

# Einführung in FFmpeg

Reto Kromer • AV Preservation by reto.ch

Memoriav-Workshop  
**FFmpeg-Aufbaukurs**  
Bern, 16. März 2026

1

# Benutzerschnittstellen

- Lochkarten und Matrixdrucker
- grafische Benutzeroberfläche  
Englisch: graphical user interface (GUI)
- Kommandozeile  
Englisch: command-line interface (CLI)
- berührungslose Schnittstellen

2

ASCII (1977/1986)																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0x	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1x	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2x	SP	!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
3x	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4x	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5x	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6x	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7x	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL

Changed or added in 1963 version  
 Changed in both 1963 version and 1965 draft

source: wikipedia.org

3

# Unix/Linux-Befehlsstruktur

**\$0**            **\$1**            **\${n}**  
*command argument\_1 ... argument\_n*

Beispiele gängiger Syntaxen der Argumente:

*--parameter*  
*--parameter=value*  
*-p*  
*-p value*

4

# Software

5

## FFmpeg wird überall eingesetzt

- VLC
- mpv
- Audacity
- Handbrake
- Shutter Encoder
- AEO-Light
- vrecord
- QCTools
- Google Chrome
- YouTube
- Adobe Premiere
- DaVinci Resolve
- ...

6

## Die FFmpeg-Familie

### Programme

- ffmpeg
- ffprobe
- ffplay

### Bibliotheken

- libavformat
- libavcodec
- libavfilter
- libavutil
- libavdevice
- libswscale
- libswresample

7

## Dateiumwandlungen

### ffmpeg (CLI)

→ [ffmpeg.org](https://ffmpeg.org)

### FFmpeg Cookbook for Archivists

→ [avpres.net/FFmpeg/](https://avpres.net/FFmpeg/)

### ffmprovisr

→ [amiaopensource.github.io/ffmprovisr/](https://amiaopensource.github.io/ffmprovisr/)

8

## Metadaten extrahieren

**MediaInfo** (GUI, CLI)

→ [mediaarea.net/MediaInfo](http://mediaarea.net/MediaInfo)

**ffprobe** (CLI)

→ [ffmpeg.org](http://ffmpeg.org)

9

## Mediaplayer

**VLC** (GUI)

→ [www.videolan.org/vlc/](http://www.videolan.org/vlc/)

**mpv** (CLI/GUI)

→ [mpv.io](http://mpv.io)

**ffplay** (CLI)

→ [ffmpeg.org](http://ffmpeg.org)

10

## Grafische Benutzeroberfläche

- ...
- Audacity
- Handbrake
- Shutter Encoder

11

## FFmpeg-Befehlsstruktur

**\$0**      **\$1**      **\${n}**  
*command argument\_1 ... argument\_n*

**FFmpeg-Syntax der Argumente:**

*-parameter*  
*-parameter value*  
*-p*  
*-p value*

12

## FFmpeg-Syntax

**ffmpeg**

[*global\_options*]

[*input\_options\_n*] -i *input\_file\_n*

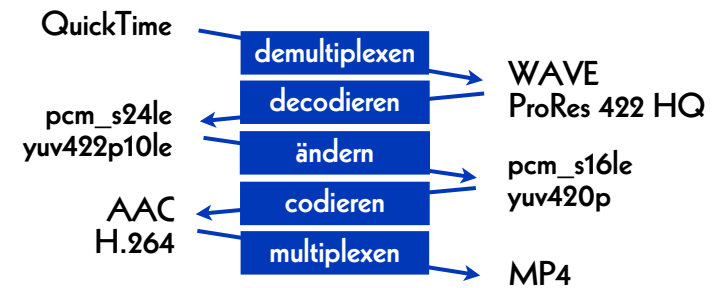
[*output\_options\_n*] *output\_file\_n*

