

Topic 1 - Digital Files

Peter Bubestinger

File Format Considerations

The Goal



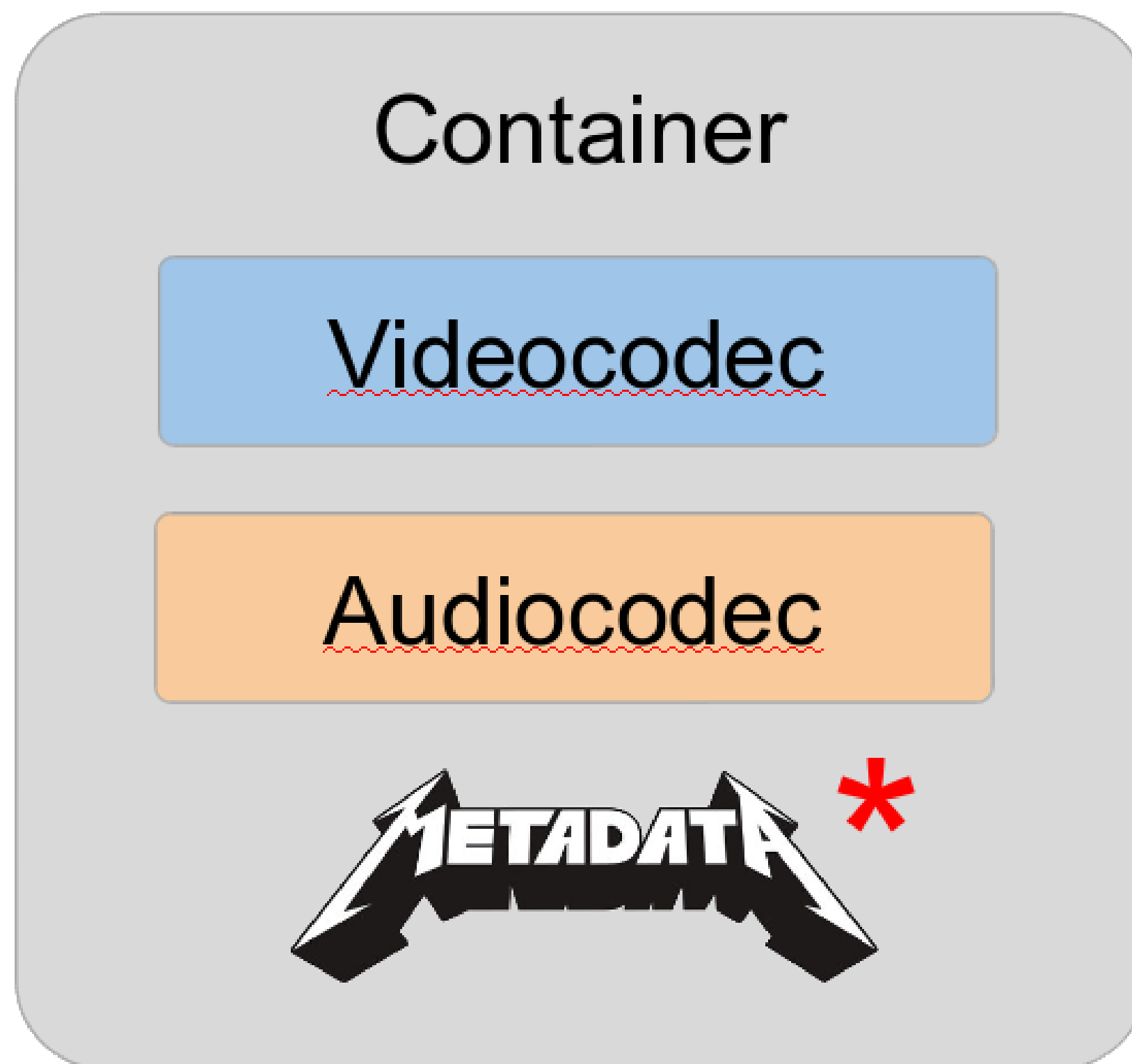
* Best quality * Preserve original properties * Last forever * Lowest size * Fast and easy to open/use * +cherries & ice cream on top * ...

