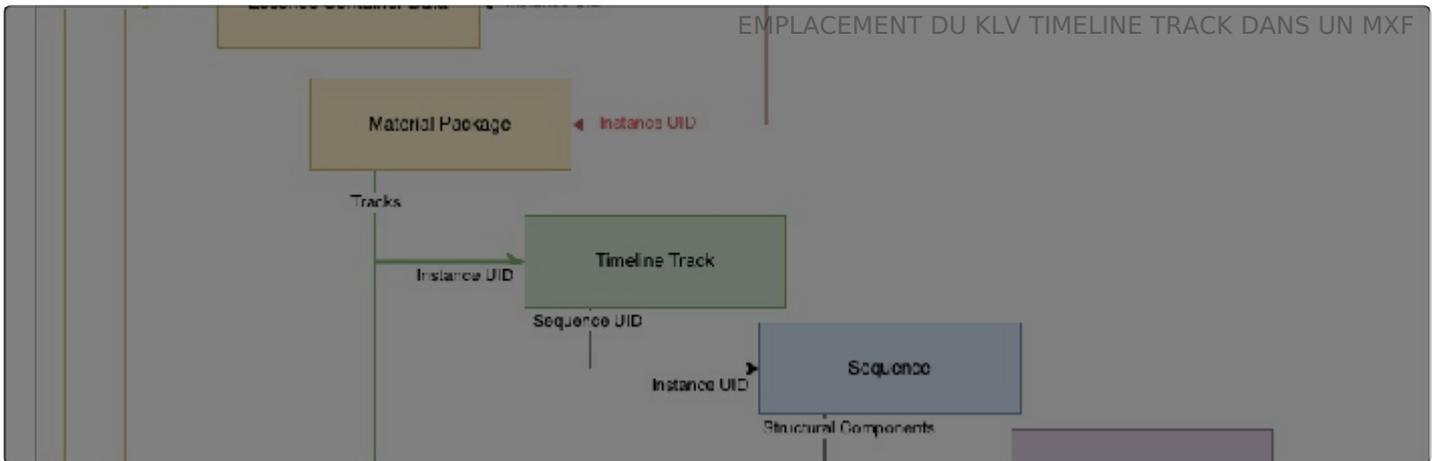


<b>Références</b>	<a href="#">SMPTE 377-1-2011 - MXF - File Format Specification</a> Chapitre B.12 - Timeline Track <sup>P123</sup> Chapitre B.15 - Timeline Track - Timecode <sup>P124</sup> Chapitre B.18 - Timeline Track - Picture <sup>P126</sup> Chapitre B.21 - Timeline Track - Sound <sup>P127</sup> Chapitre B.24 - Timeline Track - Data <sup>P127</sup> Chapitre B.27.1 - Timeline Track - Descriptive Metadata (DM) <sup>P127</sup>
<b>Modèle KLV</b>	<a href="#">Local Sets</a>
<b>Universal Label</b>	06.0e.2b.34.02.53.01.01.0d.01.01.01.01.01.3b.00 (SMPTE)

## PRÉFACE



**Timeline Track** fera une simple relation vers le KLV **Sequence**

**Timeline Track**, comme **Static Track**, font partie du type **Track**, ils partageront donc des points communs.

**Timeline Track** peut travailler sur une partie de la piste (contrairement à **Static Track** qui ne pourra travailler que sur l'entièreté de la piste).

Dans un MXF DCP, nous aurons des **Timeline Track**, dans les blocs (**Material Package** et **Source Package**). Chacun feront références à des KLV **Sequence** qui lui-même fera référence à d'autres KLV comme **Source Clip** ou **Timecode Component**.

Voici un exemple de suite logique de **Timeline Track** dans chaque **Package** et ses enfants (que nous verrons par la suite) :

**Timeline Track** → **Sequence** → **Timecode Component** **Timeline Track** → **Sequence** → **Source Clip**

N'oubliez pas que **Timeline Track** = **Static Track** avec limitations temporelles. Normalement, nous pourrions remplacer **Timeline Track** par une **Static Track**. Normalement... parce qu'il existe quand même des restrictions dans les normes sur quand utiliser une **Timeline** ou quand utiliser une **Static**. Par exemple, si vous voulez utiliser un **Descriptive Metadata Segment**, vous devrez utiliser un **Static Track** comme parent :)

A la différence de **Static Track**, vous aurez des items **Origin** et **Edit Rate**.

