Audio/Video preservation: Challenges, oddities and lessons learned

Peter Bubestinger

02. Jun 2014

About me

Peter Bubestinger

- Studied Media-Computerscience at the TU-Vienna
- Practical experience with professional archives since 2002:
  - ORF (National broadcaster, Austria)
  - VoV (National broadcaster, Vietnam)
  - RTV (National broadcaster, Slovenia)
  - SRTC (National broadcaster, Sudan)
  - Fonoteca Nacional (Mexico)
  - Memnon Archiving Services (Belgium)
  - SRF (Sweden), YLE (Finland), SRR (Romania), ...
- Since 2010: engineer/developer in the video department of the Austrian Mediathek

Disclaimer

For your consideration:

This is an overview, based on my current knowledge/experience.
Please: comment, disagree and question, if you feel to do so!

Disclaimer

For your consideration:

This is an overview, based on my current knowledge/experience.
Please: comment, disagree and question, if you feel to do so!

Not only me, and not only in this lecture :)

Digital video

Digital Video

Why digital video is more complicated than audio. . .

Audio:

- Resolution: Samplerate, bits-per-sample, channels
- Uncompressed: ca. 16 MiB/min. (48kHz/24bit/stereo)
- Archiving format: PCM in WAV

Video:

- “Resolution”?
- Uncompressed: ca. 1.8 GiB/min. (PAL SD, YUV 4:2:2, 8bpc)
- Archiving format: ?

Choose your destiny

The "digital video trinity":

- Container
- Videocodec
- Audiocodec

Choose your destiny

The "digital video trinity":

- Container
- Videocodec
- Audiocodec

Answers like "The videos are in flash/AVI/quicktime format" or "The camera stores MP4 files" usually only tell you which container is used.

Video "resolution"
There's more to consider...

Multiple factors:

- Pixel resolution (straight/anamorphic)
- Group Of Pictures (GOP)
- Bitrate
- Framerate (PAL, NTSC, film, "esoteric", etc.)
- Scanning mode (interlaced/progressive)
- Colorspace (YUV, RGB, XYZ)
- Subsampling (4:4:4, 4:2:2, 4:2:0, etc)
- Bits-Per-Component (bpc)

Usually, digital cameras (even HD) use YUV, 4:2:0 subsampling and 8bpc linear.
Everything else is very often a "hardly chartered area".

Which codec?

Codecs - Compression types
lossy vs. lossless vs. uncompressed

Current state for video:

- Current default is lossy (even in professional production). What about audio production?
- Challenges currently holding back from lossless/uncompressed:
  - Size
  - Performance
  - Format support/adoption
  - Small userbase
- For archiving, lossless is always to be preferred
- Uncompressed is lossless, but lossless is not necessarily uncompressed. Uncompressed is *huge*...
- Widespread misassumption, or wrong wording: "*compressed = lossy*"

Videocodecs
Overview

Lossless:

h264-lossless: h264 also has a lossless mode. h264-lossy is widely supported, but lossless is hardly implemented/tested.

JPEG2000-lossless: JPEG2000 can also do lossless. Also less widely implemented/tested.

FFV1: "FFmpeg Video codec 1". Pure lossless-Codec. Completely open format. Not (yet) standardized.

Dirac: BBC's lossy/lossless wavelet video codec. Free and Open Standard.

HuffYUV: Huffman-based. Very fast, but compression not so good.

Uncompressed: Different codec implementations per color- and subsampling-variations

`http://download.das-werkstatt.com/pb/mthk/info/video/comparison_video_codecs_containers.html`

Which container?

Unsettled format question

Different requirements:

- Different focus/demands:
  Broadcast, production, preservation
- "*Minimalistic standards*" vs. "*All-In-One*"
- "*The large file*" vs. *segmentation*.
  Use cases to consider:
  - partial file retrieval (over network, from tape)
  - integrity checks and data restoration
- Metadata:
  - Schemas, formats and "All-In-One" file or not?
  - Updating the AIP (e.g. augmenting metadata)

Industry proposed standard:
JPEG2000-lossless or Uncompressed in MXF.

Unsettled format question
Jack of all trades - master of ~~none~~ some?

