RenderWare TXD</a> <sup>[40]</sup> , <a href="#">Madcow</a> , <a href="#">CMV</a> , <a href="#">TGV</a> , <a href="#">TGQ</a> , <a href="#">TQI</a> , <a href="#">Midivid VQ (MVDV)</a> , <a href="#">MidiVid 3.0 (MV30)</a> , and <a href="#">Midivid Archival (MVHA)</a>
	Audio	<a href="#">Electronic Arts ADPCM variants</a>
<a href="#">Netpbm</a>	Image	<a href="#">PBM</a> , <a href="#">PGM</a> , <a href="#">PPM</a> , <a href="#">PNM</a> , <a href="#">PAM</a> and <a href="#">PFM</a>
<a href="#">MIT/X Consortium/The Open Group</a>	Image	<a href="#">XBM</a> , <a href="#">XPM</a> and <a href="#">xwd</a>
<a href="#">HPE</a> / <a href="#">SGI</a> / <a href="#">Silicon Graphics</a>	Video	<a href="#">Silicon Graphics RLE 8-bit video</a> , <a href="#">Silicon Graphics MVC1/2</a>
	Image	<a href="#">Silicon Graphics Image</a>
<a href="#">Oracle/Sun Microsystems</a>	Image	<a href="#">Sun Raster</a>
<a href="#">IBM</a>	Video	<a href="#">IBM UltiMotion</a>
<a href="#">Avid Technology</a> / <a href="#">Truevision</a>	Video	<a href="#">Avid 1:1x</a> , <a href="#">Avid Meridien</a> , <a href="#">Avid DNxHD</a> and <a href="#">DNxHR</a>
	Image	<a href="#">Targa</a>
<a href="#">Autodesk</a> / <a href="#">Alias</a>	Video	<a href="#">Autodesk Animator Studio Codec</a> and <a href="#">FLIC</a>
	Image	<a href="#">Alias PIX</a>
<a href="#">Grass Valley</a> / <a href="#">Canopus</a>	Video	<a href="#">HQ</a> , <a href="#">HQA</a> , <a href="#">HQX</a> and <a href="#">Lossless</a>
<a href="#">NewTek</a>	Video	<a href="#">SpeedHQ</a>
<a href="#">Academy Software Foundation</a> / <a href="#">ILM</a>	Image	<a href="#">OpenEXR</a>
<a href="#">Mozilla Corporation</a>	Video	<a href="#">APNG</a>
<a href="#">Matrox</a>	Video	<a href="#">Matrox Uncompressed SD (M101)</a> / <a href="#">HD (M102)</a>
<a href="#">AMD/ATI</a>	Video	<a href="#">ATI VCR1/VCR2</a>
<a href="#">Asus</a>	Video	<a href="#">ASUS V1/V2 codec</a>
<a href="#">Commodore</a>	Video	<a href="#">CDXL codec</a>
<a href="#">Kodak</a>	Image	<a href="#">Photo CD</a>
<a href="#">Blackmagic Design</a> / <a href="#">Cintel</a>	Image	<a href="#">Cintel RAW</a>
<a href="#">Houghton Mifflin Harcourt</a> / <a href="#">The Learning Company</a> / <a href="#">ZSoft Corporation</a>	Image	<a href="#">PCX</a>
<a href="#">Bell Labs</a>	Image	<a href="#">X-Face</a>
<a href="#">Bluetooth Special Interest Group</a>	Audio	<a href="#">SBC</a> , and <a href="#">mSBC</a>
<a href="#">Qualcomm</a>	Audio	<a href="#">QCELP</a> , <a href="#">aptX</a> , and <a href="#">aptX HD</a>

## Muxers

Output formats (container formats and other ways of creating output streams) in FFmpeg are called "muxers". FFmpeg supports, among others, the following:

- [AIFF](#)
- [ASF](#)
- [AVI](#) and also input from [AviSynth](#)

