MediaConch

Implementation and policy checking
on FFV1, Matroska, LPCM, and more

Jérôme Martinez, MediaArea

Experience Workshop - November 2016

PREFORMA

SEVENTH FRAMEWORK PROGRAMME

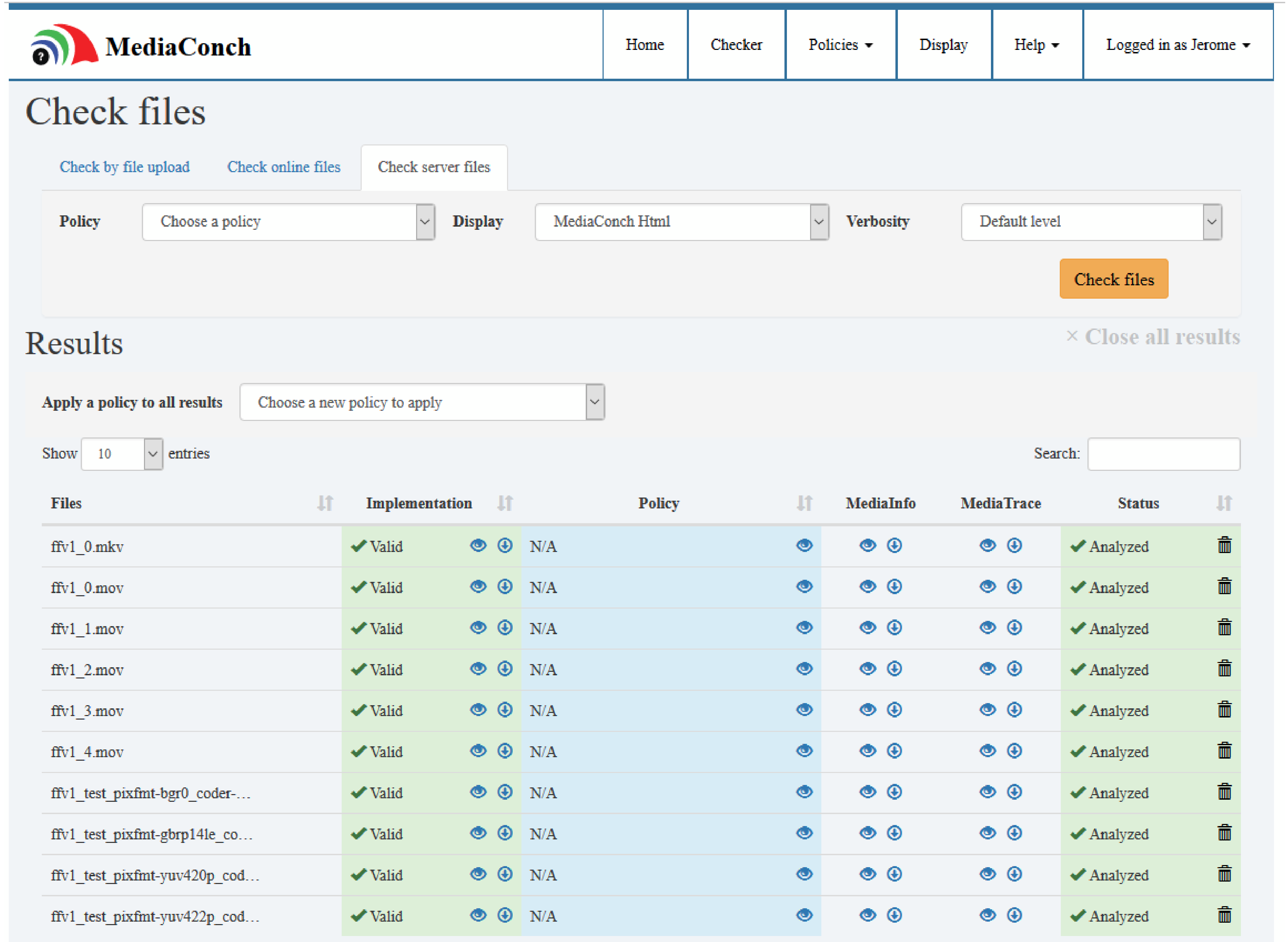
What is MediaConch?

MediaConch is a conformance checker

- Implementation checker
- Policy checker
- Reporter
- Fixer

What is MediaConch?

Implementation and Policy reporter



Check files

Policy: Choose a policy

Display: MediaConch Html

Verbosity: Default level

Results

Apply a policy to all results: Choose a new policy to apply

Show 10 entries

Search:

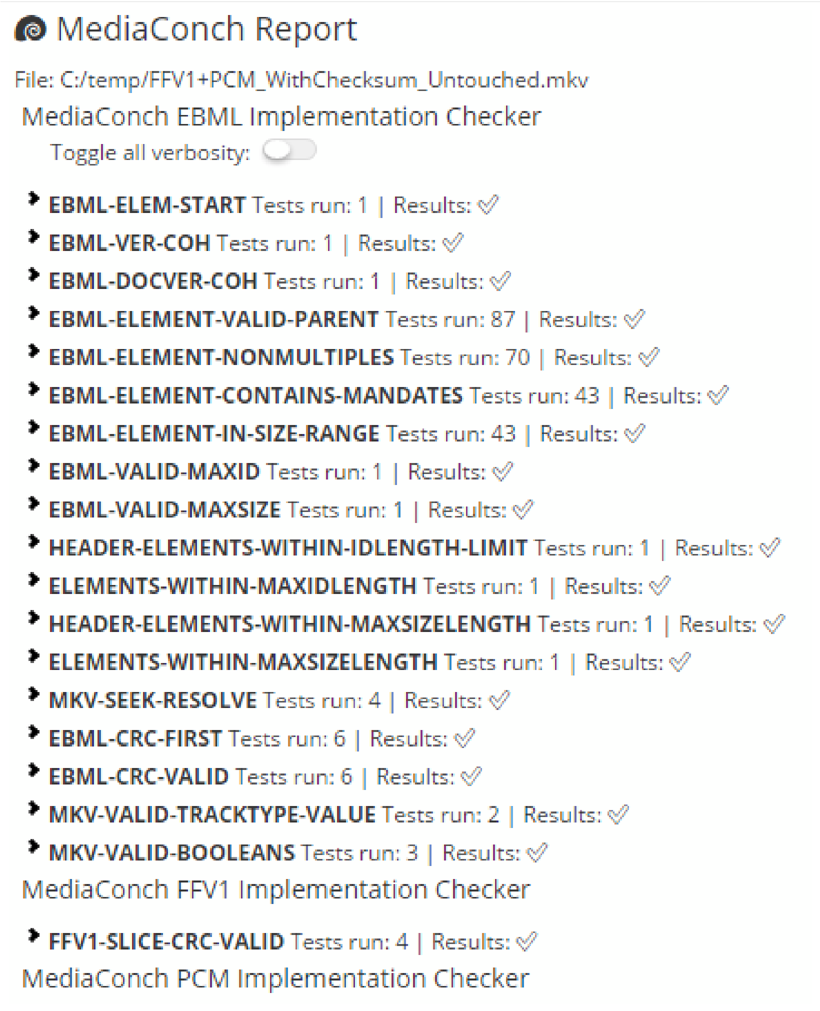
Files	Implementation	Policy	MediaInfo	MediaTrace	Status
ffv1_0.mkv	✓ Valid 👁 ⬇	N/A 👁	👁 ⬇	👁 ⬇	✓ Analyzed 🗑
ffv1_0.mov	✓ Valid 👁 ⬇	N/A 👁	👁 ⬇	👁 ⬇	✓ Analyzed 🗑
ffv1_1.mov	✓ Valid 👁 ⬇	N/A 👁	👁 ⬇	👁 ⬇	✓ Analyzed 🗑
ffv1_2.mov	✓ Valid 👁 ⬇	N/A 👁	👁 ⬇	👁 ⬇	✓ Analyzed 🗑
ffv1_3.mov	✓ Valid 👁 ⬇	N/A 👁	👁 ⬇	👁 ⬇	✓ Analyzed 🗑
ffv1_4.mov	✓ Valid 👁 ⬇	N/A 👁	👁 ⬇	👁 ⬇	✓ Analyzed 🗑
ffv1_test_pixfmt-bgr0_coder-…	✓ Valid 👁 ⬇	N/A 👁	👁 ⬇	👁 ⬇	✓ Analyzed 🗑
ffv1_test_pixfmt-gbrp14le_co…	✓ Valid 👁 ⬇	N/A 👁	👁 ⬇	👁 ⬇	✓ Analyzed 🗑
ffv1_test_pixfmt-yuv420p_cod…	✓ Valid 👁 ⬇	N/A 👁	👁 ⬇	👁 ⬇	✓ Analyzed 🗑
ffv1_test_pixfmt-yuv422p_cod…	✓ Valid 👁 ⬇	N/A 👁	👁 ⬇	👁 ⬇	✓ Analyzed 🗑

