Matroska and FFV1: One File Format for Film and Video Archiving?

Reto Kromer

Reto Kromer was formerly Head of preservation at the Cinémathèque suisse, and lectured at the University of Lausanne and the Academy of Fine Arts, Vienna. He now has his own preservation company and lectures at the Bern Institute of Applied Sciences.

To have a single file format suitable for the preservation of both film and video has been the dream of many archives, especially smaller ones, since the beginning of digital moving images. Today, on the horizon, we can see something that will provide exactly this capability, in a free and open-source environment.

The following article is an updated version of three presentations I made last year on this topic:

- with Kieran O'Leary from the IFI Irish Film Archive[1] at the symposium *No Time to Wait: Standardising FFV1 and Matroska for Preservation*, Berlin, 18–20 July,
- at *The Reel Thing* technical symposium, Hollywood, 18–20 August,[2]
- at a meeting of the Memoriav video specialists, Bern,22 November.[3]

It presents the situation as at the beginning of 2017, and my goal is to discuss the evident potential of this new system, as well as to address the still-unresolved aspects of the Matroska container and the FFV1 video codec.

DEFINITIONS

I describe *film* as having single-image-based content, mainly represented in the RGB or R'G'B' colour space at 4:4:4 chroma sampling, and, at present, usually stored in a folder: for example, TIFF files in a folder, DPX files in an MXF container, and JPEG 2000 files in an AXF container.[4] I call *video* stream-based content, mainly in the colour space $Y'C_BC_R$ at 4:2:2 chroma sub-sampling, currently often stored uncompressed in either an MOV (QuickTime), an AVI, or an MP4 container.[5] In practice, the choice of container does not matter, because only the file header (and possibly the file footer)

1. <https://mediaarea.net/MediaConch/2016/07/26/No-Time-To-Wait-Preservation-FFV1-Matroska-Symposium/>.
2. <http://www.the-reel-thing.org/program-abstracts-4/>.
3. <https://reto.ch/training/2016/20161122.pdf> (in German).
4. The prime (') indicates that the value is gamma-corrected, i.e., adapted to the human eye and not to physical reality. It allows for the same numbers of steps on the dark side as on the light side of the so-called medium grey. Note that this is not an apostrophe.
5. $Y'C_BC_R$ is sometimes written YCbCr, and, often incorrectly, YUV which is actually the colour space used for PAL video and not for digital video.

are different in different containers, while the stream is bit-by-bit identical for the full image content. The file can be trans-muxed (i.e., the file is de-muxed and then re-muxed) very quickly, because transcoding (i.e., extremely time-consuming decoding and re-encoding) of the file's content is not required. Re-wrapping can be easily done if needed, e.g., during a data migration, without any additional cost. Therefore, the passionate discussions about the best container choice – MP4, AVI or MOV – should now be relegated to the past. The important factor for an archive is that $Y'C_BC_R$ 4:2:2 content is often used by the video and broadcast community to achieve the best quality of high-level professional production and post-production. An archive should therefore be able to provide historic content in a format that commercial clients can use, perhaps without any transcoding.

STANDARDISATION

Standardisation is fundamental to every technical field. Different bodies have recently standardised, or are currently standardising file formats that closely relate to the audiovisual preservation field:

The Society of Motion Picture and Television Engineers (SMPTE) has standardised the CineForm or VC-5 and the ProRes video codecs. ProRes has been one highly relevant *de facto* standard in post-production, but Apple will soon stop supporting QuickTime on Windows platforms – and probably on macOS in the not-too-distant future. While the popularity of GoPro's CineForm/VC5 seems to be increasing at present, sadly, the published standard does not contain all the relevant information needed to implement the codec.[6]

A group of scholars, led by the University of Basel in Switzerland, is preparing a proposal for an archival version of the popular TIFF file format, which they plan to submit to the International Organization for Standardization (ISO) for approval and inclusion. The format was initially called TIFF/A, like PDF/A, but Adobe, who claim some rights in TIFF, would not agree to this; the new format is therefore called TI/A for Tagged Image for Archival.[7]

The standardisation of EBML, Matroska (MKV), FFV1, and FLAC is currently being undertaken by the IETF's CELLAR group (see below). This is the main topic of my paper.

WHAT DO ALL THESE ACRONYMS MEAN?

The Internet Engineering Task Force (IETF)[8] is the body that governs the internet from the technical point of view, in particular, the TCP/IP Internet protocol suite. It develops and promotes voluntary internet standards, the so-called Request for Comments (RFC). It is an open standards organisation, with no formal membership or membership requirements. All participants and managers are volunteers, though their work is usually funded by their employers or by sponsors.

One of IETF's numerous working groups is called Codec Encoding for LossLess Archiving and Realtime transmission (CELLAR). This group is attempting to standardise a coherent set of open, transparent, self-descriptive, and lossless formats,[9] an important mission for the open-source community to undertake for the archival world. CELLAR is standardising four different elements.

The first element is the Extensible Binary Meta-Language (EBML).[10] You may think of it as a binary equivalent to XML, which allows the encoding of bitstreams instead of bytes, like Unicode characters for XML.