- [BFI](#)<sup>[41]</sup>
- [CAF](#)
- [FLV](#)
- [GIF](#)
- [GXF](#), General eXchange Format, SMPTE 360M
- [HLS](#), HTTP Live Streaming
- [IFF](#)<sup>[42]</sup>
- [ISO base media file format](#) (including [QuickTime](#), [3GP](#) and [MP4](#))
- [Matroska](#) (including [WebM](#))
- [Maxis XA](#)<sup>[43]</sup>
- [MPEG-DASH](#)<sup>[44]</sup>
- [MPEG program stream](#)
- [MPEG transport stream](#) (including [AVCHD](#))
- [MXF](#), Material eXchange Format, SMPTE 377M
- [MSN Webcam stream](#)<sup>[45]</sup>
- [NUT](#)<sup>[16]</sup>
- [Ogg](#)
- [OMA](#)<sup>[46]</sup>
- [RL2](#)<sup>[47]</sup>
- [Segment](#), for creating segmented video streams
- [Smooth Streaming](#)
- [TXD](#)<sup>[40]</sup>
- [WTV](#)

## Pixel formats

FFmpeg supports many pixel formats.<sup>[48]</sup> Some of these formats are only supported as input formats. The command `ffmpeg -pix_fmts` provides a list of supported pixel formats.

Type	Color	Packed		Planar			Palette
		Without alpha	With <u>alpha</u>	Without alpha	With alpha	Chroma-interleaved	With alpha
<b>Monochrome</b>	<u>Binary</u> (1-bit monochrome)	monoback, monowhite	-	-	-	-	-
	<u>Grayscale</u>	8/9/10/12/14/16bpp	-	-	16/32bpp	-	-
<b>RGB</b>	RGB 1:2:1 (4-bit color)	4bpp	-	-	-	-	-
	RGB 3:3:2 (8-bit color)	8bpp	-	-	-	-	-
	RGB 5:5:5 (High color)	16bpp	-	-	-	-	-
	RGB 5:6:5 (High color)	16bpp	-	-	-	-	-
	RGB/BGR	24/30 <sup>[p 1]</sup> /48bpp	32 <sup>[p 2]</sup> /64bpp	-	-	-	8bit->32bpp
	GBR <sup>[p 3]</sup>	-	-	8/9/10/12/14/16bpc	8/10/12/16bpc	-	-
<b>RGB Float</b>	GBR	-	-	32bpc	32bpc	-	-
<b>YUV</b>	YVU 4:1:0	-	-	(9bpp (YVU9)) <sup>[p 4]</sup>	-	-	-
	YUV 4:1:0	-	-	9bpp	-	-	-
	YUV 4:1:1	8bpc (UYVYY)	-	8bpc	-	(8bpc (NV11))	-
	YVU 4:2:0	-	-	(8bpc (YV12)) <sup>[p 4]</sup>	-	8 (NV21)	-
	YUV 4:2:0	-	-	8 <sup>[p 5]</sup> /9/10/12/14/16bpc	8/9/10/16bpc	8 (NV12)/10 (P010)/16bpc (P016)	-
	YVU 4:2:2	-	-	(8bpc (YV16)) <sup>[p 4]</sup>	-	(8bpc (NV61))	-
	YUV 4:2:2	8bpc (YUYV <sup>[p 6]</sup> and UYVY) <sup>[p 7]</sup>	-	8 <sup>[p 8]</sup> /9/10/12/14/16bpc	8/9/10/12/16bpc	8 (NV16)/10bpc (NV20 a.k.a. P210) <sup>[p 9]</sup>	-
	YUV 4:4:0	-	-	8/10/12bpc	-	-	-
	YVU 4:4:4	-	-	(8bpc (YV24)) <sup>[p 4]</sup>	-	8bpc (NV42)	-
	YUV 4:4:4	(10 (Y410) and 16bpc (Y416))	16bpc <sup>[p 10]</sup>	8 <sup>[p 11]</sup> /9/10/12/14/16bpc	8/9/10/12/16bpc	8bpc (NV24)	-
<b>XYZ</b>	XYZ 4:4:4 <sup>[p 12]</sup>	12bpc	-	-	-	-	-
<b>Bayer</b>	BGGR/RGGB/GBRG/GRBG	8/16bpp	-	-	-	-	-

