

## PRÉAMBULE

Cette documentation a pour but de décrire de façon la plus large possible - et dans la limite du raisonnable - le **Digital Cinema Package (DCP)**.

**Débuté en 2020** de façon candide, elle a dérivé sur une agrégation des différentes techniques et informations que j'utilisais depuis des années (décennies même, à l'origine du DCI). Cette documentation se veut **enfin** une compilation de toutes mes recherches à travers le temps - à propos du Cinéma numérique, du DCI et surtout du DCP - en un seul point unique.

Depuis les premières spécifications DCI et la publication de différents outils depuis 2004-2005, il a toujours été difficile d'obtenir des informations sur les entrailles d'un DCP (du fait surtout de normes non publiques). Chacun y allant de ses implémentations sans expliquer, à moindre expert, le fonctionnement. Les différentes informations techniques sont la compilation de mes études, codes et notes techniques à travers le temps.

Cette documentation est à destination autant de l'expert que du néophyte :

- Un **néophyte**, comprendre rapidement et facilement ce qu'est un DCP.
- Un **expert** pourra s'y reporter rapidement sans avoir à relire les milliers de pages de normes SMPTE sur le sujet (ou même y redécouvrir des choses ?)

Cette documentation ne prétend pas être parfaite, elle peut comporter des erreurs, des oublis, des manquements ou des incompréhensions :

- Si vous avez un certain niveau technique et historique et que vous identifiez une erreur ou un oubli, **ne pas hésiter à me prévenir**.
- Si vous êtes néophyte et que certains passages vous semblent incompréhensibles ou manquant de clarté, **ne pas hésiter à me prévenir également**.
- Si vous pensez qu'il manque un sujet dans la documentation, ne pas hésiter à m'en faire part, elle peut faire l'objet d'une extension.

**Cette documentation ne se veut pas une traduction** des différentes normes SMPTE et spécifications DCI. Ces dernières sont tellement vastes qu'il serait illusoire et contreproductif de le faire, en sachant qu'elles évoluent.

**Cette documentation a pour but d'être évolutive** malgré tout, elle n'est pas figée dans le temps et sera donc mise à jour régulièrement, par exemple pour rajouter des évolutions aux normes ou des nouveaux codes sources. Ne pas hésiter à y revenir ou consulter le [changelog](#).

**Les codes sources et techniques données ne se veulent pas être des « best practices of coding »**, les codes sources seront les plus simplifiés - parfois jusqu'au ridicule - afin de permettre de bien comprendre chaque étape. Je ne supporte pas les documentations donnant des codes sources que seules les experts peuvent relire. C'est pour moi une façon détournée malvenue qu'a l'auteur de justifier son "expertise face aux autres". Si vous trouvez un code source ou une technique encore trop complexe, ne pas hésiter à m'en faire part, je simplifierai davantage encore.

Ces codes sources ont une vocation secondaire : A être intégré à d'autres codes ou projets : c'est comme **une bibliothèque de codes prêt-à-emploi**, piochez et intégrez dans vos projets pour aller plus vite.

## LES OUTILS ET PROTOTYPES

Durant l'écriture de cette documentation, j'ai eu besoin d'outils, donc je les ai développés. Les outils & prototypes utilisés dans la documentation ont été développés en parallèle de celle-ci. Vous croiserez ces outils à travers les

différents chapitres. En supplément, vous découvrirez des versions allégées et plus simples de ces outils pour mieux comprendre les mécanismes sous-jacents.

En complément des données d'exemples à travers le [répertoire des assets](#) (que vous pouvez fouiller à travers les chapitres), vous aurez des petits programmes et morceaux de codes sources (parties pouvant être intégrées à la documentation pour études).

Ci-dessous, une brève présentation sur les principaux outils.

Les différents programmes ci-dessous sont des POC (proof-of-concept), leurs codes ne sont pas optimisés et méritent une réécriture totale (c'est prévu...) cependant, ils sont fonctionnels pour les besoins de cette documentation.

## MXF ANALYZER

Un analyseur et extracteur des métadonnées et données MXF.