The second element of CELLAR's standardisation work is Matroska,[11] a container or wrapper with the file extension ".mkv". It can contain, among many other elements and possible formats, an image stream encoded by the FF Video Codec 1 (FFV1),[12] and one or more audio streams encoded by the Free Lossless Audio Codec (FLAC).[13] Matroska is actually a fork

6. <https://kws.smpte.org/kws/public/projects/project/details?project_id=15>, and <https://kws.smpte.org/kws/public/projects/project/details?project_id=278>.
7. <http://ti-a.org>.
8. <https://www.ietf.org>.
9. <https://datatracker.ietf.org/wg/cellar/>.
10. <https://github.com/Matroska-Org/ebml-specification>.
11. <https://github.com/Matroska-Org/matroska-specification>.
12. <https://github.com/ffmpeg/ff1>.
13. <https://xiph.org/flac/format.html>.

of a unfinished container called Multimedia Container Format (MCF). Google's WebM container is technically a fork – mathematically a subset – of Matroska.

The third element is FFV1, a simple and efficient lossless intra-frame-only video codec. This content can be compressed losslessly, needing roughly 40% of the uncompressed storage space, using the FFV1 video codec. This is a similar compression rate to that achieved by the JPEG 2000 video codec, but FFV1's compression time is less than that of JPEG 2000 because of its much simpler compression algorithm. This is true for both the stream-based $Y'C_BC_R$ 4:2:2 content, as used in the video and broadcast world, and the single-image-based R'G'B' or RGB 4:4:4 linear or logarithmic content, as used by the cinema industry.[14]

The fourth element is FLAC, an audio codec. While the Broadcast WAVE Format (BWF) is a good archival choice for sound, FLAC provides lossless compression as well, though this is less relevant for sound than for image because of their very different sizes. During CELLAR's first year of activity, nothing has been done on FLAC standardisation, but, as Google Chrome has just (January 2017) announced that it will support FLAC, I imagine this will become a priority during the year.

When standardised by the IETF, this suite of objects provides the key to a non-proprietary, trans-generational, functional, and stable deep-storage schema for data that can exist as fixed media (tape, HDD or SSD), or on servers, or in a complex and multi-level environment such as that known as "cloud storage". It allows for the deployment of archive data across many storage environments, and through generations of migration with a high degree of confidence and interoperability.

WHAT IS INSIDE MY DPX?

One of the current, so-called "raw" formats for scanner output is Digital Picture Exchange, or DPX. Kieran O'Leary offers an in-depth discussion of many aspects of the current situation in an outstanding blog: *Introduction to FFV1 and Matroska for Film Scans*.[15] I would mention here only the real advantage of storing CRC-32 checksums for every slice of frame that FFV1 provides over DPX or TIFF, which do not contain any embedded fixity information. This is a key factor that allows institutions with only a small infrastructure to achieve professional preservation of audiovisual files. O'Leary also notes that FFV1 does not encode or retrieve (decode) all metadata correctly at present. This is partly related to the fact that DPX can code the RGB information it holds in many different ways – which means the archive really must know what is inside the DPX files.

DPX is a strange construct, an umbrella that groups together many different encodings, which derives from the Cineon format developed by Kodak for digital intermediate workflow in the early 1990s. At that time, films were shot on analogue film and screened in the same format. Cineon was designed for an interim step, i.e., for post-production purposes, not for conservation. Therefore a .dpx file may contain different encodings of RGB-based information:

- log neg encoding
  Examples: Cineon Printing Density (CPD/DPX), ARRI log C,
- log RGB encoding or quasi-log encoding
  Examples: FilmStream (log_{60}), SI-log (Silicon Imaging, log_{90}), ARRI log F, Panalog (Panavision), S-log (Sony), REDlogFilm,
- gamma encoding or power function encoding
  Examples: sRGB, CineGamma, Film Rec (Panasonic), hyper-gamma,
- scene-linear encoding
  Example: ACES.

As O'Leary says, at present, it is very hard – maybe even impossible – for an archivist to know exactly what is inside the different DPX files, from different sources, held by his/

14. For R'G'B' or RGB 4:4:4 at 16 bit per colour channel, the compression rate could be a little improved. Currently the implementation of Bayer-filter-based formats is just an idea; nobody is actively working on it.

15. <https://kieranjol.wordpress.com/2016/10/07/introduction-to-ffv1-and-matroska-for-film-scans/>.

her archive. Production and post-production processes don't give high priority to technical metadata, perhaps because it is not particularly relevant if the colourist has to tweak the controls a little during the creative phase. It is entirely the opposite for the archivist, of course: it is crucial to preserve the document as it is, without any additional creative work.

ARCHIVE MASTER AND MEZZANINE

The Matroska container and the FFV1 video codec are good choices for single-image-based content when making archive masters. Often, a resolution of 2K, or sometimes 4K, an RGB colour space, the 4:4:4 chroma sampling, and a bit-depth of 16 bit per colour channel are canonical choices. For stream-based content, the Matroska container and the FFV1 video codec are also good choices for the archive master. A resolution of HD (with pillarboxing or letterboxing if required), in general, the $Y'C_BC_R$ colour space, the 4:2:2 subsampling, and a bit-depth of 10 bit are usually considered best practice.