- 10-bit color components with 2-bit padding (X2RGB10)
- RGBx (rgb0) and xBGR (0bgr) are also supported
- used in YUV-centric codecs such like H.264
- YVU9, YV12, YV16, and YV24 are supported as *rawvideo* codec in FFmpeg.
- I420 a.k.a. YUV420P
- aka YUY2 in Windows
- Y210 (YUYV 10bpc) is not supported. UYVY 10bpc without a padding is supported as *bitpacked* codec in FFmpeg. UYVY 10bpc with 2-bits padding is supported as *v210* codec in FFmpeg. 16bpc (Y216) is supported as *targa\_y216* codec in FFmpeg.
- I422 a.k.a. YUV422P
- 16bpc (P216) is not supported
- 8bpc (AYUV) is not supported
- I444 a.k.a. YUV444P
- used in JPEG2000

FFmpeg does not support IMC1-IMC4, AI44, CYMK, RGBE, Log RGB and other formats. It also does not yet support ARGB 1:5:5:5, 2:10:10:10, or other BMP bitfield formats that are not commonly used.

## Supported protocols

### Open standards

- IETF RFCs:
  - FTP
  - Gopher

- [HLS](#)
- [HTTP](#)
- [HTTPS](#)
- [RTP](#)
- [RTSP](#)
- [SCTP](#)
- [SDP](#)
- [SRTP](#)
- [TCP](#)
- [TLS](#)
- [UDP](#)
- [UDP-Lite](#)
- IETF I-Ds:<sup>[49]</sup>
  - [SFTP](#) (via libssh)
- Microsoft OSP:
  - [CIFS/SMB](#) (via libsmbclient)
  - [MMS](#) over TCP (MS-MMSP)
  - [MMS](#) over HTTP (MS-WMSP)

## De facto standards

- [RTSP over TLS](#)<sup>[50][51]</sup>
- [Icecast](#) protocol
- Adobe [RTMP](#), [RTMPT](#), [RTMPE](#), [RTMPTE](#) and [RTMPS](#)
- RealMedia [RTSP/RDT](#)

## Supported filters

---

FFmpeg supports, among others, the following filters.<sup>[52]</sup>

### Audio

- [Resampling](#) (aresample)
- [Pass/Stop filters](#)
  - [Low-pass filter](#) (lowpass)
  - [High-pass filter](#) (highpass)
  - [All-pass filter](#) (allpass)
  - [Butterworth Band-pass filter](#) (bandpass)
  - [Butterworth Band-stop filter](#) (bandreject)
- [Arbitrary Finite Impulse Response Filter](#) (afir)
- [Arbitrary Infinite Impulse Response Filter](#) (aiir)
- [Equalizer](#)
  - [Peak Equalizer](#) (equalizer)
  - [Butterworth/Chebyshev Type I/Type II Multiband Equalizer](#) (anequalizer)
  - [Low Shelving filter](#) (bass)
  - [High Shelving filter](#) (treble)
  - [Xbox 360 equalizer](#)
  - [FIR equalizer](#) (firequalizer)
  - [Biquad filter](#) (biquad)
- [Remove/Add DC offset](#) (dcshift)
- [Expression evaluation](#)
  - [Time domain expression evaluation](#) (aeval)
  - [Frequency domain expression evaluation](#) (afftfilt)
- [Dynamics](#)
  - [Limiter](#) (alimiter)
  - [Compressor](#) (acompressor)
  - [Dynamic range expander](#) (crystalizer)
  - [Side-chain Compressor](#) (sidechaincompress)
  - [Compander](#) (compand)
  - [Noise gate](#) (agate)
  - [Side-chain Noise gate](#) (sidechaingate)
- [Distortion](#)
  - [Bitcrusher](#) (acrusher)
- [Emphasis](#) (aemphasis)
- [Amplify/Normalizer](#)
  - [Volume](#) (volume)
  - [Dynamic Audio Normalizer](#) (dynaudnorm)
  - [EBU R 128 loudness normalizer](#) (loudnorm)
- [Modulation](#)
  - [Sinusoidal Amplitude Modulation](#) (tremolo)
  - [Sinusoidal Phase Modulation](#) (vibrato)
  - [Phaser](#) (aphaser)
  - [Chorus](#) (chorus)
  - [Flanger](#) (flanger)
  - [Pulsator](#) (apulsator)
- [Echo/Reverb](#)
  - [Echo](#) (aecho)