Data Structure

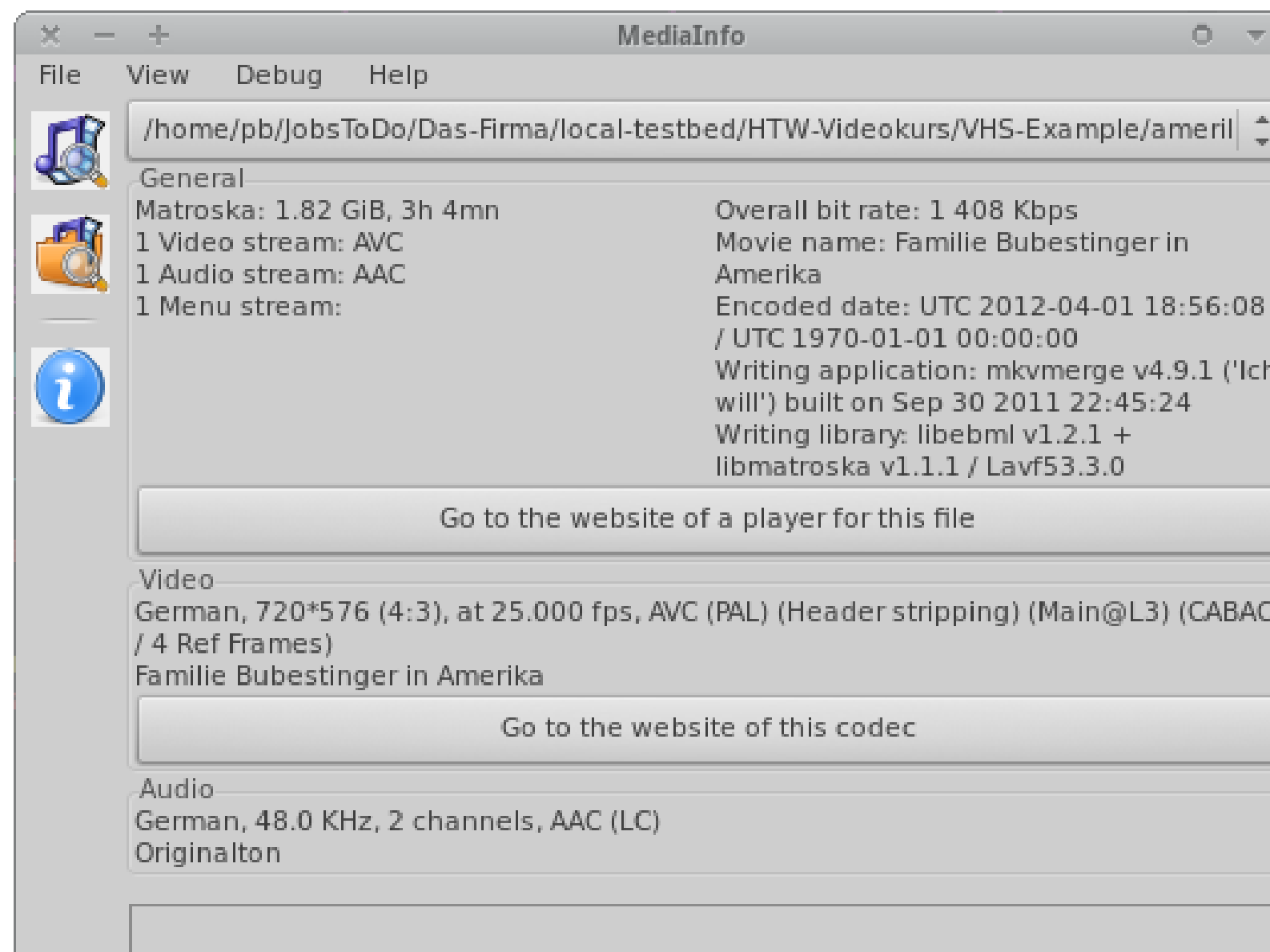
Subchunk1
Size AudioFormat NumChannels Format Subchunk1
ID

- Container
- Video
- Audio

Digital Video Trinity



What's inside?