## LES MÉTADONNÉES

Voici quatres exemples de **Timeline Track**.

L'une venant du bloc **Material Package** (lié au MXF et à la *timeline*) :

```
Timeline Track #1
3C0A - Instance ID      || 3cac453b.a5aa4f38.a2c2ac3d.26641cef
4801 - Track ID        || 1
4804 - Track Number    || 0
4802 - Track Name      || Timecode Track
4803 - Sequence        || d329b223.05f444f3.b53d99a4.3736bc4f
4B01 - Edit Rate       || 24/1
4B02 - Origin          || 0

Timeline Track #2
3C0A - Instance ID      || cddf8d1.99104345.8552a513.56e8025f
4801 - Track ID        || 2
4804 - Track Number    || 0
4802 - Track Name      || Picture Track
4803 - Sequence        || 26be6965.2e49401a.9403760e.698fbb80
4B01 - Edit Rate       || 24/1
4B02 - Origin          || 0
```

Et l'autre venant du bloc **Source Package** (lié aux essences) :

```
Timeline Track #1
3C0A - Instance UID    || 46a6c435.4e214d7f.8974cd5a.5c77fce1
4801 - Track ID        || 1
4804 - Track Number    || 0
4802 - Track Name      || Timecode Track
4803 - Sequence UID    || d5e6616d.fbcc42eb.bbbda96f.9d356068
4B01 - Edit Rate       || 24/1
4B02 - Origin          || 0

Timeline Track #2
3C0A - Instance UID    || fa1243a1.21874f1f.b34014bf.3353ee05
4801 - Track ID        || 2
4804 - Track Number    || 352389121
4802 - Track Name      || Picture Track
4803 - Sequence UID    || d424bd75.6f84499e.8728eee5.daa65edf
4B01 - Edit Rate       || 24/1
4B02 - Origin          || 0
```

**Sequence** fait référence au KLV **Sequence** via son **Instance UID**.

**Track ID** va définir un simple numéro (1, 2, 3, etc...) de piste pour ce package et qui sera en lien avec le **SourceTrackID** dans le KLV **Source Clip**. Le **Track ID** commence en général à 1. S'il commence par 0, c'est que vous avez un vieux MXF car cette valeur est considérée comme obsolète.

**Track Number** spécifie une référence au type de container des essences (oui, le choix du nom est hasardeux). Vous n'aurez que des valeurs nulles sauf pour le dernier **Timeline Track** qui est en lien avec l'essence. Et selon son essence, vous aurez comme code :

Code Hexadécimal	Type d'essence
0x15010801	Picture
0x16010101	Sound
0x17010b01	Subtitle

Ces valeurs hexadécimales représentent des identifiants définis dans la norme du MXF Generic Container. <sup>1</sup>

Par exemple :

- Le premier élément représente "Item Type Identifier", par exemple :
  - 0x15 représente un Generic Container Picture
  - 0x16 représente un Generic Container Sound
- Le second élément est le "Essence Element Count", le nombre d'élément, donc toujours à 01 puisqu'il n'y a qu'un élément par container d'essence.

- Le troisième élément est le "Essence Element Type", le type d'essence défini dans le SMPTE 331M.
- Le dernier représente un simple numéro d'essence (en général 1)

Mais pour faire plus simple, si vous regardez plus en détail, vous verrez que cette suite de chiffre ne vous est pas inconnue. Et oui, c'est la fin d'un **Universal Label** et pas n'importe lesquels, ceux des KLV Essences :

```
060e2b34.01020101.0d010301.15010801 : Picture Essence - 1 frame
060e2b34.01020101.0d010301.16010100 : Sound Essence - Wave Frame-Wrapped Element
060e2b34.01020109.0d010301.17010b01 : TimedText Essence
```

**EditRate** est la vitesse de la timeline en Hertz. Par exemple, 24/1 ou 48/1 ou pire... 24000/1001, qui est le ... 23.97 (qui devrait brûler dans les tréfonds des anciennes technos)

**Origin** indique s'il y a des éléments avant ou après le début de la **Timeline Track**, le nombre de frame avant ou après le point d'origine. Cet item est utilisé s'il existe un décalage dans la timeline. Vous aurez toujours la valeur 0 car on n'effectue pas de décalage dans un MXF de DCP.