- Routing/Panning
  - Stereo widening (stereowiden)
  - Increase channel differences (extrastereo)
  - M/S to L/R (stereotools)
  - Channel mapping (channelmap)
  - Channel splitting (channelsplit)
  - Channel panning (pan)
  - Channel merging (amerge)
  - Channel joining (join)
  - for Headphones
    - Stereo to Binaural (earwax, ported from SoX)<sup>[53]</sup>
    - Bauer Stereo to Binaural (bs2b, via libbs2b)
    - Crossfeed (crossfeed)
    - Multi-channel to Binaural (sofalizer, requires libnetcdf)
  - Delay
    - Delay (adelay)
    - Delay by distance (compensationdelay)
- Fade
  - Fader (afade)
  - Crossfader (acrossfade)
- Audio time-scale/pitch modification
  - Time stretching (atempo)
  - Time-stretching and Pitch-shifting (rubberband, via librubberband)
- Editing
  - Trim (atrim)
  - Silence-padding (apad)
  - Silence remover (silenceremove)
- Show frame/channel information
  - Show frame information (ashowinfo)
  - Show channel information (astats)
  - Show silence ranges (silencedetect)
  - Show audio volumes (volumedetect)
  - ReplayGain scanner (replaygain)
- Modify frame/channel information
  - Set output format (aformat)
  - Set number of sample (asetnsamples)
  - Set sampling rate (asetrate)
- Mixer (amix)
- Synchronization (asyncts)
- HDCD data decoder (hdcd)
- Plugins
  - LADSPA (ladspa)
  - LV2 (lv2)
- Do nothing (anull)

## Video

- Transformations
  - Cropping (crop, cropdetect)
  - Fading (fade)
  - Scaling (scale)
  - Padding (pad)
  - Rotation (rotate)
  - Transposition (transpose)
  - Others:
    - Lens correction (lenscorrection)
    - OpenCV filtering (ocv)
    - Perspective correction (perspective)
- Temporal editing
  - Framerate (fps, framerate)
  - Looping (loop)
  - Trimming (trim)
- Deinterlacing (bwdif, idet, kerndeint, nnedi, yadif, w3fdif)
- Filtering
  - Blurring (boxblur, gblur, avgblur, sab, smartblur)
  - Convolution filters
    - Convolution (convolution)
    - Edge detection (edgedetect)
    - Sobel Filter (sobel)
    - Prewitt Filter (prewitt)
    - Unsharp masking (unsharp)
- Denoising (atadenoise, bitplanenoise, dctdnoiz, owdenoise, removegrain)
- Logo removal (delogo, removelogo)
- Subtitles (ASS, subtitles)
- Alpha channel editing (alphaextract, alphamerge)
- Keying (chromakey, colorkey, lumakey)
- Frame detection
  - Black frame detection (blackdetect, blackframe)
  - Thumbnail selection (thumbnail)
- Frame Blending (blend, tblend, overlay)
- Video stabilization (vidstabdetect, vidstabtransform)
- Color and Level adjustments
  - Balance and levels (colorbalance, colorlevels)
  - Channel mixing (colorchannelmixer)
  - Color space (colorspace)
  - Parametric adjustments (curves, eq)
- Histograms and visualization
  - CIE Scope (ciescope)
  - Vectorscope (vectorscope)
  - Waveform monitor (waveform)
  - Color histogram (histogram)
- Drawing
- OCR
- Quality measures
  - SSIM (ssim)
  - PSNR (psnr)
- Lookup Tables
  - lut, lutrgb, lutyuv, lut2, lut3d, haldclut

## Applications

---

### Legal aspects

FFmpeg contains more than 100 codecs,<sup>[54]</sup> most of which use compression techniques of one kind or another. Many such compression techniques may be subject to legal claims relating to software patents.<sup>[55]</sup> Such claims may be enforceable in countries like the United States which have implemented software patents, but are considered unenforceable or void in member countries of the European Union, for example. Patents for many older codecs, including AC3 and all MPEG-1 and MPEG-2 codecs, have expired.

FFmpeg is licensed under the LGPL license, however, if a particular build of FFmpeg is linked against any GPL libraries (notably x264), then the entire binary is licensed under the GPL.