Showing 1 to 10 of 16 entries

Previous 1 2 Next

What is MediaConch?

Implementation report:



MediaConch Report

File: C:/temp/FFV1+PCM_WithChecksum_Untouched.mkv

MediaConch EBML Implementation Checker

Toggle all verbosity:

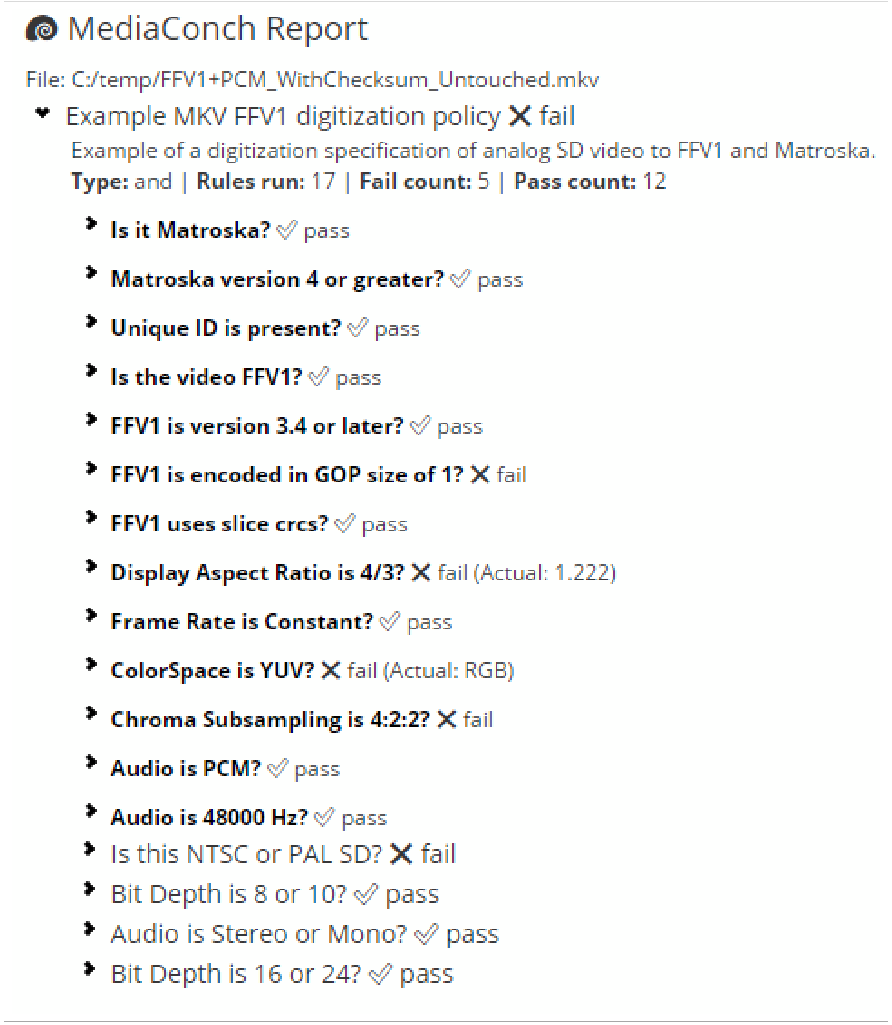
- **EBML-ELEM-START** Tests run: 1 | Results: ✓
- **EBML-VER-COH** Tests run: 1 | Results: ✓
- **EBML-DOCVER-COH** Tests run: 1 | Results: ✓
- **EBML-ELEMENT-VALID-PARENT** Tests run: 87 | Results: ✓
- **EBML-ELEMENT-NONMULTIPLES** Tests run: 70 | Results: ✓
- **EBML-ELEMENT-CONTAINS-MANDATES** Tests run: 43 | Results: ✓
- **EBML-ELEMENT-IN-SIZE-RANGE** Tests run: 43 | Results: ✓
- **EBML-VALID-MAXID** Tests run: 1 | Results: ✓
- **EBML-VALID-MAXSIZE** Tests run: 1 | Results: ✓
- **HEADER-ELEMENTS-WITHIN-IDLENGTH-LIMIT** Tests run: 1 | Results: ✓
- **ELEMENTS-WITHIN-MAXIDLENGTH** Tests run: 1 | Results: ✓
- **HEADER-ELEMENTS-WITHIN-MAXSIZELENGTH** Tests run: 1 | Results: ✓
- **ELEMENTS-WITHIN-MAXSIZELENGTH** Tests run: 1 | Results: ✓
- **MKV-SEEK-RESOLVE** Tests run: 4 | Results: ✓
- **EBML-CRC-FIRST** Tests run: 6 | Results: ✓
- **EBML-CRC-VALID** Tests run: 6 | Results: ✓
- **MKV-VALID-TRACKTYPE-VALUE** Tests run: 2 | Results: ✓
- **MKV-VALID-BOOLEANS** Tests run: 3 | Results: ✓

MediaConch FFV1 Implementation Checker

- **FFV1-SLICE-CRC-VALID** Tests run: 4 | Results: ✓

MediaConch PCM Implementation Checker

Policy report:



MediaConch Report

File: C:/temp/FFV1+PCM_WithChecksum_Untouched.mkv

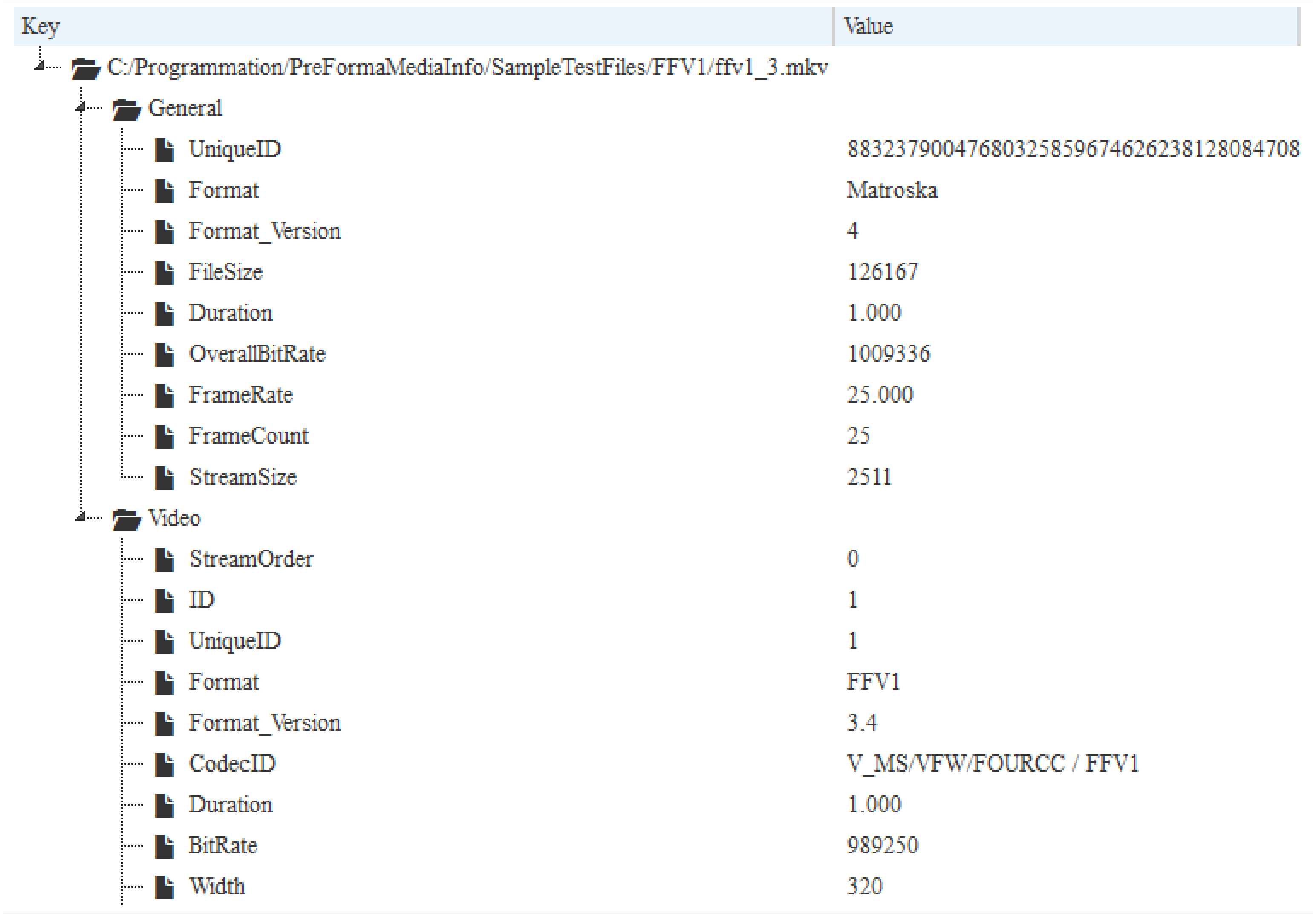
Example MKV FFV1 digitization policy ✗ fail

Example of a digitization specification of analog SD video to FFV1 and Matroska.
Type: and | **Rules run:** 17 | **Fail count:** 5 | **Pass count:** 12

- **Is it Matroska?** ✓ pass
- **Matroska version 4 or greater?** ✓ pass
- **Unique ID is present?** ✓ pass
- **Is the video FFV1?** ✓ pass
- **FFV1 is version 3.4 or later?** ✓ pass
- **FFV1 is encoded in GOP size of 1?** ✗ fail
- **FFV1 uses slice crcs?** ✓ pass
- **Display Aspect Ratio is 4/3?** ✗ fail (Actual: 1.222)
- **Frame Rate is Constant?** ✓ pass
- **ColorSpace is YUV?** ✗ fail (Actual: RGB)
- **Chroma Subsampling is 4:2:2?** ✗ fail
- **Audio is PCM?** ✓ pass
- **Audio is 48000 Hz?** ✓ pass
- Is this NTSC or PAL SD? ✗ fail
- Bit Depth is 8 or 10? ✓ pass
- Audio is Stereo or Mono? ✓ pass
- Bit Depth is 16 or 24? ✓ pass

What is MediaConch?