Characteristics

Image

	File 1	File 2	File 3
Format	TIFF 6.0	TIFF 6.0	TIFF 4.0
Colorspace	RGB	CMYK	Grayscale
DPI	600 dpi	150 dpi	150 dpi
Resolution	4328 x 2979px	1024 x 768	1024 x 768

Characteristics

Audiovisual

	File 1	File 2	File 3
Format	MOV	MOV	MOV
Resolution	720 x 576px	1920 x 1080	640 x 480
FPS	25	24	29.97
Samplerate	48 kHz	48 kHz	44.1 kHz
Channels	Stereo	Surround 5.1	Mono

Data rate / Filesize

Type	Duration	Bitrate	Filesize
Video	1 hour	210 Mb/s	92 GB
		50 Mb/s	22 GB
		25 Mb/s	11 GB
		1,5 Mb/s	1 GB
Audio	1 hour	4,6 Mb/s	2 GB
		128 kb/s	56 MB

Different Formats, different use cases

- Preservation
- Mezzanine
- Access

Different Formats, different use cases

- **Preservation:** Stand the test of time.
Highest original quality.
- **Mezzanine:** For daily work.
High quality.
- **Access** For quick and easy access.
Quality not necessarily best/high.

Examples: Video

- **Preservation:**

* Uncompressed * FFV1 * J2K-lossless * ...

- **Mezzanine:**

* ProRes * H.264 * DVCPRO50 * ...

- **Access:**

* MP4 * WebM * DVD * BluRay * ...

Speaker notes

I've intentionally used fuzzy wording for access formats, since it better depicts the perception and wording used in daily life.

For Preservation and Mezzanine, I've left out audio- and container-formats, since it would unnecessarily complicate the slide.

Later in this session, we'll get to how to select properties for these use cases, which helps to decide which format to use for which case.

Size (still) matters

Type	Duration	Bitrate	Filesize	Usage
Video	1 hour	210 Mb/s	92 GB	Preservation
		50 Mb/s	22 GB	Preservation
		25 Mb/s	11 GB	Preservation / Mezzanine
Audio	1 hour	1,5 Mb/s	1 GB	Access
		4,6 Mb/s	2 GB	Preservation
		128 kb/s	56 MB	Access

Data rate = Bitrate

- $\text{Mbps} / 8 = \text{MB} / \text{second}$
- $\text{MB/s} * 60 = \text{MB} / \text{minute}$
- $\text{MB/min} * 60 = \text{MB} / \text{hour}$

Speaker notes

If you know the bitrate, you can calculate the size of your files, by multiplying the bitrate by the runtime duration (time).

A fixed bitrate however is only available for:

- constant bitrate (lossy) compression.
- uncompressed.

For lossy compression, bitrate is an encoding parameter, but for uncompressed we'll show later in this session how to calculate the size.

For lossless or variable bitrate, the exact size cannot be pre-calculated - only estimated. The actual size can still vary, since the size of these kind of encodings greatly depends on the content being encoded. Rule of thumb: Less motion = smaller size and vice versa. Noise make files larger.

Definition fuzziness in the preservation community: Some say “Significant” are the properties that must be maintained as-is and kind of “must never change”, whereas others define it as “should be aware of and decided how to deal with them”.

Further properties: * scan type (interlaced / progressive) * field order * color information * ...

Significant properties

- Depends on media type.
- Examples for A/V:
 - resolution
 - framerate
 - aspect ratio
 - colorspace
 - subsampling
 - ...

More: FADGI Significant Properties for Digital Video - DRAFT

Significant properties

Image?

Significant properties

Audio?

Lossy, Lossless, Uncompressed?

How it affects quality and preservation.

Lossy



Speaker notes

Of course this image is exaggerated, but it shows pretty well what lossy compression is and what artefacts a typical MPEG-like compression algorithm produces.

btw: This is a snapshot image of the highest-quality version of the video on the original website (around 2009).

Generation Loss

Speaker notes

Here's an example where one can see the difference between each encoded version in popular broadcast codecs.

Since it's almost certain that there's not "evergreen format", any format will sooner or later have to be migrated to another one. The longer an item is preserved, the more often it will encour such a format change (transcoding). Therefore, generation loss should be considered.