**ffprobe** [*input\_options*] *input\_file*

**ffplay** [*input\_options*] *input\_file*

13

## Beispiel: Bild und Ton



14

## Programmbibliotheken

demultiplexen: libavformat

decodieren: libavcodec

ändern: libavfilter

codieren: libavcodec

multiplexen: libavformat

15

# Übungen

16

## Bilddatei erzeugen

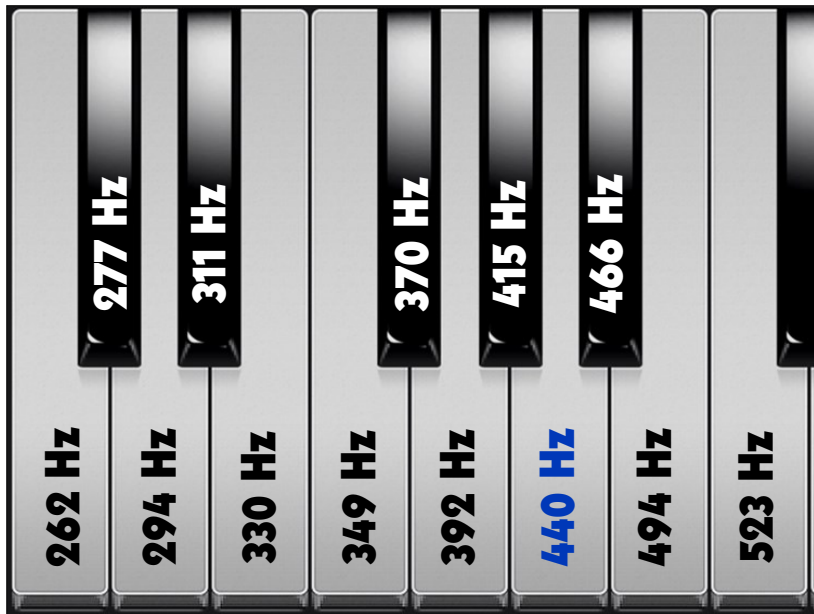
```
ffmpeg  
-f lavfi -i mandelbrot  
-t 10  
-c:v rawvideo  
-pix_fmt uyvy422  
mandelbrot.avi
```

17

## Bilddatei abspielen

```
ffplay  
mandelbrot.avi
```

18



19

## Tondatei erzeugen

```
ffmpeg  
-f lavfi -i "sine=frequency=440"  
-t 10  
-c:a pcm_s16le  
-ar 48k  
-ac 2  
la.wav
```

Übung: andere Werte anstatt 440 benutzen (z. B. 262, 330, 392 oder 523)

20

## Tondatei abspielen

```
ffplay  
la.wav
```

21

## Dateien zusammenfügen

```
ffmpeg  
-i mandelbrot.avi  
-i la.wav  
-c:v copy  
-c:a copy  
mandela.avi
```

22

## AV-Datei abspielen

```
ffplay  
mandela.avi
```

23

## Metadaten extrahieren

```
ffprobe  
mandela.avi
```

24

## Container-Metadaten

```
ffprobe  
-show_format  
mandela.avi
```

25

## Codec-Metadaten

```
ffprobe  
-show_streams  
mandela.avi
```

26

## Container und Codec

```
ffprobe  
-show_format  
-show_streams  
mandela.avi
```

27

## Metadaten formatieren

```
ffprobe  
-show_format  
-show_streams  
-print_format json  
mandela.avi
```

Übung: flat und xml anstelle von json benutzen

28

## Metadaten speichern

```
ffprobe  
-show_format  
-show_streams  
-print_format json  
mandela.avi  
> mandela.txt
```

29

## Metadaten extrahieren

```
mediainfo  
mandela.avi
```

30

## Hilfe finden

```
ffmpeg -h  
ffmpeg -codecs  
ffmpeg -decoders  
ffmpeg -h decoder=aac  
ffmpeg -encoders  
ffmpeg -h encoder=libx264  
ffmpeg -filters  
ffmpeg -h filter=scale  
ffmpeg -pix_fmts  
ffmpeg -sample_fmts
```

