

Data Migration, Our Frenemy

Reto Kromer • AV Preservation by reto.ch

No Time to Wait!

Open Media, Open Formats, Open Archives

Österreichisches Filmmuseum, Vienna, Austria

9–10 November 2017

The Reel Thing XLII

New Orleans, Louisiana, USA

29 November 2017

1

Table of Contents

- data migration
- handling of established and experimental file formats
- wishes for CELLAR

2

Lessons Learned

3

Data Migrations

2014

- our internal archive from LTO-4 to LTO-6 (5.7 PB)

2014–2017

- [...]

2017

- a client's archive from LTO-5 to LTO-7 (270 GB)

4

```
Terminal — -bash
[11:02:03]reto@Castor:~/TEST$ ./migratelto
Help:
./migratelto -h
[11:02:06]reto@Castor:~/TEST$ ./migratelto -h
Abstract:
migratelto - Migrate one or more LTO cartridges from one generation to
another.
Syntax:
migratelto {-f} -t [-i] [-n] [-x] | -h
Parameters:
-f from desk
-t to desk
-i new cartridge identifier
-n new cartridge name
-x path to script to execute
-h this help
See also:
man openlto
About:
Version: 2017-10-20_alpha
Website: https://avpres.net/openLTO/migratelto/
[11:02:35]reto@Castor:~/TEST$
```

5

```
Terminal — less · man ~/TEST/openlto.1
migratelto {-f desk} -t desk [-i identifier] [-n name] [-x path] | -h
Migrate from one LTO generation to another LTO generation.

-f one or more source desk's identifiers
-t destination desk's identifier
-i cartridge's identifier
-n cartridge's name
-x path to script to execute
The data are read from the source desk, piped to the script,
piped to writing procedure to the destination desk:
read | script | write

-h display a help message

mountlto -i identifier [-d desk] | -h
Mount an LTO cartridge which is loaded in a desk.

-d desk's identifier
-i cartridge's identifier
```

6

read | script | write

script to modify

- container
- codec
- both container and codec

7

Examples

8

Video

from:

- AVI / 8-bit and 10-bit uncompressed
- MOV / 8-bit and 10-bit uncompressed
- MP4 / 8-bit and 10-bit uncompressed

to:

- Matroska / FFV1

9

Container and Codec

- read file from source LTO
- demultiplex file
- decode file
 - $Y'CbCr$, 4:2:2, 8 bit, «raw» [uyvy422]
- encode file
- multiplex file
- write file to destination LTO

10

Container and Codec

- read file from source LTO
- demultiplex file
- decode file
 - $Y'CbCr$, 4:2:2, 10 bit, «raw» [yuv422p10le]
- encode file
- multiplex file
- write file to destination LTO

11

Film

from:

- TIFF
- DPX

to:

- Matroska / FFV1 using RGB48 and additional metadata

12

Reading

Reto Kromer: **Matroska and FFV1: One File Format for Film and Video Archiving?**, in «Journal of Film Preservation», n. 96 (April 2017), FIAF, Brussels, Belgium, p. 41–45

→ https://retokromer.ch/publications/JFP_96.html

13

Old Experimentations

from:

- AVI / HuffYUV
- AVI / FFV1 version 1

to:

- Matroska / FFV1 [version 3]

14

Mid_{old} Experimentations

from:

- AVI / CineForm (VC-5) with Bayer
- MOV / CineForm (VC-5) with Bayer
- MOV / ProRes

to:

- Matroska / FFV1 after de-mosaicking
- Matroska / ProRes

15

Container, Codec and Data

→ read file from source LTO

→ demultiplex file

→ decode data

- 12 bit [bayer_bggr16le]

→ encode data

→ multiplex file

→ write file to destination LTO

16

Container, Codec and Data

- read file from source LTO
- demultiplex file
- decode data
 - R'G'B', 4:4:4, 16 bit, «raw» [rgb48le]
- encode data
- multiplex file
- write file to destination LTO