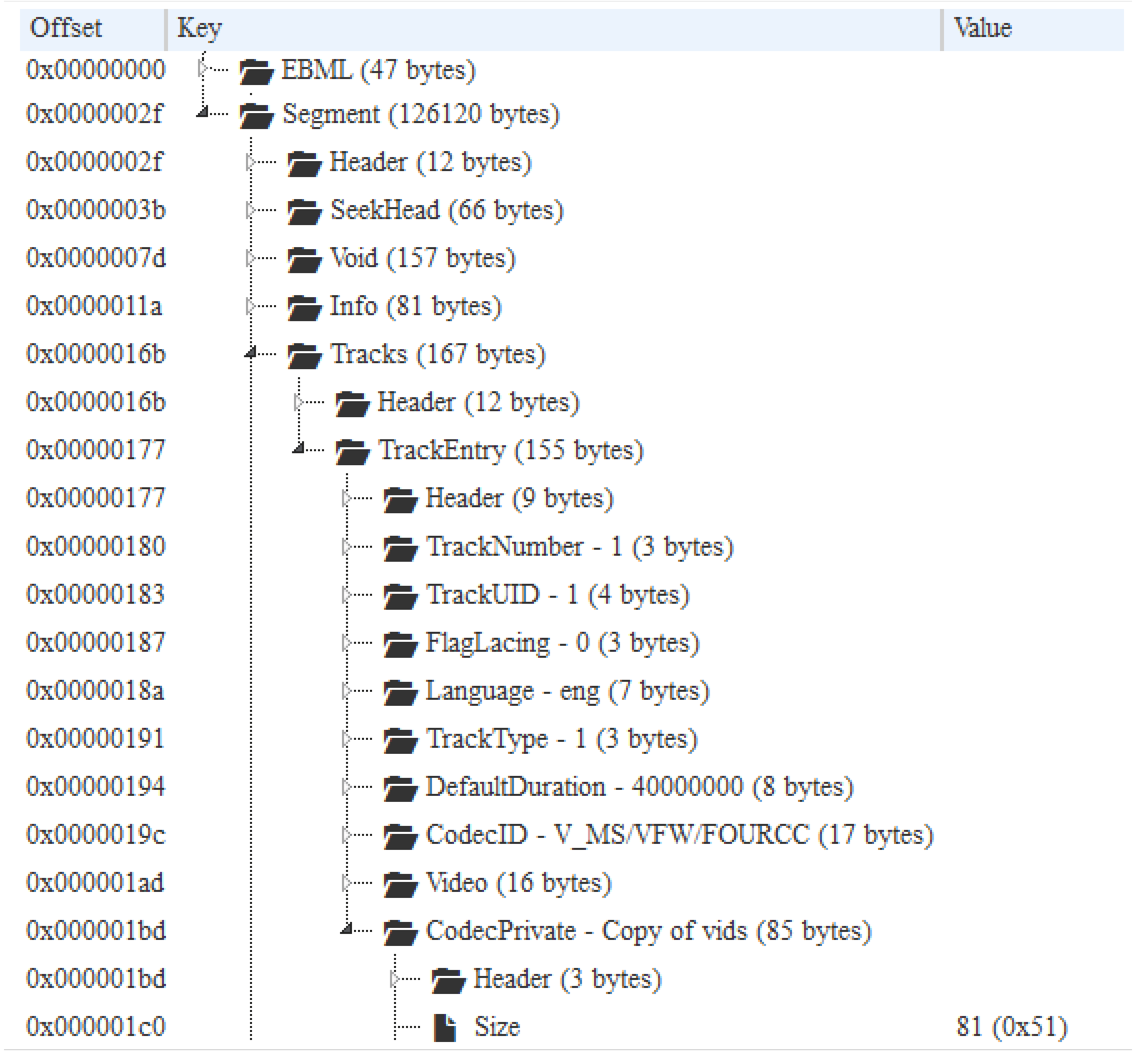
General information about your files



Key	Value
C:/Programmation/PreFormaMediaInfo/SampleTestFiles/FFV1/ffv1_3.mkv	
General	
UniqueID	88323790047680325859674626238128084708
Format	Matroska
Format_Version	4
FileSize	126167
Duration	1.000
OverallBitRate	1009336
FrameRate	25.000
FrameCount	25
StreamSize	2511
Video	
StreamOrder	0
ID	1
UniqueID	1
Format	FFV1
Format_Version	3.4
CodecID	V_MS/VFW/FOURCC / FFV1
Duration	1.000
BitRate	989250
Width	320

What is MediaConch?

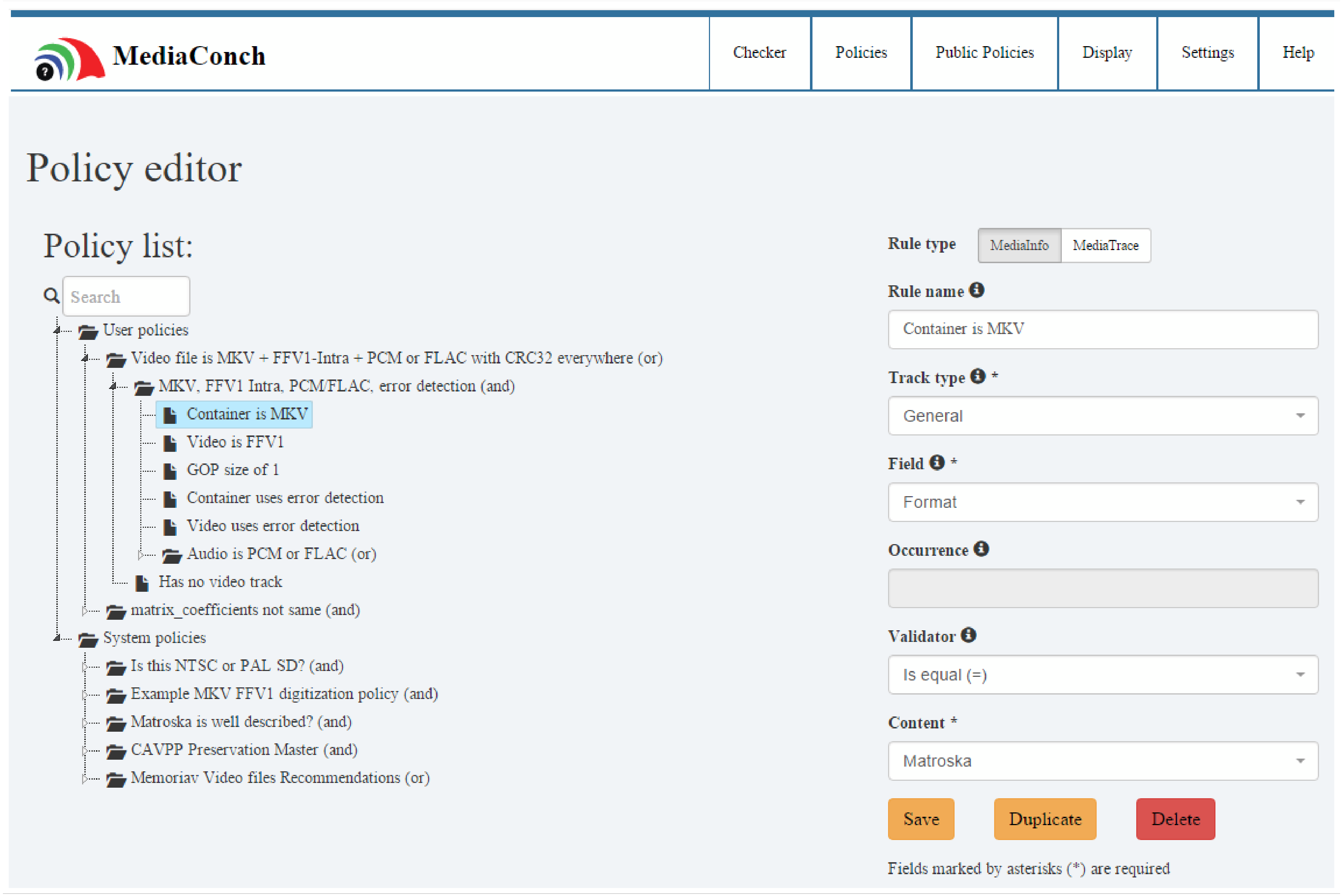
Inspect your files



```
Offset      Key                                                 Value
0x00000000  EBML (47 bytes)
0x0000002f  Segment (126120 bytes)
0x0000002f      Header (12 bytes)
0x0000003b      SeekHead (66 bytes)
0x0000007d      Void (157 bytes)
0x0000011a      Info (81 bytes)
0x0000016b      Tracks (167 bytes)
0x0000016b          Header (12 bytes)
0x00000177          TrackEntry (155 bytes)
0x00000177              Header (9 bytes)
0x00000180              TrackNumber - 1 (3 bytes)
0x00000183              TrackUID - 1 (4 bytes)
0x00000187              FlagLacing - 0 (3 bytes)
0x0000018a              Language - eng (7 bytes)
0x00000191              TrackType - 1 (3 bytes)
0x00000194              DefaultDuration - 40000000 (8 bytes)
0x0000019c              CodecID - V_MS/VFW/FOURCC (17 bytes)
0x000001ad              Video (16 bytes)
0x000001bd              CodecPrivate - Copy of vids (85 bytes)
0x000001bd                  Header (3 bytes)
0x000001c0                  Size                            81 (0x51)
```

What is MediaConch?

Policy editor



Policy editor

Policy list:

Search

- User policies
  - Video file is MKV + FFV1-Intra + PCM or FLAC with CRC32 everywhere (or)
    - MKV, FFV1 Intra, PCM/FLAC, error detection (and)
      - Container is MKV
      - Video is FFV1
      - GOP size of 1
      - Container uses error detection
      - Video uses error detection
      - Audio is PCM or FLAC (or)
    - Has no video track
  - matrix_coefficients not same (and)
- System policies
  - Is this NTSC or PAL SD? (and)
  - Example MKV FFV1 digitization policy (and)
  - Matroska is well described? (and)
  - CAVPP Preservation Master (and)
  - Memoriav Video files Recommendations (or)

Rule type: [✓ MediaInfo] MediaTrace

Rule name: Container is MKV

Track type *: General

Field *: Format

Occurrence:

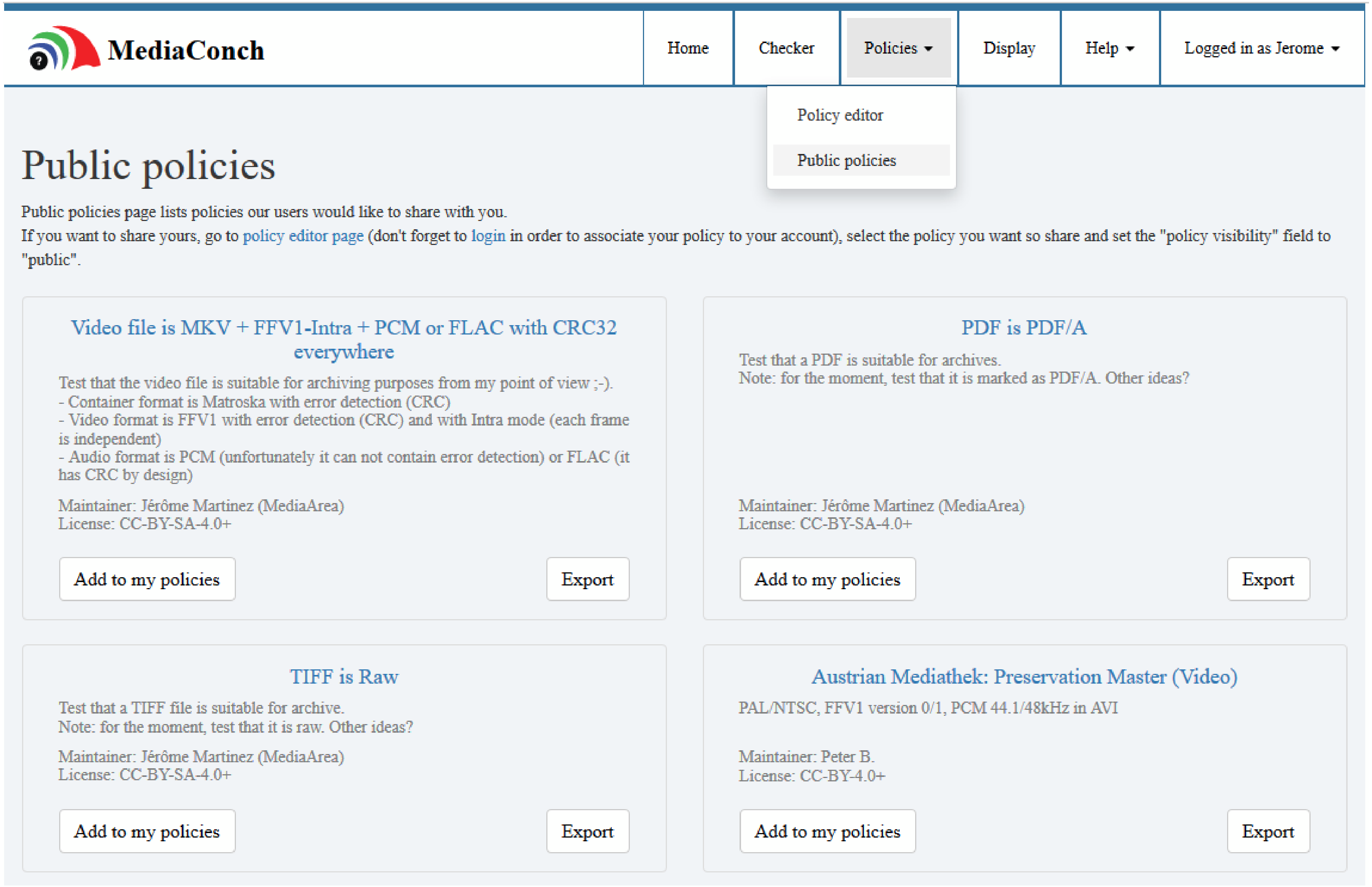
Validator: Is equal (=)

Content *: Matroska

Fields marked by asterisks (*) are required

What is MediaConch?

Public policies



Public policies

Public policies page lists policies our users would like to share with you.
If you want to share yours, go to policy editor page (don't forget to login in order to associate your policy to your account), select the policy you want so share and set the "policy visibility" field to "public".

Video file is MKV + FFV1-Intra + PCM or FLAC with CRC32 everywhere

Test that the video file is suitable for archiving purposes from my point of view ;-).
- Container format is Matroska with error detection (CRC)
- Video format is FFV1 with error detection (CRC) and with Intra mode (each frame is independent)
- Audio format is PCM (unfortunately it can not contain error detection) or FLAC (it has CRC by design)

Maintainer: Jérôme Martinez (MediaArea)
License: CC-BY-SA-4.0+

Add to my policies | Export

PDF is PDF/A

Test that a PDF is suitable for archives.
Note: for the moment, test that it is marked as PDF/A. Other ideas?

Maintainer: Jérôme Martinez (MediaArea)
License: CC-BY-SA-4.0+

Add to my policies | Export

TIFF is Raw

Test that a TIFF file is suitable for archive.
Note: for the moment, test that it is raw. Other ideas?

Maintainer: Jérôme Martinez (MediaArea)
License: CC-BY-SA-4.0+

Add to my policies | Export

Austrian Mediathek: Preservation Master (Video)

PAL/NTSC, FFV1 version 0/1, PCM 44.1/48kHz in AVI

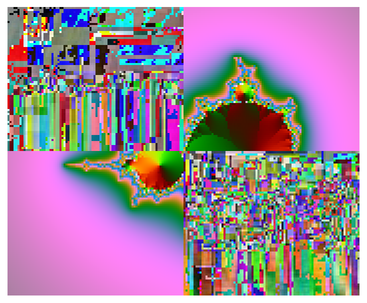
Maintainer: Peter B.
License: CC-BY-4.0+

Add to my policies | Export

What is MediaConch?