Also: Usage of archive material: What happens if you hand out your material to an editor, say on DVD or a mezzanine format? The editor then edits and exports to another format, the broadcaster or cinema then to another? ...and in the end this work is then also converted to an access format.

And theeeen, years later that access copy is used as a source for a documentary or online edit.

How many generation losses accumulate here? How could they be reduced?

Generally: please try to avoid unnecessary generation losses. Why not? :)

Lossless

“It’s like ZIP for film!”

- No generation loss
- Way larger than lossy
- Smaller than uncompressed

Uncompressed

- No generation loss
- Dead simple (=preserves well)
- The largest possible version
- There's *more than just 1* “uncompressed”

Uncompressed - Think of it as:

1px RGB Image:

RRR GGG BBB AAA

1px YUV Image:

YYY UUU VVV

1 sample audio:

LLLLLLLLL RRRRRRRR

Speaker notes

Image: 8 bits-per-component = 3 Bytes (therefore 3 characters here per component):

- Red, Green, Blue
- YUV
- Alpha

Audio: 8 bits per sample, 2 channels:

- Left
- Right

All you need to know is (somewhat):

- Color model
- Width (=line length)
- Sample size
- Channels

Uncompressed Image

- Width(px) x Height(px)
- x Bits-Per-Pixel(bpp)
- x FPS
- / 8 = **1 second (in Byte)**

Exercise:

- How large is an 8bpc SD PAL minute - with 4 channels audio at SDI standard (48kHz/16bit)?
- Or a 2k 16bpc(!) scan (including 6 channels audio at 48kHz/24bit)?
- Or 74min. of audio CD (red-book standard: 44.1kHz/16bit)?

Uncompressed Audio

- Samplerate x bit-depth
- x channels ^(even if silence!)
- / 8 = **1 second (in Byte)**

Film is currently stored as still image sequence (1 frame = 1 file) in a folder, along with the audio as one or more WAV (PCM) files and optionally metadata sidecar files (XML, etc).

Reels are stored in individual subfolders.

As you can see, I didn't mention which image codec for video, because there is currently still no format option that satisfies the needs of different communities like: archive vs consumer vs production vs broadcast vs cinema.

Question: Where's the difference between digital film and digital video?

Default Formats

Film:

- **Image:** DPX / TIFF files
- **Audio:** PCM in WAV
- **Metadata:** Mostly sidecar, some MD in image files.

Video:

- **Image:** Default = lossy encoding
- **Audio:** production = PCM, consumer = AAC
- **Metadata:** Often embedded. Sometimes sidecar.

Best practices for A/V Formats

- Capture analog video uncompressed (v210) or lossless (FFV1, J2K) to avoid adding digital generation loss.
- Or as fallback option:
At the highest quality (data rate) you can store and manage well over time.
- Capture digital tape in its native format without generation loss (MiniDV, DAT, DigiBeta, etc.)
- Store born-digital files “as original” as possible.
- Audio preservation format is uncompressed WAV (PCM) for analog originals.
- For video container formats, consider using MKV or MOV.
MXF *only* for broadcast.
- Choose formats that can be kept alive (=open & documented)

- What do you think “preserves well”?
 - What to look out for?
 - What to check/ask/know?
 - When to acquire that information?

File formats and preservation implications

Complexity

Speaker notes

Simpler = easier to keep alive, reconstruct or hack.

Good rule: “Minimalistic Standards” * As simple as possible
* As complicated as necessary

Be careful with “one size fits all”: Sporks are good for camping, but there’s a reason why we still have separate tools for the job: spoon, knife and fork.