17

Container, Codec and Data

- read file from source LTO
- demultiplex file
- decode data
- **filter data (de-mosaic)**
- encode data
- multiplex file
- write file to destination LTO

18

Container

- read file from source LTO
- demultiplex file
 - ProRes 422, 10 bit [yuv422p10le]
 - ProRes 4444, 10 bit [yuv444p10le or yuva444p10le] or 12 bit [yuv444p12le]
- multiplex file
- write file to destination LTO

19

Mid_{new} Experimentations

from:

- AVI / CineForm (VC-5) with Bayer
- MOV / CineForm (VC-5) with Bayer
- MOV / ProRes

to:

- Matroska / FFV1 version 4 [with Bayer]
- Matroska / ProRes with 12-bit support

20

New Experimentations

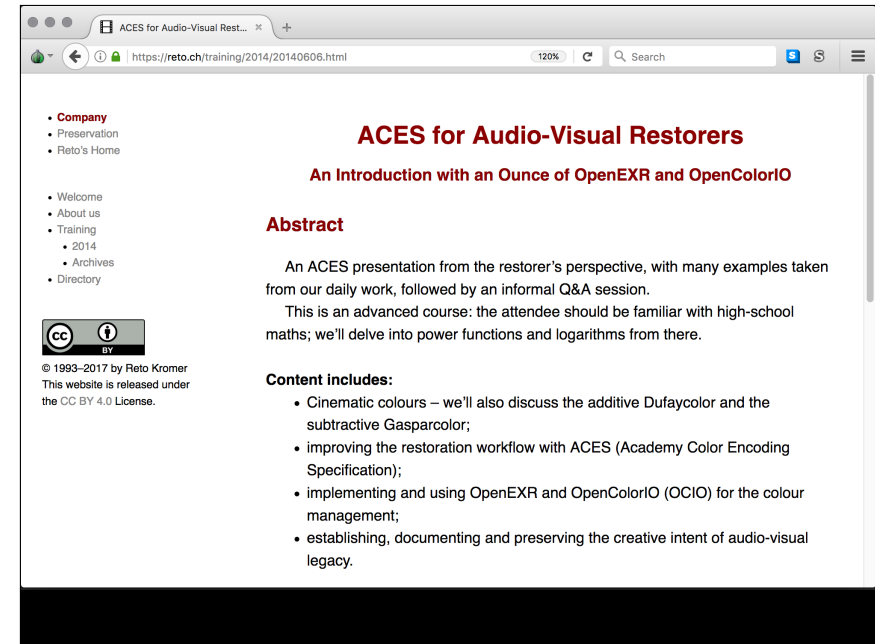
from:

- MXF / OpenEXR
- flavour of NUT / multiple RGB48
- flavour of NUT / RGB72 or $Y'CbCr$ 24-bit
- flavour of NUT / expanded OpenEXR

to:

- Matroska / FFV1 version 4 using RGB48, RGB72, floats and additional metadata

21



ACES for Audio-Visual Restorers

https://reto.ch/training/2014/20140606.html

- Company
- Preservation
- Reto's Home

- Welcome
- About us
- Training
 - 2014
 - Archives
- Directory

© 1993–2017 by Reto Kromer
This website is released under the CC BY 4.0 License.

ACES for Audio-Visual Restorers

An Introduction with an Ounce of OpenEXR and OpenColorIO

Abstract

An ACES presentation from the restorer's perspective, with many examples taken from our daily work, followed by an informal Q&A session.

This is an advanced course: the attendee should be familiar with high-school maths; we'll delve into power functions and logarithms from there.

Content includes:

- Cinematic colours – we'll also discuss the additive Dufaycolor and the subtractive Gasparcolor;
- improving the restoration workflow with ACES (Academy Color Encoding Specification);
- implementing and using OpenEXR and OpenColorIO (OCIO) for the colour management;
- establishing, documenting and preserving the creative intent of audio-visual legacy.