JPEG2000/MXF

- Both formats are Open Standards
- "All-In-One" approach

Let's reconsider if "All-In-One" is really always simpler/better...

Unsettled format question

Jack of all trades - master of ~~none~~ some?

The "Spork" = Spoon + Fork (+ Knife):

Source: `http://www.lightmyfire.com/products/products/spork/spork-original.aspx`

Unsettled format question

Jack of all trades - master of ~~none~~ some?

Wenger "Giant":

Source: http://www.amazon.de/Wenger-Schweizer-Offiziersmesser-Messer-Schatulle/dp/B000R0JDSI

Theory and practice

- Why MXF is a bit tricky:

Theory and practice

- Why MXF is a bit tricky:
  - Imagine XML.
    Imagine interoperability without schema definitions.
    Now imagine it being binary, not text.

Theory and practice

- Why MXF is a bit tricky:
  - Imagine XML.
    Imagine interoperability without schema definitions.
    Now imagine it being binary, not text.
  - Vendor-specific extensions and embedded metadata.

Official standards
Things you should know

Theory and practice

- Why MXF is a bit tricky:
  - Imagine XML.
    Imagine interoperability without schema definitions.
    Now imagine it being binary, not text.
  - Vendor-specific extensions and embedded metadata.
  - Verify standards-compliance:
    Which MXF profile?
    Which tools?
    What about the bitstreams inside?

Theory and practice

- Why MXF is a bit tricky:
  - Imagine XML.
    Imagine interoperability without schema definitions.
    Now imagine it being binary, not text.
  - Vendor-specific extensions and embedded metadata.
  - Verify standards-compliance:
    Which MXF profile?
    Which tools?
    What about the bitstreams inside?
  - BBC wrote 3 MXF-to-MXF converters for in-house demands.

Official standards
Things you should know

Theory and practice

- Why MXF is a bit tricky:
  - Imagine XML.
    Imagine interoperability without schema definitions.
    Now imagine it being binary, not text.
  - Vendor-specific extensions and embedded metadata.
  - Verify standards-compliance:
    Which MXF profile?
    Which tools?
    What about the bitstreams inside?
  - BBC wrote 3 MXF-to-MXF converters for in-house demands.
  - Cube-Tec's "MXF Legalizer": 30.000 EUR/license

Official standards
Things you should know

Theory and practice

- Why MXF is a bit tricky:
  - Imagine XML.
    Imagine interoperability without schema definitions.
    Now imagine it being binary, not text.
  - Vendor-specific extensions and embedded metadata.
  - Verify standards-compliance:
    Which MXF profile?
    Which tools?
    What about the bitstreams inside?
  - BBC wrote 3 MXF-to-MXF converters for in-house demands.
  - Cube-Tec's "MXF Legalizer": 30.000 EUR/license
  - FADGI experts group working on MXF profile since around 2010.

Official standards
Things you should know

Theory and practice

- Why MXF is a bit tricky:
  - Imagine XML.
    Imagine interoperability without schema definitions.
    Now imagine it being binary, not text.
  - Vendor-specific extensions and embedded metadata.
  - Verify standards-compliance:
    Which MXF profile?
    Which tools?
    What about the bitstreams inside?
  - BBC wrote 3 MXF-to-MXF converters for in-house demands.
  - Cube-Tec's "MXF Legalizer": 30.000 EUR/license
  - FADGI experts group working on MXF profile since around 2010.
- Why JPEG2000-lossless is also tricky:

Official standards
Things you should know

Theory and practice

- Why MXF is a bit tricky:
  - Imagine XML.
    Imagine interoperability without schema definitions.
    Now imagine it being binary, not text.
  - Vendor-specific extensions and embedded metadata.
  - Verify standards-compliance:
    Which MXF profile?
    Which tools?
    What about the bitstreams inside?
  - BBC wrote 3 MXF-to-MXF converters for in-house demands.
  - Cube-Tec's "MXF Legalizer": 30.000 EUR/license
  - FADGI experts group working on MXF profile since around 2010.
- Why JPEG2000-lossless is also tricky:
  - Who *is* using JPEG2000? (e.g. DCP)

Official standards
Things you should know

Theory and practice

- Why MXF is a bit tricky:
  - Imagine XML.
    Imagine interoperability without schema definitions.
    Now imagine it being binary, not text.
  - Vendor-specific extensions and embedded metadata.
  - Verify standards-compliance:
    Which MXF profile?
    Which tools?
    What about the bitstreams inside?
  - BBC wrote 3 MXF-to-MXF converters for in-house demands.
  - Cube-Tec's "MXF Legalizer": 30.000 EUR/license
  - FADGI experts group working on MXF profile since around 2010.
- Why JPEG2000-lossless is also tricky:
  - Who *is* using JPEG2000? (e.g. DCP)
  - Who *is* doing lossless?