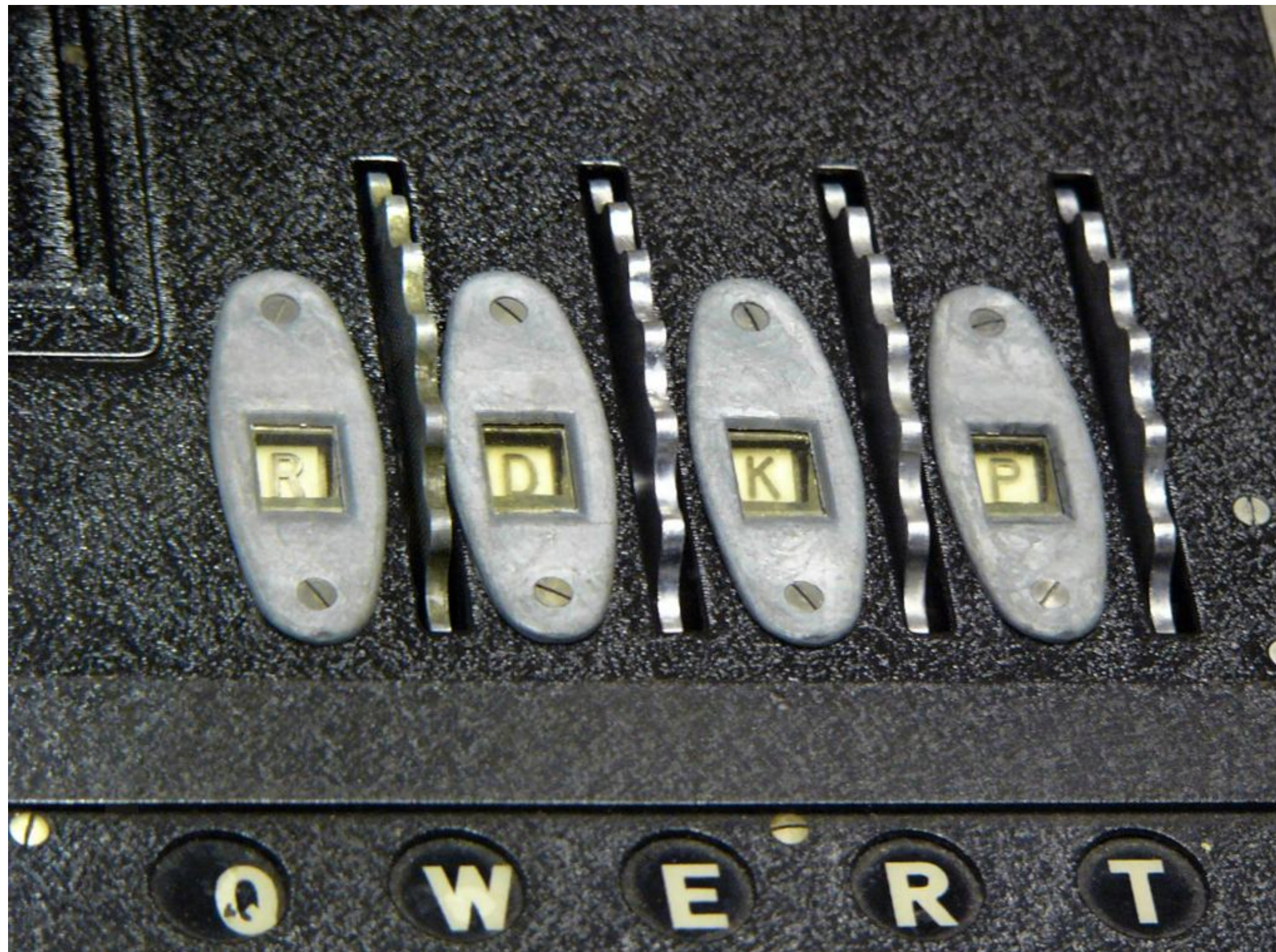
Considerations:

- More features = more complex / chance that only parts of specifications are supported by tool X.
- Can be non-trivial to judge what is “simple” and what is “complex”
- Find the sweet spot for your use case(s).

Format Support

- Popular?
- Documented?
- Well supported?
- Can *you* handle/access it beyond shelf-life?