22

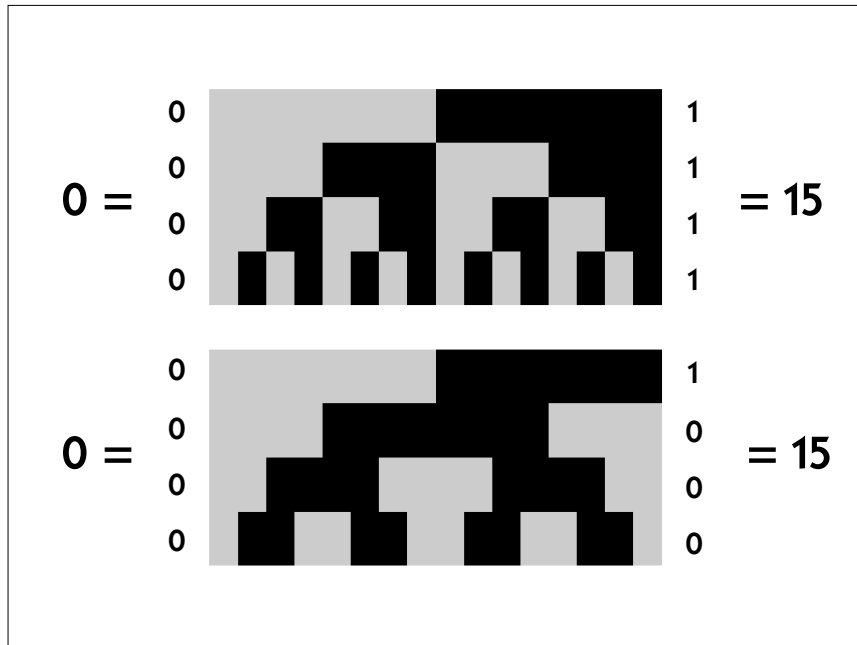
Wishes

23

EBML

- additional support:
 - Gray codes

24

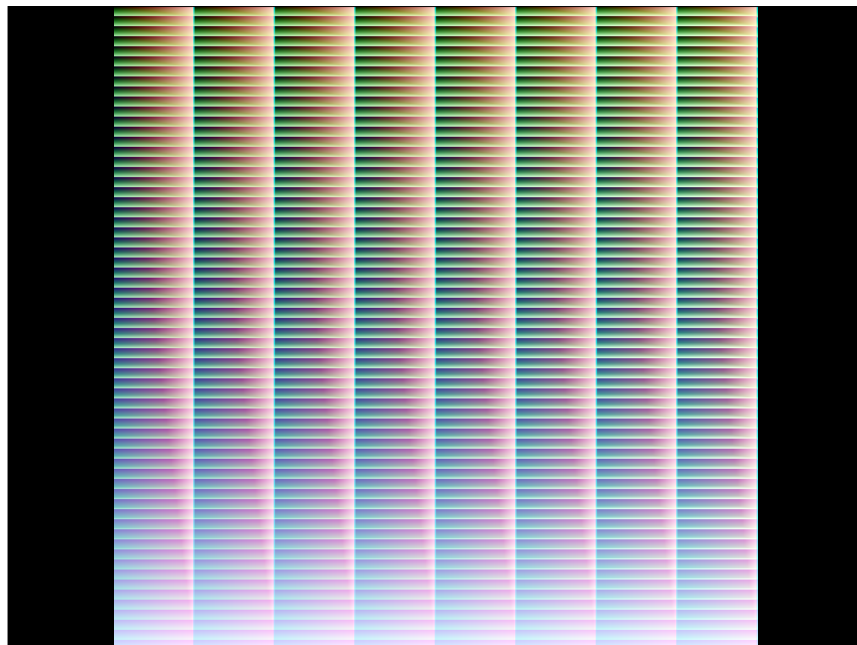


25

Matroska

- additional metadata
- additional integration:
 - 1D LUT
 - 3D LUT
- additional support:
 - Gray codes