Official standards
Things you should know

Theory and practice

- Open Standards are a *very* good thing!
- Importance of reference implementations.
  - AMIA: "You cannot force vendors to adhere to specifications."
  - Vendor: "We only follow customer demands. It's *their* duty to verify standards compliance."
  - Standards bodies (SMPTE, EBU, ISO, etc):
    Their experts are often vendors
- Real world examples:
  - Why *remuxing* makes sense (Often A/V sync issues)
  - NTSC: 30000/1001 vs. 2997/100 vs. 29.97 fps
  - MPEG-4 from smartphone: 38.462 fps
  - ORF: Needed to convert IMX/MXF to IMX/MXF
  - Samma: non-interoperable MXF/JPEG2000-lossless files
- AVI/MOV/MKV:
  - Documented standards or proprietary formats?

Archiving

The 21st century
Same challenges, different conditions

The 2 "*Triangles of Trickyness*" with digital A/V formats:

1: End-user (archive):

decision makers / users / engineers

- Decision makers don't know/care about the technical details.
- Users just want their stuff to work. Now!
- Existing engineering staff: great expertise and practical experience with (mostly) pre-digital A/V tech-details.

It's very often exactly the tech-details that make a difference for long-term preservation.

2: Available industry solutions:

decision makers / salesmen / developers

- Decision makers don't know/care about the technical details.
- Salesmen want to sell their products. Now!
  Remember "HD-ready"?
- Developers great with digital, but not necessarily practically experienced with professional A/V media.

In these triangle-situations: Who shall decide what to use - or question/improve it?
Are there conflicts of interests?
Oh, and: "Yes, it's highly political."

Data format(s) for long-term preservation

Format requirements

Wishlist:

- Lossless (image/audio)
- Preserve colorspace and subsampling exactly ("pix_fmt")
- Preserve metadata
- In any case an open format. Standardized if possible.
  Also with standards: Pitfalls with proprietary, closed implementations (hardware, software, camera).
- Sustainability / accessability
- Archiving the source code:
  "git clone" = archiving replayer + schematics + revisions
- The simpler, the better.

Data format(s) for long-term preservation
What can I still open in 'x' years?

Decisions:

- "All-In-One" file, or a "Folder Package"?
- Format support/obsolescence?
- Requirements (hardware, software)?
- Dependencies (hardware, software, licenses)?
- Storage costs (including backups)?

Data format(s) for long-term preservation
What can I still open in 'x' years?

Decisions:

- "All-In-One" file, or a "Folder Package"?
- Format support/obsolescence?
- Requirements (hardware, software)?
- Dependencies (hardware, software, licenses)?
- Storage costs (including backups)?

Pragmatic solution for long-term video archiving at the moment:
FFV1/PCM in AVI/MOV/MKV.

Data format(s) for long-term preservation
FFV1 lossless video codec

FFV1 history and facts:

- Originated in FFmpeg in 2006
- Inventor: Michael Niedermayer (Austria)
- All implementations are free and open
- Compression almost equal to JPEG2000-lossless
- Better interoperability than JPEG2000-lossless
- Realtime on commodity hardware:
  - Way faster than wavelet compression
  - SD PAL in realtime (single-threading) - 2010
  - Full-HD in realtime (multi-threading) - 2013
- Embedded CRCs
- Not an official standard (yet)

Data format(s) for long-term preservation
What can I still open in 'x' years?

Practical experiences:

- Closed vs. Standard vs. "Just Open"
  - Open Standards are a very good thing!
    If done correctly.
  - Interesting: Files created by open an implementation can practically be more sustainable than files produced with closed implementations.
- Minimalistic standards:
  - As complex as necessary
  - As simple as possible
- Robustness and longevity of "textfiles"
  - binary vs. text
  - XML is also text

Standing the test of time

Standing the test of time
What can we learn from other areas?