## LES DONNÉES BRUTES DU KLV

Voir l'étude du parsing d'un KLV [Local Set](#) dans [Preface](#), avec en supplément :

- Le parsing des Strings UTF-16 a été vu dans [Identification](#)

Et le format d'**EditRate** que nous allons étudier.

Voici un exemple de données brutes au format hexadécimal avec un code couleur pour les différentes parties :

```
3c0a0010fa1243a121874f1fb34014bf3353ee05480100040000000248040004150108014802
001a005000690063007400750072006500200054007200610063006b48030010d424bd756f84
499e8728eee5daa65edf4b0100080000001800000014b02000800000000000000
```

## STRUCTURES DES DONNÉES

Local Tag	Nom de l'attribut	Type	Taille (*)	Fixe/Variable SMPTE	Obligatoire
<b>3C0A</b>	Instance ID	UUID	16 octets	Fixe	Oui
<b>4801</b>	Track ID	uint32	4 octets	Fixe	Oui
<b>4804</b>	Track Number	uint32	4 octets	Fixe	Oui
<b>4802</b>	Track Name	String UTF-16	26 octets	<b>Variable</b>	<b>Non</b>
<b>4803</b>	Sequence UID	UUID	16 octets	Fixe	Oui
<b>4B01</b>	Edit Rate	Rationnel (Int32/Int32)	2 x 4 octets	Fixe	Oui
<b>4B02</b>	Origin	int64 (1)	8 octets	Fixe	Oui

(\*) N'oubliez pas que les tailles sont purement indicatives car elles sont dynamiques et non fixes et dépendent des **Length** de chaque item. Notez cependant que la documentation SMPTE peut définir malgré tout certains items avec une valeur fixe, regardez dans la colonne **Fixe/Variable SMPTE** qui indique si la taille a été normée.

(1) Ce n'est pas uint64 car le nombre peut-être négatif dans certaines occasions.

L'**EditRate** est relativement simple, c'est un nombre rationnel donc sous forme XX/YY. Pour son encodage, il a été décidé d'encoder le premier nombre dans les premiers octets, et le second nombre dans les octets suivants. Nous avons une taille de 8 octets, les 4 premiers octets seront pour le numérateur, les 4 autres seront pour le dénominateur.

Ainsi, notre valeur hexadécimale `0000001800000001` doit s'interpréter de la sorte :

- `0x00000018` est le numérateur
- `0x00000001` est le dénominateur

Si nous convertissons ces valeurs hexadécimales, nous aurons `24` et `1`, notre EditRate sera donc `24/1`

## HIÉRARCHIE DU FORMAT

---

Interchange Object → Generic Track → Timeline Track

## ETUDE RAPIDE DE L'UNIVERSAL LABEL DE TIMELINE TRACK

---

```
UL = 06.0E.2B.34.02.53.01.vv.0D.01.01.01.01.3B.00
      ^^-----
      ^^----- Item Designator : Organizationally registered
      ^^----- Organization : AAF
      ^^----- Application : MXF / AAF Association Structural M
      ^^----- Structure Version : Structure Version 1
      ^^----- Structure Kind : MXF / AAF Association compatible s
      ^^----- MXF Set Definition : Timeline Track
      ^^----- MXF Set Definition : Timeline Track
```

## NOTES

---

1. « *The Track Number in the File Package Picture Track Is a UInt32 with the same value as bytes 13-16 of the MXF Generic Container Element Key used to wrap the essence data. This value is unique in any partition. The correct partition is found by using the BodySID mechanism as detailed in the MXF File Format Specification and as outlined here* » -- SMPTE 379 - MXF Generic Container ↩