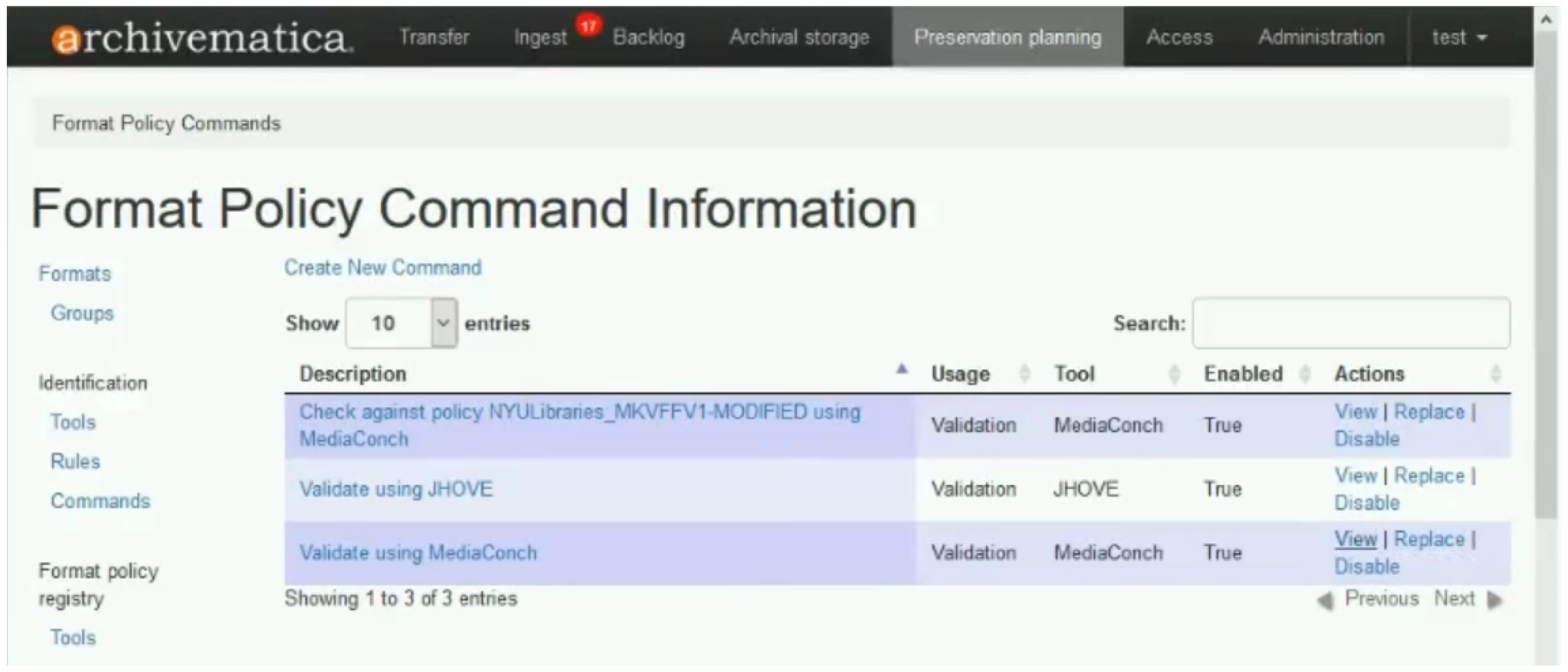
Fixer

- Segment sizes in Matroska
- Matroska “bit flip” correction
- FFV1 “bit flip” correction



Integration

Archivematica is an integrated suite of open-source software tools that allows users to process digital objects from ingest to access in compliance with the ISO-OAIS functional model



Format Policy Command Information

Create New Command

Show 10 entries Search:

Description	Usage	Tool	Enabled	Actions
Check against policy NYULibraries_MKVFFV1-MODIFIED using MediaConch	Validation	MediaConch	True	View | Replace | Disable
Validate using JHOVE	Validation	JHOVE	True	View | Replace | Disable
Validate using MediaConch	Validation	MediaConch	True	View | Replace | Disable

Showing 1 to 3 of 3 entries

Previous Next

MediaConch interfaces

- Graphical interface
- Web interface
- Command line
- Server (REST API)
- (Work in progress) a library (.dll/.so/.dylib)

MediaConch output formats

- XML (native format)
- Text
- HTML
- (Work in progress) PDF
- Tweakable! (with XSL)

Open source

- GPLv3+ and MPLv2+
- Relies on MediaInfo (metadata extraction tool)
- Use well-known open source libraries: Qt, sqlite, libevent, libxml2, libxslt, libexslt...

Supported formats

- Priorities for the implementation checker
  - Matroska
  - FFV1
  - PCM
- Can accept any format supported by MediaInfo for the policy checker
  - MXF + JP2k
  - QuickTime/MOV
  - Audio files (WAV, BWF, AIFF...)
  - ...

Supported formats

Can be expanded

- By plugins
  - Support of PDF checker: VeraPDF plugin
  - Support of TIFF checker: DPF Manager plugin
  - You use another checker? Let us know
- By internal development
  - More tests on your preferred format is possible
  - It depends on you!

Versatile

Several input formats are accepted

- FFV1 from MOV or AVI
- Matroska with other video formats
- (Work in progress) Extraction of a PDF or TIFF attachement from a Matroska container and analyze with a plugin (e.g. VeraPDF and DPF Manager)
- ...

Versatile

Input can be from:

- Files (local/network)
- FTP/FTPS/SFTP
- HTTP/HTTPS
- Amazon S3

Versatile

Binaries are provided for:

- Windows
- Mac

  Homebrew users:"brew install mediaconch", that's all!
- Linux (Ubuntu, Debian, Fedora, OpenSUSE...)

  Since Ubuntu 16.04 and Debian Testing/9 users:

  "apt-get install mediaconch" or in Ubuntu Store, that's all!

  (it is in the official distros repository)
- Embedded devices? Doable

  (we tested it on a Raspberry Pi)
- Can be ported on other distros (BSD...)

Standardization

- Matroska is widely used but not (yet) standardized
- FFV1 is gaining increasing usage in preservation contexts but is not (yet) standardized

CELLAR: IETF workgroup

- Open standards group
- Goal to IETF-standardize Matroska/FFV1/FLAC
- A lot of progress, especially with Matroska/EBML specs
- https://datatracker.ietf.org/wg/cellar/charter/

FFV1 performance

- NOA tested on SD 8-bit content:
  - i7-2600 (4 cores+HT, 3.4-3.8 GHz)
  - 3-4x real time
  - 4-5x decoding speed increase compared to JP2k
- VIAA is testing on SD 10-bit content (FFmpeg 3.2):
  - E5-2698V3 (16 cores+HT, 2.3-3.6 GHz)
  - 0.7x real time/thread, 11-12x real time/all cores+HT
  - 3-4x decoding speed increase compared to JP2k
  - Better compression ratio by 8-10% compared to JP2k

FFV1 performance

- This is an average, results varies depending on the content of files
  - From 0.4x to 2.4x (average 0.7x) real time/thread (encoding/decoding)
  - From 0.7x to 16x (average 3.5x) the speed of JP2k (FFmpeg)
- Not convinced?
  - Test on your own files
  - MediaArea will provide test scripts
  - We can perform tests for you

Worldwide

- 2 project leaders
  - Jerôme Martinez (Digital Media Analysis Specialist, France)
  - Dave Rice (Archivist, USA)
- Presentations worldwide
  - IASA, France
  - FIAT/IFTA, Austria
  - FOSDEM, Belgium
  - AMIA, USA
  - Code4Lib, USA
  - JTS, Singapore
  - (3-6 October 2016) IPRES, Switzerland
  - (25-29 September 2016) IASA, USA

Matroska research corpus

- We analyze all Matroska files from archive.org
- Interface with some statistics of Matroska elements usage (e.g. files with CRC-32 elements…)
  https://mediaarea.net/MediaConchCorpus/

What's next?