## Projects using FFmpeg

FFmpeg is used by software such as VLC media player, xine, Shotcut, Cinelerra-GG video editor, Plex, Kodi, Blender, HandBrake, YouTube,<sup>[56]</sup> VirtualDub2, a VirtualDub fork,<sup>[57]</sup> and MPC-HC,<sup>[58]</sup> it handles video and audio playback in Google Chrome,<sup>[58]</sup> and Linux version of Firefox.<sup>[59]</sup> Graphical user interface front-ends for FFmpeg have been developed, including XMedia Recode.

FFmpeg is used by ffdshow, LAV Filters, GStreamer FFmpeg plug-in, Perian, OpenMAX IL, and FFmpegInterop to expand the encoding and decoding capabilities of their respective multimedia platform.

## See also

- MEncoder, a similar project
- List of open-source codecs

## References


1. "Initial revision - git.videolan.org/ffmpeg.git/commit" (<https://web.archive.org/web/20131225083213/http://git.videolan.org/?p=ffmpeg.git%3Ba%3Dcommit%3Bh%3D9aeeeb63f7e1ab7b0b7bb839a5f258667a2d2d78>). git.videolan.org. 2000-12-20. Archived from the original (<http://git.videolan.org/?p=ffmpeg.git;a=commit;h=9aeeeb63f7e1ab7b0b7bb839a5f258667a2d2d78>) on 2013-12-25. Retrieved 2013-05-11.
2. "version 4.3.1" ([https://git.videolan.org/?p=ffmpeg.git;a=blob\\_plain;f=Changelog;hb=n4.3.1](https://git.videolan.org/?p=ffmpeg.git;a=blob_plain;f=Changelog;hb=n4.3.1)). *FFmpeg Git*. 2020-07-11. Retrieved 2020-07-11.
3. "Developer Documentation" (<https://ffmpeg.org/developer.html>). ffmpeg.org. 2011-12-08. Retrieved 2012-01-04.
4. "Platform Specific Information" (<https://ffmpeg.org/platform.html>). *FFmpeg.org*. Retrieved 25 February 2020.
5. "Download" (<https://ffmpeg.org/download.html>). *ffmpeg.org*. FFmpeg. Retrieved 2012-01-04.
6. FFmpeg can be compiled with various external external libraries, some of which have licenses that are incompatible with the FFmpeg's primary license, the GNU GPL.
7. "FFmpeg: Lavf: I/O and Muxing/Demuxing Library" ([https://www.ffmpeg.org/doxygen/2.2/group\\_\\_libavf.html](https://www.ffmpeg.org/doxygen/2.2/group__libavf.html)). *ffmpeg.org*. Retrieved 21 October 2016.
8. Bellard, Fabrice (18 February 2006). "FFmpeg naming and logo" (<https://ffmpeg.org/pipermail/ffmpeg-devel/2006-February/010315.html>). *FFmpeg developer mailing list*. FFmpeg website. Retrieved 24 December 2011.
9. Carlsen, Steve (1992-06-03). "TIFF 6.0 specification" (<ftp://ftp.sgi.com/graphics/tiff/TIFF6.ps>) (PS). Aldus. p. 98. Retrieved 2016-08-14. "Zig-Zag Scan" Alt URL (<http://partners.adobe.com/public/developer/en/tiff/TIFF6.pdf>)
10. "FFmpeg License and Legal Considerations" (<https://www.ffmpeg.org/legal.html>). ffmpeg.org. Retrieved 2012-01-04.
11. Niedermayer, Michael. "[FFmpeg-devel] FFmpeg's future and resigning as leader" (<https://ffmpeg.org/pipermail/ffmpeg-devel/2015-July/176489.html>). Retrieved 2015-09-22.
12. "FFmpeg and a thousand fixes" (<https://security.googleblog.com/2014/01/ffmpeg-and-thousand-fixes.html>). *googleblog.com*. January 10, 2014. Retrieved 21 October 2016.
13. "ffserver – FFmpeg" (<https://trac.ffmpeg.org/wiki/ffserver>). *trac.ffmpeg.org*. Retrieved 2018-02-03.
14. "ffserver program being dropped" (<https://ffmpeg.org/index.html#ffserv>). *ffmpeg.org*. 2016-07-10. Retrieved 2018-02-03.
15. "ffmpeg.org/download.html#releases" (<https://ffmpeg.org/download.html#releases>). ffmpeg.org. Retrieved 2015-04-27.
16. "NUT" (<http://wiki.multimedia.cx/index.php?title=NUT>). Multimedia Wiki. 2012. Retrieved 2014-01-03.
17. Glaser, Fiona (2010-07-23), *Diary Of An x264 Developer: Announcing the world's fastest VP8 decoder* (<https://web.archive.org/web/20100930181634/http://x264dev.multimedia.cx/?p=499>), archived from the original (<http://x264dev.multimedia.cx/?p=499>) on 2010-09-30, retrieved 2012-01-04
18. *FFmpeg Announces High-Performance VP8 Decoder* (<http://news.slashdot.org/story/10/07/24/1227241/FFmpeg-Announces-High-Performance-VP8-Decoder>), Slashdot, 2010-07-24, retrieved 2012-01-04
19. "FFmpeg Goes WebM, Enabling VP8 for Boxee & Co" (<http://newteevee.com/2010/06/17/ffmpeg-goes-webm-enabling-vp8-for-boxee-co/>). newteevee.com. 2010-06-17. Retrieved 2012-01-04. "...with VLC, Boxee, MythTV, Handbrake and MPlayer being some of the more popular projects utilizing FFmpeg..."
20. "Native VP9 decoder is now in the Git master branch" (<https://launchpad.net/ffmpeg/+announcement/12045>). *Launchpad*. 2013-10-03. Retrieved 2013-10-21.
21. "FFmpeg Now Features Native HEVC/H.265 Decoder Support" (<http://news.softpedia.com/news/FFmpeg-Now-Features-Native-HEVC-H-265-Decoder-Support-391582.shtml>). Softpedia. 2013-10-16. Retrieved 2013-10-16.
22. FFmpeg (2016-02-15). "February 15th, 2016, FFmpeg 3.0 "Einstein" " (<https://ffmpeg.org/index.html#pr3.0>). Retrieved 2016-04-02.