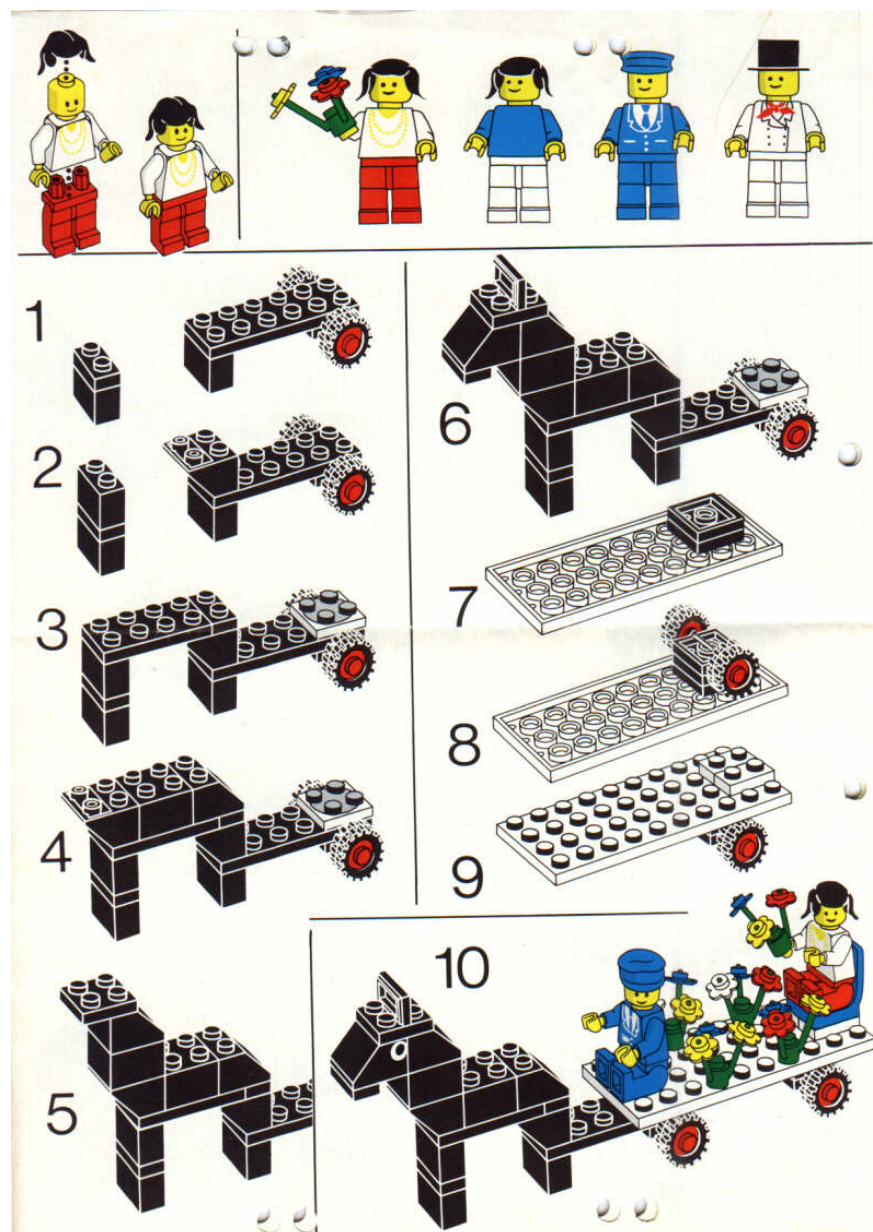
Obsolescence: Open vs Closed



Implementation trumps paper specs.

If the implementation is open: you can fix/improve if necessary. If it's closed/proprietary then there's nothing you can do about it: Black box. Vendor lock-in.