31

# Dateiumwandlungen

32

## Container ändern

```
ffmpeg  
-i mandelbrot.avi  
-c copy  
mandelbrot.mov
```

33

## MD5-Prüfsummen (1)

```
ffmpeg  
-i mandelbrot.avi  
-f framemd5  
mandelbrot.avi_framemd5.txt
```

34

## MD5-Prüfsummen (2)

```
ffmpeg  
-i mandelbrot.mov  
-f framemd5  
mandelbrot.mov_framemd5.txt
```

35

## Prüfsummen vergleichen

```
# Linux, Windows Subsystem for Linux,  
# macOS  
diff -s  
mandelbrot.avi_framemd5.txt  
mandelbrot.mov_framemd5.txt  
  
# Windows  
fc  
mandelbrot.avi_framemd5.txt  
mandelbrot.mov_framemd5.txt
```

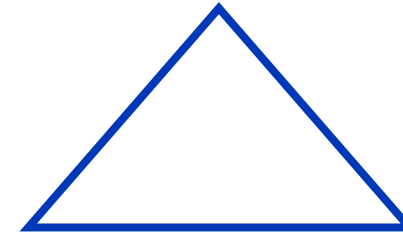
36

# Einzelbilder abspielen

```
ffplay  
-loop 0  
DUFAY_TIFF/Dufay_%06d.tif
```

37

## Bildqualität



Codierungszeit

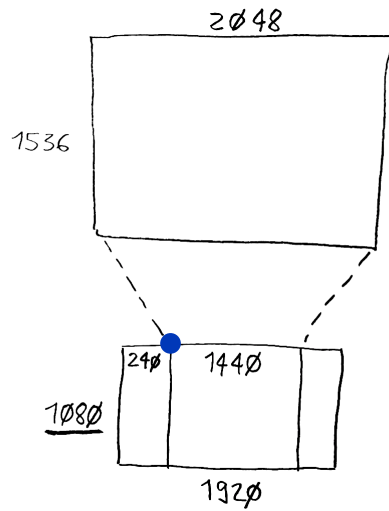
Dateigrösse

38

2K  
4:3



HD  
16:9

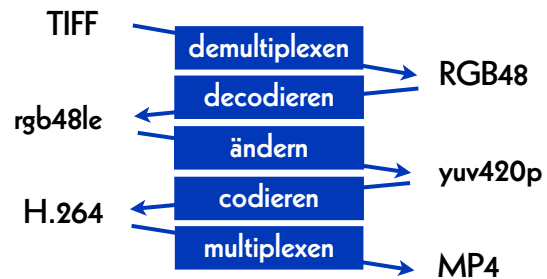


39

# H.264

40

## Von TIFF nach H.264



41

## Archiv -> Zugang

```
ffmpeg  
-f image2 -framerate 24  
-i DUFAY_TIFF/Dufay_%06d.tif  
-filter:v  
  "scale=1440:1080:flags=lanczos,  
  pad=1920:1080:240:0"  
-c:v libx264 -preset veryslow -crf 21  
-pix_fmt yuv420p  
-movflags +faststart+write_colr  
Dufay_21_H264.mp4
```

42

## Qualitätskontrolle

43

## Qualitätskontrolle

- technische Metadaten prüfen
- Signal analysieren
- Bild ansehen und Ton hören
  
- geteilter Bildschirm aus zwei Dateien
- Differenzdatei zweier Dateien

44

## Werkzeuge zur Qualitätskontrolle

### Container und Codec

- MediaInfo, ffprobe, MediaConch
- hexdump, fq

### Bild- und Toninhalt

- QCTools, qcli, SignalServer
- VLC, mpv, ffplay, GridPlayer

45

## Geteilter Bildschirm

```
ffmpeg
-i Dufay_21_H264.mp4
-i Dufay_35_H264.mp4
-filter_complex
  "[0] crop=iw/2:ih:0:0 [left];
  [1] crop=iw/2:ih:iw/2:0 [right];
  [left][right] hstack [final]"
-map "[final]"
Dufay_split.mp4
```