The Matroska container can also be used for audio, with FLAC as the audio codec. Good parameters are a sample rate of 96 kHz for preservation and mezzanine, and 48 kHz for access,[16] with quantisation of 24 bit for preservation and 16 bit for access. The advantages are having one container format for both single-image-based and stream-based content. Unfortunately, it's too early to recommend the same format for both the archive master and the mezzanine, because, though this may change in the near future, at present, FFV1 is natively supported by only a few applications.

ACCESS FORMAT

The Matroska container is currently not popular enough for it to be recommended for access. While Matroska's subset WebM is being used more and more in modern browsers, it needs the V9 video codec. In practice, however, MP4 is currently the better choice. An HD resolution (with pillarboxing or letterboxing if necessary), can be used for screening on a television or computer monitor. The "natural" video codec would be H.264, encoding $Y'C_BC_R$ with a 4:2:0 chroma subsampling for the image.[17] Unfortunately, AAC (Alternative and Augmentative Communication) is the only audio codec permitted by the MP4 container. We recommend a sample rate of 48 kHz and a quantisation of 16 bit.

OUTLOOK

Though some issues remain unresolved, Matroska, with FFV1 (and FLAC), is on the way to becoming a solid alternative – especially for small archives or archives with extremely limited resources – for preservation masters and mezzanine files. It is still too early to recommend a change for access.

Both SMPTE and the Library of Congress are evaluating data implementation to accomplish the same goals. It is important for the entire community of archives, from the largest state institutions and media companies to the most modest local repositories, to understand the economic and technical value that collective, open-source solutions can offer. We are designing and implementing systems that will retain data over timespans substantially longer than that of the life of motion picture film.

The author wishes to acknowledge the help provided by Kieran O'Leary and Adrian Wood.

16. I don't believe the so-called "CD quality" at 44.1 kHz to be a good choice. Its storage economy is minimal, while its sound quality is significantly diminished.

17. While the H.264 codec's definition allows uncompressed coding, as far as we know, these files can only be handled by FFmpeg-based players. We therefore suggest some slight compression.

fr

Beaucoup d'archives rêvent depuis longtemps d'un format unique permettant une sauvegarde optimale des films quel que soit le support d'origine, pellicule ou vidéo. Aujourd'hui, les contours de cette solution se précisent, et ceci dans un contexte libre et ouvert.

Différents organismes ont récemment standardisé des formats de fichiers pour contenus audiovisuels, ou s'y attèlent. L'un d'eux travaille ainsi sur le conteneur Matroska (MKV) et le codec FFV1, qui permet de comprimer sans perte tant l'image $Y'C_BC_R$ 4 :2 :2 de la télévision que l'image R'G'B' ou RGB 4 :4 :4 du cinéma.

Kieran O'Leary propose un état des lieux de la situation actuelle, mettant en exergue certaines caractéristiques intéressant tout particulièrement les archives audiovisuelles, notamment les sommes de contrôle pour chaque photogramme intégrées au flux, qui permettent d'en vérifier aisément l'intégrité. Il souligne en outre que les métadonnées ne sont pas toujours stockées correctement dans les DPX, qui est un format source pour de nombreux scanners.

Tous les problèmes n'ont pas été résolus, mais Mastroska et FFV1 sont en passe de s'imposer comme une alternative solide pour la réalisation de masters à fin d'archivage, en particulier pour des petites archives disposant de ressources limitées. En revanche, il apparaît prématuré de recommander ce choix aussi comme mezzanine, puisqu'à l'heure actuelle, FFV1 n'est supporté nativement que par un petit nombre de logiciels.

es

Muchos archivos sueñan desde hace mucho tiempo con un solo formato que permita obtener copias de seguridad óptimas independientemente del soporte original, ya sea película o vídeo. Hoy en día, la configuración de esta solución se va precisando y además en un contexto libre y abierto.

Varios organismos han estandarizado recientemente sus formatos de archivos para contenidos audiovisuales, o lo están considerando. Uno de ellos trabaja también con el contenedor Matroska (MKV) y el códec FFV1, que permite comprimir sin pérdida tanto la imagen $Y'C_BC_R$ 4:2:2 de televisión como la imagen R'G'B 'o RGB 4: 4: 4 de cine.

Kieran O'Leary ofrece una visión general de la situación actual, destacando algunas características que interesan particularmente a los archivos audiovisuales, incluyendo las sumas de comprobación para cada fotograma integrado en el flujo, lo que permite comprobar la integridad con facilidad. Además, subraya que los metadatos no siempre se almacenan adecuadamente en DPX, que es el formato origen de muchos escáneres.

A pesar de que no todos los problemas hayan sido resueltos, Mastroska y FFV1 están a punto de imponerse como alternativa sólida para la realización de copias maestras para archivos, especialmente para aquellos pequeños archivos con recursos limitados. Sin embargo, parece prematuro recomendar esta elección también como opción intermedia, ya que en la actualidad, FFV1 sólo puede ser soportado por un pequeño número de softwares.