Theory vs Practice



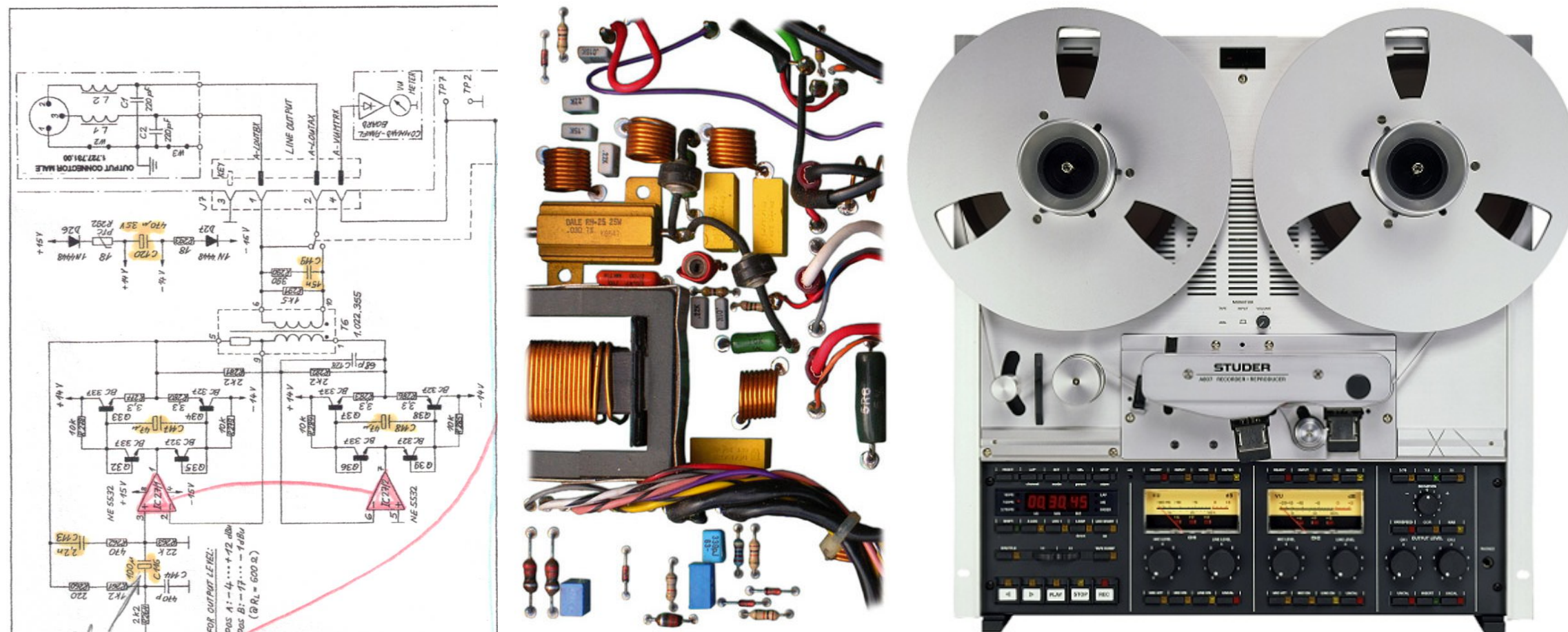
As an example:

- Schematics
- Building components
- +the right to U.S.S.I them.

= “The Immortal Replayer”

Because it can be kept alive, or rebuilt or adapted to future needs or with future technology (whatever there may be).

Schematics



Error resilience



Popular formats

Speaker notes

Let's take a look at pros/cons of some popular (=significantly present in the wild) and their preservability properties.

Containers

- AVI: Audio Video Interleave
- MOV: Quicktime
- MKV: Matroska Video
- MXF: Material eXchange Format
- WAV / RIFF
- MPG, MTS: MPEG Transport Stream

Video Codecs

- H.264 (lossy, lossless, uncompressed)
- H.265
- MPEG-2 (**IMX**, **XDCAM**)
- ProRes
- J2K (lossy, lossless)
- FFV1 (lossless)
- “Uncompressed”

Audio

- AAC
- MP3
- Opus
- PCM
- FLAC

- Data errors: Files are corrupt. These errors may include filenames or filesystem tech-MD.
 - Obsolescence: Not supported anymore by accessible tools. This is not God-given, nor irreversible, unless it's a black-box format. Documentation/schematics are an important game changer!
 - Interoperability issues: If a format is read and written differently by different applications, it might “morph”. This morphed version might work fine with the tools used in a certain environment, but might be completely broken if read/written by another tool that misunderstands the “dialect”.
 - Vendor lock-in: For long-term preservation, vendor- and technology-neutrality is a must: They will come and go, and with lock-in situations Eternal Migration is hindered or even impossible. Format normalization helps here.

Risks to format longevity

- Data errors
- Obsolescence
- Interoperability issues
- Vendor lock-in

Countermeasures?

Comments?
Questions?

Links

- [Primer on Codecs for Moving Image and Sound Archives](#)
- [Hex Editing for Archivists](#)
- [Comparing Video Codecs and Containers for Archives](#)