26

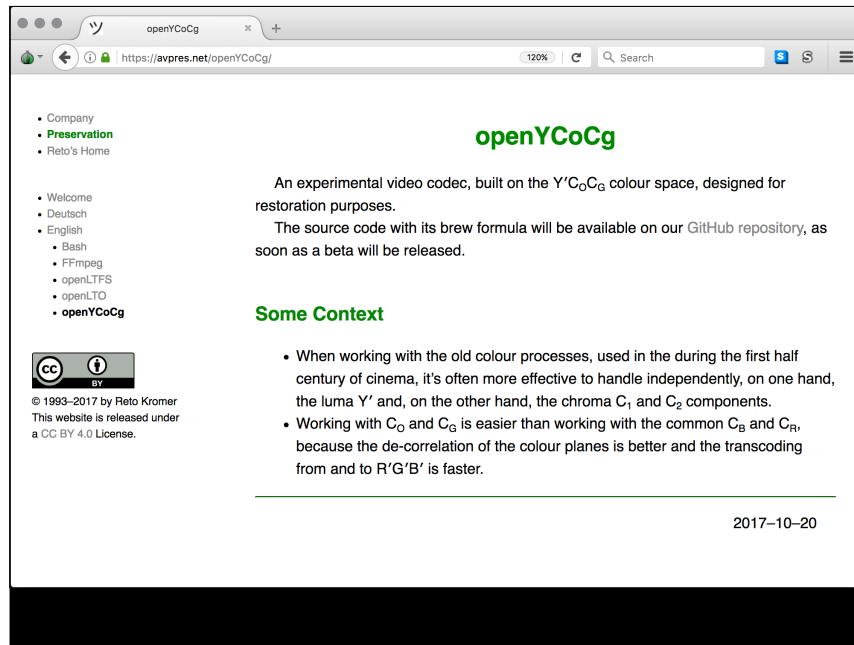


27

FFV1

- additional metadata
- additional encoding:
 - RGB72, $Y'CbCr$ 24-bit and floats
 - Bayer
- additional support:
 - Gray codes
 - $Y'CoCG$

28



29

$$\begin{pmatrix} R' \\ G' \\ B' \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1.396523 \\ 1 & -0.342793 & -0.711348 \\ 1 & 1.765078 & 0 \end{pmatrix} \begin{pmatrix} Y' \\ C_B \\ C_R \end{pmatrix}$$

$$\begin{pmatrix} Y' \\ C_B \\ C_R \end{pmatrix} = \begin{pmatrix} 0.299 & 0.587 & 0.114 \\ -0.168074 & -0.329965 & 0.498039 \\ 0.498039 & -0.417947 & -0.080992 \end{pmatrix} \begin{pmatrix} R' \\ G' \\ B' \end{pmatrix}$$

30

$$\begin{pmatrix} R' \\ G' \\ B' \end{pmatrix} = \begin{pmatrix} 1 & 1 & -1 \\ 1 & 0 & 1 \\ 1 & -1 & -1 \end{pmatrix} \begin{pmatrix} Y' \\ C_O \\ C_G \end{pmatrix}$$

$$\begin{pmatrix} Y' \\ C_O \\ C_G \end{pmatrix} = \begin{pmatrix} \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{2} & 0 & -\frac{1}{2} \\ -\frac{1}{4} & \frac{1}{2} & -\frac{1}{4} \end{pmatrix} \begin{pmatrix} R' \\ G' \\ B' \end{pmatrix}$$

31

FLAC

- additional support:
 - Gray codes
- additional encoding:
 - 24-bit sample format

32

Kate Murray:

«More adoption
means better sustainability.»

[while proudly speaking about MXF/JPEG 2000
at the Matroska/FFV1 symposium in Vienna]

33

Matroska Reader

- full transparency with DPX [et similia]
→ RAWcooked
- create, alter and inspect Matroska files
→ MKVToolNix
- verify Matroska and WebM files
→ mkvalidator

34

Matroska Player

- apply 1D and 3D LUT to the video stream
→ VLC
→ mpv
→ ffplay
→ FFmpeg

35

AV Preservation by reto.ch

chemin du Suchet 5
1024 Écublens
Switzerland

Web: reto.ch
Twitter: @retoch
Email: info@reto.ch



36