23. FFmpeg (2017-10-15). "October 15th, 2017, FFmpeg 3.4 "Cantor" " (<https://ffmpeg.org/index.html#pr3.4>). Retrieved 2019-05-10.
24. FFmpeg (2018-11-06). "November 6th, 2018, FFmpeg 4.1 "al-Khwarizmi" " (<https://ffmpeg.org/index.html#pr4.1>). Retrieved 2019-05-10.
25. Jan Ozer (2019-03-04). "Good News: AV1 Encoding Times Drop to Near-Reasonable Levels" (<https://www.streamingmedia.com/Articles/ReadArticle.aspx?ArticleID=130284>). *StreamingMedia.com*. Retrieved 2019-05-10.
26. *Libav project site* (<http://www.libav.org/>), retrieved 2012-01-04
27. Ronald S. Bultje (2011-03-14), *Project renamed to Libav* (<https://web.archive.org/web/20161107163133/http://article.gmane.org/gmane.comp.video.ffmpeg.devel/129579>), archived from the original (<http://article.gmane.org/gmane.comp.video.ffmpeg.devel/129579>) on 2016-11-07, retrieved 2012-01-04
28. *A group of FFmpeg developers just forked as Libav* ([https://www.phoronix.com/scan.php?page=news\\_item&px=OTlwNw](https://www.phoronix.com/scan.php?page=news_item&px=OTlwNw)), Phoronix, 2011-03-14, retrieved 2012-01-04
29. *What happened to FFmpeg* (<https://codecs.multimedia.cx/2011/03/what-happened-to-ffmpeg/>), 2011-03-30, retrieved 2012-05-19
30. *FFmpeg turmoil* (<https://lwn.net/Articles/423702/>), 2011-01-19, retrieved 2012-01-04
31. "The FFmpeg/Libav situation" (<http://blog.pkh.me/p/13-the-ffmpeg-libav-situation.html>). *blog.pkh.me*. Retrieved 2015-09-22.
32. "This video of Linux desktop (X11) was captured by ffmpeg and encoded in realtime" (<https://commons.wikimedia.org/wiki/File:Gentoo-portage-installing-and-using-of-equery.webm>).
33. "FFmpeg Automated Testing Environment" (<http://fate.multimedia.cx/>). *Fate.multimedia.cx*. Retrieved 2012-01-04.
34. "FFmpeg Hardware Acceleration" (<https://trac.ffmpeg.org/wiki/HWAccelIntro>). *ffmpeg.org Wiki*. Retrieved 2016-11-12.
35. "Hardware/VA-API – FFmpeg" (<https://trac.ffmpeg.org/wiki/Hardware/VA-API>). *trac.ffmpeg.org*. Retrieved 2017-10-16.
36. "FFmpeg Changelog" (<https://github.com/FFmpeg/FFmpeg/blob/master/Changelog>).
37. "Changelog" (<http://svn.mplayerhq.hu/ffmpeg/trunk/Changelog?revision=8747&view=marku>). *FFmpeg trunk SVN*. FFmpeg. 17 April 2007. Retrieved 26 April 2007.
38. "git.ffmpeg.org Git - ffmpeg.git/commit" (<https://git.ffmpeg.org/gitweb/ffmpeg.git/commit/43778a501f1bfbceeddcc8eaeaa2ea2b3506beeda>). *git.ffmpeg.org*. Retrieved 2018-04-23.
39. "ATrac3plus decoder - git.videolan.org Git - ffmpeg.git/commit" (<https://web.archive.org/web/20131230235231/http://git.videolan.org/?p=ffmpeg.git%3Ba%3Dcommit%3Bh%3D2e1fb96af36348479e14aa85f445915ea09e70fa>). *git.videolan.org*. 2013-12-28. Archived from the original (<http://git.videolan.org/?p=ffmpeg.git;a=commit;h=2e1fb96af36348479e14aa85f445915ea09e70fa>) on 2013-12-30. Retrieved 2013-12-28.
40. "FFmpeg development mailing list" (<https://archive.today/20070811203618/http://lists.mplayerhq.hu/pipermail/ffmpeg-devel/2007-May/028760.html>). *FFmpeg development*. FFmpeg. 7 May 2007. Archived from the original (<http://lists.mplayerhq.hu/pipermail/ffmpeg-devel/2007-May/028760.html>) on 11 August 2007. Retrieved 24 December 2010.
41. vitor (13 April 2008). "FFmpeg development mailing list" (<https://lists.mplayerhq.hu/pipermail/ffmpeg-cvslog/2008-April/013231.html>). *FFmpeg development*. FFmpeg website. Retrieved 14 April 2008.