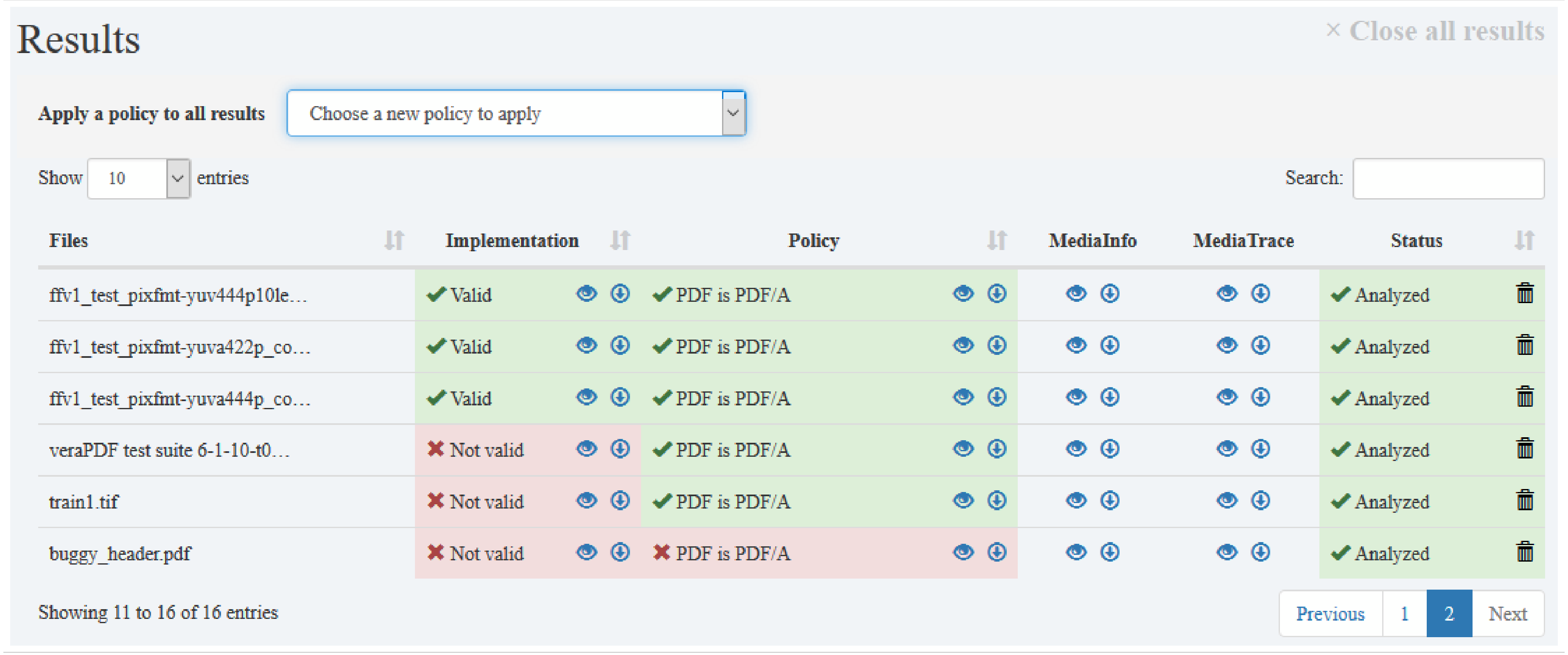
- Continue to improve handling of huge collections
- Continue to improve user interface
- Support of embedded attachments
- Statistics
- Finish standardization of Matroska and FFV1
- More conformance tests
- More fixing cases

And after PREFORMA sponsorship?

It depends on you!

- This is open source
- Driven by user requests
- Everyone can develop or sponsor a development
- Potential features:
  - Support of tests for your prefered format (MOV? MXF? JP2k? WAV?)
  - Support of other checkers (BWF MetaEdit? QCTools?)
  - Integration in your workflow
  - ...

Example (Plugins)



Results

× Close all results

Apply a policy to all results: Choose a new policy to apply

Show 10 entries

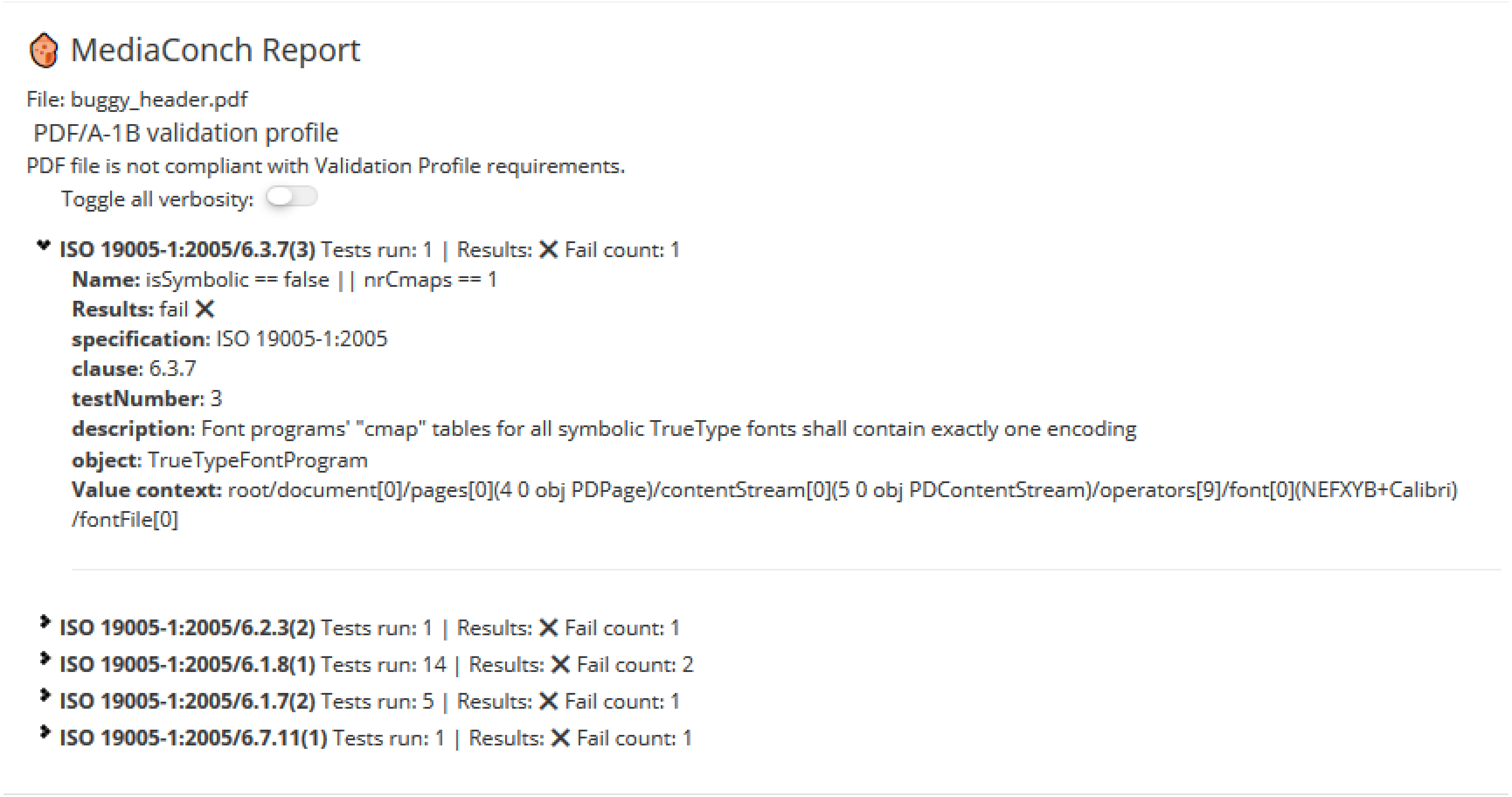
Search:

Files	Implementation	Policy	MediaInfo	MediaTrace	Status
ffv1_test_pixfmt-yuv444p10le…	✓ Valid 👁 ⬇	✓ PDF is PDF/A 👁 ⬇	👁 ⬇	👁 ⬇	✓ Analyzed 🗑
ffv1_test_pixfmt-yuva422p_co…	✓ Valid 👁 ⬇	✓ PDF is PDF/A 👁 ⬇	👁 ⬇	👁 ⬇	✓ Analyzed 🗑
ffv1_test_pixfmt-yuva444p_co…	✓ Valid 👁 ⬇	✓ PDF is PDF/A 👁 ⬇	👁 ⬇	👁 ⬇	✓ Analyzed 🗑
veraPDF test suite 6-1-10-t0…	✗ Not valid 👁 ⬇	✓ PDF is PDF/A 👁 ⬇	👁 ⬇	👁 ⬇	✓ Analyzed 🗑
train1.tif	✗ Not valid 👁 ⬇	✓ PDF is PDF/A 👁 ⬇	👁 ⬇	👁 ⬇	✓ Analyzed 🗑
buggy_header.pdf	✗ Not valid 👁 ⬇	✗ PDF is PDF/A 👁 ⬇	👁 ⬇	👁 ⬇	✓ Analyzed 🗑

Showing 11 to 16 of 16 entries

Previous 1 2 Next

Example (Plugins)



MediaConch Report

File: buggy_header.pdf

PDF/A-1B validation profile

PDF file is not compliant with Validation Profile requirements.