Le projet est maintenant disponible sur github : <https://github.com/sherpadown/mxf-reader/>

```
$ mxf_analyzer.py --help
-h, --help          show this help message and exit
-f FILENAME, --filename FILENAME = mxf filename
-x EXTRACT, --extract EXTRACT = extract each KLV into files
-k KEY, --key KEY   AES Key, ex. --key 00000000000000000000000000000000
-v, --verbose       increase output verbosity
-n, --no-resolv     Do not resolv UL (speed)
--filter FILTER     filter by name
--fuzzy            Fuzzy mode only (very slow)
--limit LIMIT       stop after x klv parsed
--slow             Slowdown parse to avoid flood loadavg
```

```
# mxf_analyzer.py -f 2D.mxf
offset | uuid | ber | data-size | data | name
0 | 060e2b34.02050101.0d010201.01020400 | 83.000078 | 120 | 00010002000000010000000000000000 | Partition
140 | 060e2b34.02050101.0d010201.01050100 | 83.000596 | 1430 | 0000004f000000120201060e2b340101 | Primer Pad
1590 | 060e2b34.02530101.0d010101.01012f00 | 83.0000be | 190 | 3c0a0010aaffbd07420e44a2b3f87d3c | Preface
1800 | 060e2b34.02530101.0d010101.01013000 | 83.0000d8 | 216 | 3c0a0010c00c5d91bc1b4e58aefabf33 | Identifica
2036 | 060e2b34.02530101.0d010101.01011800 | 83.00005c | 92 | 3c0a0010acc6d12cda0c465b861a1c33 | Content St
2148 | 060e2b34.02530101.0d010101.01012300 | 83.000048 | 72 | 3c0a0010bf8a6ebc3aca4164a47e45e0 | Essence Co
2240 | 060e2b34.02530101.0d010101.01013600 | 83.0000a0 | 160 | 3c0a001096b13153bd5d4cd29abcd608 | Material P
2420 | 060e2b34.02530101.0d010101.01013b00 | 83.000070 | 112 | 3c0a0010005ea82d6afb4a37aae4d74b | Timeline T
2552 | 060e2b34.02530101.0d010101.01010f00 | 83.000050 | 80 | 3c0a00108afb9187a608464f85713462 | Sequence
2652 | 060e2b34.02530101.0d010101.01011400 | 83.00004b | 75 | 3c0a00108fe911af95d4464e9e794651 | Timecode C
2747 | 060e2b34.02530101.0d010101.01013b00 | 83.00006e | 110 | 3c0a0010f8427198af884d4f8c9855f3 | Timeline T
2877 | 060e2b34.02530101.0d010101.01010f00 | 83.000050 | 80 | 3c0a0010e2bfd055e5d249a0bfc4da8 | Sequence
2977 | 060e2b34.02530101.0d010101.01011100 | 83.00006c | 108 | 3c0a00106daf21bab69a41c99ec54f4f | Source Cli
3105 | 060e2b34.02530101.0d010101.01013700 | 83.000116 | 278 | 3c0a0010b87f0871f98446d39a842893 | Source Pad
3403 | 060e2b34.02530101.0d010101.01013b00 | 83.000070 | 112 | 3c0a00106378d2218be746d28508fedf | Timeline T
3535 | 060e2b34.02530101.0d010101.01010f00 | 83.000050 | 80 | 3c0a001075ef53606937484a9ee14472 | Sequence
3635 | 060e2b34.02530101.0d010101.01011400 | 83.00004b | 75 | 3c0a001059cf1ad86f314b5db0164871 | Timecode C
3730 | 060e2b34.02530101.0d010101.01013b00 | 83.00006e | 110 | 3c0a0010d4e733700afb45329cc796c6 | Timeline T
3860 | 060e2b34.02530101.0d010101.01010f00 | 83.000050 | 80 | 3c0a0010e2f035aa64684a7f98eddfa7 | Sequence
3960 | 060e2b34.02530101.0d010101.01011100 | 83.00006c | 108 | 3c0a0010c93d2257b09a4d4bea24c782e | Source Cli
4088 | 060e2b34.02530101.0d010101.01012900 | 83.0000bd | 189 | 3c0a00102cb46505be6b41c3abbe8848 | RGBA Essen
4297 | 060e2b34.02530101.0d010101.01015a00 | 83.0000b5 | 181 | 3c0a00106054268f8fdf47ba98db7167 | JPEG2000 P
4498 | 060e2b34.01010102.03010210.01000000 | 83.002e5a | 11866 | 00000000000000000000000000000000 | KLV Fill i
16384 | 060e2b34.02050101.0d010201.01030400 | 83.000078 | 120 | 00010002000000010000000000004000 | Partition
16524 | 060e2b34.01020101.0d010301.15010801 | 83.009cc8 | 40136 | ff4fff51002f000400001000000000870 | Picture Es
56680 | 060e2b34.02050101.0d010201.01040400 | 83.000078 | 120 | 0001000200000001000000000000dd68 | Partition
56820 | 060e2b34.02530101.0d010201.01100100 | 83.000083 | 131 | 3c0a00101c86aa0136fb414ebcd47120 | Index Tabl
```