42. vitor (30 March 2008). "FFmpeg development mailing list" (<https://lists.mplayerhq.hu/pipermail/ffmpeg-cvslog/2008-March/012963.html>). *FFmpeg development*. FFmpeg website. Retrieved 30 March 2008.
43. benoit (14 April 2008). "FFmpeg development mailing list" (<https://lists.mplayerhq.hu/pipermail/ffmpeg-cvslog/2008-April/013245.html>). *FFmpeg development*. FFmpeg website. Retrieved 14 April 2008.
44. Michael Niedermayer, Timothy Gu (2014-12-05). "RELEASE NOTES for FFmpeg 2.5 "Bohr" " ([https://web.archive.org/web/20141208011850/http://git.videolan.org/?p=ffmpeg.git%3Ba%3Dblob%3Bf%3DRELEASE\\_NOTES%3Bh%3Drelease%2F2.5](https://web.archive.org/web/20141208011850/http://git.videolan.org/?p=ffmpeg.git%3Ba%3Dblob%3Bf%3DRELEASE_NOTES%3Bh%3Drelease%2F2.5)). VideoLAN. Archived from the original ([http://git.videolan.org/?p=ffmpeg.git;a=blob;f=RELEASE\\_NOTES;hb=release/2.5](http://git.videolan.org/?p=ffmpeg.git;a=blob;f=RELEASE_NOTES;hb=release/2.5)) on 2014-12-08. Retrieved 2014-12-05.
45. ramiro (18 March 2008). "FFmpeg development mailing list" (<https://web.archive.org/web/20080817072304/http://lists.mplayerhq.hu/pipermail/ffmpeg-cvslog/2008-March/012708.html>). *FFmpeg development*. FFmpeg website. Archived from the original (<http://lists.mplayerhq.hu/pipermail/ffmpeg-cvslog/2008-March/012708.html>) on 17 August 2008. Retrieved 18 March 2008.
46. banan (8 June 2008). "FFmpeg development mailing list" (<https://web.archive.org/web/20090114195646/http://lists.mplayerhq.hu/pipermail/ffmpeg-cvslog/2008-June/014417.html>). *FFmpeg development*. FFmpeg website. Archived from the original (<http://lists.mplayerhq.hu/pipermail/ffmpeg-cvslog/2008-June/014417.html>) on 14 January 2009. Retrieved 8 June 2008.
47. faust3 (21 March 2008). "FFmpeg development mailing list" (<https://web.archive.org/web/20080425024126/http://lists.mplayerhq.hu/pipermail/ffmpeg-cvslog/2008-March/012799.html>). *FFmpeg development*. FFmpeg website. Archived from the original (<http://lists.mplayerhq.hu/pipermail/ffmpeg-cvslog/2008-March/012799.html>) on 25 April 2008. Retrieved 21 March 2008.
48. "FFmpeg: libavutil/pixfmt.h File Reference - enum AVPixelFormat" ([https://ffmpeg.org/doxygen/trunk/pixfmt\\_8h.html#a9a8e335cf3be472042bc9f0cf80cd4c5](https://ffmpeg.org/doxygen/trunk/pixfmt_8h.html#a9a8e335cf3be472042bc9f0cf80cd4c5)). FFmpeg Project. 2015-11-17. Retrieved 2015-11-18.
49. van Kesteren, Anne (2010-09-01). "Internet Drafts are not Open Standards" (<https://annevankesteren.nl/2010/09/draft-vs-standards>). *annevankesteren.nl*. Self-published. Retrieved 2015-03-22.
50. Real Time Streaming Protocol 2.0 (RTSP) draft-ietf-mmusic-rfc2326bis-40 (<https://tools.ietf.org/html/draft-ietf-mmusic-rfc2326bis-40>) P.231
51. "rtsp: Support tls-encapsulated RTSP - git.videolan.org Git - ffmpeg.git/commit" (<http://git.videolan.org/gitweb.cgi/ffmpeg.git/?a=commit;h=8b2e9636c57b22582143467a8a06b509b47b92f9>). *videolan.org*. Retrieved 21 October 2016.
52. "FFmpeg Filters" (<https://ffmpeg.org/ffmpeg-filters.html>). *ffmpeg.org*. Retrieved 2017-03-27.
53. How it works (<https://web.archive.org/web/20080928130012/http://www.geocities.com/beinges/works.htm>) earwax.ca
54. "Codecs list" ([https://ffmpeg.org/general.html#Supported-File-Formats\\_002c-Codecs-or-Features](https://ffmpeg.org/general.html#Supported-File-Formats_002c-Codecs-or-Features)). *ffmpeg.org*. Retrieved 2012-01-01.
55. "Legal information on FFmpeg's website" (<https://ffmpeg.org/legal.html>). *ffmpeg.org*. Retrieved 2012-01-04.