46

## Filter testen

```
ffplay
-vf "negate"
Dufay_1_H264.mp4
```

47

## Differenzdatei

```
ffmpeg
-i Dufay_21_H264.mp4
-i Dufay_35_H264.mp4
-filter_complex
  "[1] format=yuva444p,
  lut=c3=128,
  negate [1_with_alpha];
  [0][1_with_alpha] overlay [final]"
-map "[final]"
Dufay_delta.mp4
```

48

## Tondatei erzeugen

```
ffmpeg  
-f lavfi -i "anoisesrc=color=brown"  
-filter:a "tremolo=f=0.1:d=0.9"  
-c:a pcm_s24le  
-ar 96k  
-ac 2  
-t 60  
seashore_good.wav
```

49

## Tondatei abspielen

```
ffplay  
seashore_good.wav
```

50

## Tondatei beschädigen

```
ffmpeg  
-i seashore_good.wav  
-c copy  
-bsf:a "noise=amount=-1"  
seashore_bad.wav
```

51

## beschädigte Tondatei abspielen

```
ffplay  
seashore_bad.wav
```

52

## Lautstärke anzeigen (unversehrt)

```
ffplay
-f lavfi
  "amovie=seashore_good.wav,
  asplit [a][out1];
[a] showvolume=c=VOLUME:
  w=1000:h=100:
  ds=lin [out0]"
```

53

## Lautstärke anzeigen (beschädigt)

```
ffplay
-f lavfi
  "amovie=seashore_bad.wav,
  asplit [a][out1];
[a] showvolume=c=VOLUME:
  w=1000:h=100:
  ds=lin [out0]"
```

54

## Schallwelle anzeigen (unversehrt)

```
ffplay
-f lavfi
  "amovie=seashore_good.wav,
  asplit [a][out1];
[a] showwaves=mode=cline [out0]"
```

Übung: point oder p2p anstatt cline benutzen

55

## Schallwelle anzeigen (beschädigt)

```
ffplay
-f lavfi
  "amovie=seashore_bad.wav,
  asplit [a][out1];
[a] showwaves=mode=cline [out0]"
```

Übung: point oder p2p anstatt cline benutzen

56

## Spektrogramm anzeigen (unversehrt)

```
ffplay  
-f lavfi  
  "amovie=seashore_good.wav,  
   asplit [a][out1];  
 [a] showspectrum=mode=separate:  
   color=intensity:  
   slide=1:  
   scale=cbrt [out0]"
```

57

## Spektrogramm anzeigen (beschädigt)

```
ffplay  
-f lavfi  
  "amovie=seashore_bad.wav,  
   asplit [a][out1];  
 [a] showspectrum=mode=separate:  
   color=intensity:  
   slide=1:  
   scale=cbrt [out0]"
```

58

## Personalisieren

59

## Schriftarten finden

```
# Windows Subsystem for Linux, macOS  
ls /Library/Fonts  
  
# Ubuntu  
fc-list  
  
# Windows  
dir \Windows\Fonts
```

60

## Wasserzeichen hinzufügen

```
ffmpeg
-i Dufay_21_H264.mp4
-filter:v
  "drawtext=text='watermark':
    fontfile='/Library/Fonts/Arial.ttf':
    fontsize=35:
    fontcolor=white:
    alpha=0.25:
    x=(w-text_w)/2:y=(h-text_h)/2"
with_watermark.mp4
```

61

## Timecode hinzufügen

```
ffmpeg
-i Dufay_21_H264.mp4
-filter:v
  "drawtext=timecode='01\ :00\ :30\ :00':
    rate=25:
    fontfile='/Library/Fonts/Arial.ttf':
    fontsize=35:
    fontcolor=white:
    x=(w-text_w)/2:y=h/1.2"
with_timecode.mp4
```

62

## Logo hinzufügen

```
ffmpeg
-i Dufay_21_H264.mp4
-i logo.png
-filter_complex
  "overlay=10:main_h-overlay_h-10"
with_logo.mp4
```

63

AV Preservation by  
**reto.ch**

Sandrainstrasse 3/7  
3007 Bern  
Switzerland

reto.ch  
info@reto.ch



64