What do you need to. . .

- . . . access old formats?
- . . . get old gear running?
- . . . get old software running?

Standing the test of time
What can we learn from other areas?

What do you need to. . .

- . . . access old formats?
- . . . get old gear running?
- . . . get old software running?

Electronics/IT is a fast-moving domain, where “long-term” decisions are hard, having average product life-cycles of 3-5 years.

Standing the test of time
What can we learn from other IT areas?

Some IT-Examples:

- Amstrad PCW (Joyce) - (1985)
  - Emulator available, but software still under vendor-lock.
- ScummVM
  - Ever thought about how to show/play old computer games you loved?
  - Archiving software is a present and upcoming challenge :)
- Unixoid OS (1969)
  - modular, generic design: Still applicable in state-of-the-art environments. After 45 (!) years.
- WRT54GL (2005)
  - Show PDF of Website
  - One result: OpenWRT as router OS
- Rockbox
  - Support for more (future?) formats - even lossless.

Standing the test of time
What can we learn from other IT areas?

(In)dependence:

- Vendor neutrality. Avoid "vendor lock-in" wherever possible.
- Adaptations/fixes/improvements/implementations:
  - Who decides or is able to make things to do what you need?
  - Future consideration: vendor support/existence/interest
  - Preservation = infinite migration
- What are our options?

Standing the test of time
What can we learn from other IT areas?

Free Software (aka "Open Source"):

- The freedom to *use, study, share and improve*:
  Anything *not* to want?
- It's *not* about being gratis.
- The perfect conditions for long-term sustainability:
  Virtual immortality of tools and formats.

Standing the test of time
What can we learn from other IT areas?

Free Software (aka "Open Source"):

- The freedom to *use, study, share and improve*:
  Anything *not* to want?
- It's *not* about being gratis.
- The perfect conditions for long-term sustainability:
  Virtual immortality of tools and formats.

Archiving the replayer:

- Source code = schematics (building components included)
- "svn checkout", "git clone"
- Same applies to archiving your working environment
- License matters. Even if you have the source code.

The Austrian Mediathek
A real world example

Current solutions:

- Audio: PCM uncompressed WAV (+BWF metadata)
- Video: DVA-Profession, DVA-Fidelity,
  Segmented FFV1/PCM/AVI + XMLs + images
- Metadata: LoC XML profiles in METS container
- Paid software development/improvement:
  - Still cheaper/easier to use and maintain, than using closed/proprietary products
  - Solutions available for anyone (Free Software license)
- White-list for born-digital formats
- Vendor-neutral, open, managed storage
  (GNU/Linux + SoftRAID + GlusterFS)

Storing the data

Long term preservation

Most of this applies to all kind of digital files in general:

- Storage media:
  - CD/DVD/BluRay is *not* a good physical carrier option for long-term preservation
  - Harddisks: type, Hard-/Soft-RAID, filesystem, clustering?
  - Tapes: type, drives, filesystem?
  - Good to keep copies on different kinds of media (e.g. tape+disk)
- Checksums:
  - File checksums
  - Segment checksums
  - Content checksums
  - "CV-Files": Keeping track of changes
- Digital inventory:
  - Checksum verification
  - LoC's BagIt

Checksums + Digital Inventory

User friendly tip...

Library of Congress' "BagIt":

- Designed to preserve/validate file- and folder-integrity during data transfer
- Works with any kind of file/folder structure/format
- Purely textfile-based

`http://en.wikipedia.org/wiki/BagIt`

Tool: "Bagger"

- Commandline + GUI (Java)
- Cross-Platform: Linux, Mac, Win

`http://sourceforge.net/projects/loc-xferutils/files/loc-bagger/`

Questions?
Comments?

Thank you very much for your attention!

Some rights reserved...

This presentation is available under a Free License:
Creative Commons Attribution Share-Alike
(CC-BY-SA)

Contact:

- Österreichische Mediathek: www.mediathek.at
- Peter Bubestinger: peter.bubestinger@mediathek.at