- 56. "Google's YouTube Uses FFmpeg | Breaking Eggs And Making Omelettes" (<https://multimedia.cx/eggs/googles-youtube-uses-ffmpeg/>). Multimedia.cx. 2011-02-08. Retrieved 2012-08-06.
- 57. VirtualDub2 (<https://sourceforge.net/p/vdfiltermod/wiki/Home/>)
- 58. "FFmpeg-based Projects" (<https://trac.ffmpeg.org/wiki/Projects>). Ffmpeg.org. Retrieved 2012-01-04.
- 59. "Firefox Enables FFmpeg Support By Default" ([https://www.phoronix.com/scan.php?page=news\\_item&px=Firefox-FFmpeg-Default](https://www.phoronix.com/scan.php?page=news_item&px=Firefox-FFmpeg-Default)). Phoronix. 2015-11-15. Retrieved 2015-11-18.

## External links

---

- Official website (<https://ffmpeg.org>) 

---

Retrieved from "<https://en.wikipedia.org/w/index.php?title=FFmpeg&oldid=985108077>"

---

This page was last edited on 24 October 2020, at 00:37 (UTC).

Text is available under the Creative Commons Attribution-ShareAlike License; additional terms may apply. By using this site, you agree to the Terms of Use and Privacy Policy. Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a non-profit organization.