Et avec du verbose :

```
# mxf_analyzer.py -f 2D.mxf
offset | uuid | ber | data-size | data | name
0 | 060e2b34.02050101.0d010201.01020400 | 83.000078 | 120 | 00010002000000010000000000000000 | Par
```

```

Major Version           || 1
Minor Version           || 2
KAGSize                 || 1
ThisPartition           || 0
PreviousPartition       || 0
FooterPartition         || 56680
HeaderByteCount         || 16244
IndexByteCount          || 0
IndexSID                || 0
BodyOffset              || 0
BodySID                 || 0
Operational Pattern     || 060e2b34.04010102.0d010201.10000000 (Operational Patter
EssenceContainers       || 2 item(s): 060e2b34.04010103.0d010301.027f0100, 060e2b3
Resource [1]            || SMPTE.ST.0377-1-2011 - MXF - File Format Specification

140 | 060e2b34.02050101.0d010201.01050100 | 83.000596 :                1430 | 0000004f000000120201060e2b340101 | Prj
|
| 0201 | 060e2b34.01010102.04070100.00000000 - Statically Local
| 0202 | 060e2b34.01010102.07020201.01030000 - Statically Local
(...)

```

Une version (ultra) simplifiée d'analyseur MXF en ligne est disponible ici : <https://github.com/sherpadown/js-mxf/> (demo)

## UL ANALYZER

Un analyseur *Quick & Dirty* d'Universal Label

```

$ ul "060e2b34020401010d010301027e0100"
SMPTe UL Format          060e2b34.02040101.0d010301.027e0100
Hex Format (lower)       06.0e.2b.34.02.04.01.01.0d.01.03.01.02.7e.01.00
Hex Format (upper)       06.0E.2B.34.02.04.01.01.0D.01.03.01.02.7E.01.00
Hex Format (small)       060e2b34020401010d010301027e0100

name                    Encrypted Essence Container (SMPTe)
parser                  essence_encrypted
resource                SMPTe.ST.0429-6-2006 - DCP - MXF Track File Essence Encryption, Page 10

UL Header (2 bytes)    060e
UL Designator (6 bytes) 2b3402040101
Item Designator (8 bytes) 0d010301027e0100

1 - Object ID          06
2 - UL Size             0e (14 bytes)
3 - ISO/ORG Identifier / UL Code 2b
4 - SMPTe Designator   34
5 - Category Designator 02 (Groups - Sets & Packs)
6 - Registry Designator 04 (Variable-Length Pack) - [Length=BER short/long] (All Length)
7 - Structure Designator 01 (Set/Pack Registry)
8 - Version             01

9 - Item Designator     0d
10 - Item Designator    01
11 - Item Designator    03
12 - Item Designator    01
13 - Item Designator    02
14 - Item Designator    7e
15 - Item Designator    01
16 - Item Designator    00

```

## IAB ANALYZER

Un analyseur des données Immersive Audio Bitstream, comme les Dolby Atmos, DTS-X, Barco Auro, ...

Le projet est maintenant disponible sur github : <https://github.com/sherpadown/iab-reader/>

```
Usage: iab-analyzer.py <IAB data> or <IAB directory>
```



```
CME - Registration values 1: Text ISO 8859-1
CME - Text: Created with Doremi Labs DMS2000 SN70062 server v1.8.0. Src0.
( . . . )
[POC] Progression Order Change (FF5F)
POC - [2K] RSpoc (Resolution Start) : 0 (0x00)
POC - [2K] CSpoc (Component Start) : 0 (0x00)
POC - [2K] LYEpoc (Layer) : 1 (0x0001)
POC - [2K] REpoc (Resolution End) : 6 (0x06)
POC - [2K] CEpoc (Component End) : 3 (0x03)
POC - [2K] Ppoc (Progression Order) : 4 (0x04)
POC - [4K] RSpoc (Resolution Start) : 6 (0x06)
( . . . )
[TLM] Tile-part lengths, main header (FF55)
TLM - Ztlm: Index of this marker segment: 0
TLM - Stlm: Size of the Ttlm and Ptlm parameters: 01010000
TLM - Stlm Bit-Parameters: SP = 1; Ptlm parameter 32 bits
TLM - Stlm Bit-Parameters: ST = 1; Ttlm parameter 8 bits
TLM - Ttlm - Tile number of tile-part 1: 0
TLM - Ptlm - Length SOT+SOD of tile-part 1: 190 bytes
( . . . )
[SOT] Start of tile-part (FF90)
SOT - Isot, Tile number : 0
SOT - Psot, Length of SOT+SOD : 190
SOT - TPst, Tile-part number : 0
SOT - TNst, Number of tile-parts : 6
( . . . )
[SOD] Start of data (FF93)
data: 176 bytes : eff07ffe0fc0115054afff5574bcab4000000000000c2425ff69dc40000000000309097fe4a5400000000061212ffc94a
( . . . )
```

## CONCLUSION

Retrouvons le [sommaire](#) de cette documentation.

=====