Toggle all verbosity: ○

▾ **ISO 19005-1:2005/6.3.7(3)** Tests run: 1 | Results: ✕ Fail count: 1

Name: isSymbolic == false || nrCmaps == 1
Results: fail ✕
specification: ISO 19005-1:2005
clause: 6.3.7
testNumber: 3
description: Font programs' "cmap" tables for all symbolic TrueType fonts shall contain exactly one encoding
object: TrueTypeFontProgram
Value context: root/document[0]/pages[0](4 0 obj PDPage)/contentStream[0](5 0 obj PDContentStream)/operators[9]/font[0](NEFXYB+Calibri)/fontFile[0]

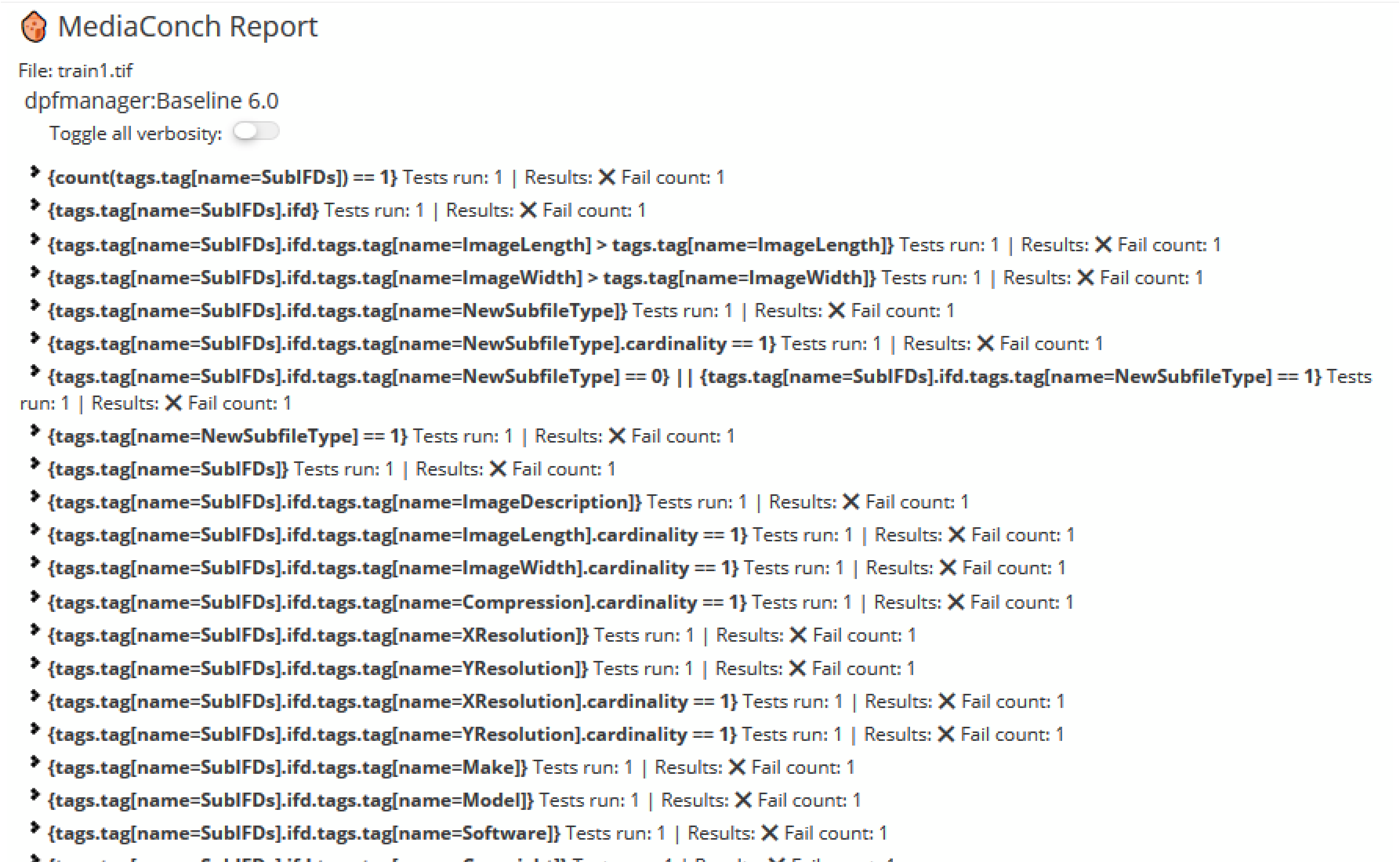
▸ **ISO 19005-1:2005/6.2.3(2)** Tests run: 1 | Results: ✕ Fail count: 1

▸ **ISO 19005-1:2005/6.1.8(1)** Tests run: 14 | Results: ✕ Fail count: 2

▸ **ISO 19005-1:2005/6.1.7(2)** Tests run: 5 | Results: ✕ Fail count: 1

▸ **ISO 19005-1:2005/6.7.11(1)** Tests run: 1 | Results: ✕ Fail count: 1

Example (Plugins)



MediaConch Report

File: train1.tif

dpfmanager:Baseline 6.0

Toggle all verbosity:

- **{count(tags.tag[name=SubIFDs]) == 1}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=ImageLength] > tags.tag[name=ImageLength]}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=ImageWidth] > tags.tag[name=ImageWidth]}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=NewSubfileType]}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=NewSubfileType].cardinality == 1}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=NewSubfileType] == 0} || {tags.tag[name=SubIFDs].ifd.tags.tag[name=NewSubfileType] == 1}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=NewSubfileType] == 1}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs]}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=ImageDescription]}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=ImageLength].cardinality == 1}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=ImageWidth].cardinality == 1}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=Compression].cardinality == 1}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=XResolution]}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=YResolution]}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=XResolution].cardinality == 1}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=YResolution].cardinality == 1}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=Make]}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=Model]}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=Software]}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=Copyright]}** Tests run: 1 | Results: ✕ Fail count: 1

- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=Copyright]}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=DateTimeOriginal]}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=DateTimeOriginal].cardinality == 20}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=DateTime].cardinality == 20}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=TIFFEPStandardID]}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=TIFFEPStandardID].cardinality == 4}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=PhotometricInterpretation] == 1} || {tags.tag[name=SubIFDs].ifd.tags.tag[name=PhotometricInterpretation] == 2} || {tags.tag[name=SubIFDs].ifd.tags.tag[name=PhotometricInterpretation] == 6} || {tags.tag[name=SubIFDs].ifd.tags.tag[name=PhotometricInterpretation] == 32803} || {tags.tag[name=SubIFDs].ifd.tags.tag[name=PhotometricInterpretation] == 32767}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=PlanarConfiguration]}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=PlanarConfiguration].cardinality == 1}** Tests run: 1 | Results: ✕ Fail count: 1
- **{tags.tag[name=SubIFDs].ifd.tags.tag[name=PlanarConfiguration] == 1} || {tags.tag[name=SubIFDs].ifd.tags.tag[name=PlanarConfiguration] ==**

Stay in touch

MediaArea: https://mediaarea.net, @MediaArea_net

MediaConch: https://mediaarea.net/MediaConch, @MediaConch

Jérôme Martinez: jerome@mediaarea.net

Slides: https://mediaarea.net/Events