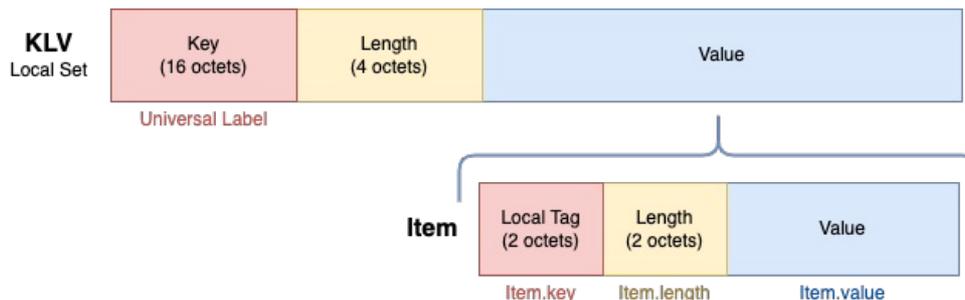


KLV : LOCAL TAG : LES MINI-IDENTIFIANTS

Si on peut résumer les Local Tags en quelques lignes :

- Un **Local Tag** est l'équivalent d'un **Universal Label** mais en 2 octets seulement :
Il est utilisé comme clef pour les KLV **Local Sets** (nos fameux bébés-KLV).
- Un **Local Tag** est une sorte de petit nom pour un **Universal Label**

En une image, voici où se situe un **Local Tag** à l'intérieur d'un KLV **Local Set**:



La valeur d'un **Local Tag** est compris entre `0x0001` à `0xFFFF` et est séparée en deux grandes catégories :

| | |
|-----------------|-------------------------------|
| 0x0000 | Ne doit pas être utilisé |
| 0x0001 - 0x00FF | Reservé compatibilité AAF |
| 0x0100 - 0x7FFF | Local Tags - Statique |
| 0x8000 - 0xFFFF | Local Tags - Dynamique |

Par exemple, voici quelques **Local Tag Statiques** représentant un type de données :

| Local Tag | Description |
|-----------|----------------------|
| 3C0A | Instance UID |
| 3B02 | Last Modified Date |
| 3B05 | Version |
| 3B07 | Object Model Version |
| 3B08 | Primary Package |
| 3B06 | Identifications |
| 3B03 | Content Storage |
| 3B09 | Operational Pattern |
| 3B0A | EssenceContainers |
| 3B0B | DM Schemes |
| 3C09 | This Generation UID |
| 3C01 | Company Name |
| 3C02 | Product Name |
| 3C03 | Product Version |
| 3C04 | Version String |

| | |
|-------------------------------|------------------------|
| 3C05 | Product UID |
| 3C06 | Modification Date |
| 3C07 | ToolkitVersion |
| 3C08 | Platform |
| Content Storage | |
| 1901 | Packages |
| 1902 | Essence Container Data |
| Content Storage Data | |
| 2701 | Linked Package UID |
| 3F06 | IndexSID |
| 3F07 | BodySID |
| Generic Package | |
| (ex. Material Package) | |
| 4401 | Package UID |
| 4402 | Name |
| 4405 | Package Creation Date |
| 4404 | Package Modified Date |
| 4403 | Tracks |
| Timeline Track | |
| 4801 | Track ID |
| 4804 | Track Number |
| 4B01 | Edit Rate |
| 4B02 | Origin |
| 4802 | Track Name |
| 4803 | Sequence |
| Sequence | |
| 1001 | Structural Components |
| Structural Component | |
| 0201 | Data Definition |
| 0202 | Duration |
| Source Clip | |
| 1201 | Start Position |
| 1101 | SourcePackageID |
| 1102 | SourceTrackID |
| DM Segment | |
| 0601 | Event Start Position |
| 0602 | Event Comment |
| | |

| | |
|--------------------------------|------------------------|
| 6102 | Track IDs |
| 6101 | DM Framework |
| Timecode Component | |
| 1501 | Start Timecode |
| 1502 | Rounded Timecode Base |
| 1503 | Drop Frame |
| Source Package | |
| 4701 | Descriptor |
| RGBA Essence Descriptor | |
| 3006 | Linked Track ID |
| 3001 | Sample Rate |
| 3002 | Container Duration |
| 3004 | Essence Container |
| 320C | Frame Layout |
| 3203 | Stored Width |
| 3202 | Stored Height |
| 320E | Aspect Ratio |
| 3201 | Picture Essence Coding |
| 3406 | Component Max Ref |
| 3407 | Component Min Ref |
| 3408 | Alpha Max Ref |
| 3409 | Alpha Min Ref |
| 3405 | ScanningDirection |
| 3401 | PixelLayout |
| 3403 | Palette |
| 3404 | PaletteLayout |
| 320D | Video Line Map |
| Index Table Segment | |
| 3F0B | Index Edit Rate |
| 3F0C | Index Start Position |
| 3F08 | Slice Count |
| 3F0D | Index Duration |
| 3F0E | PosTableCount |
| 3F09 | Delta Entry Array |
| 3F0A | Index Entry Array |
| 3F05 | Edit Unit Byte Count |

Et là, vous me dites, mais c'est quoi cette différence entre un **Local Tag Statique** et un **Local Tag Dynamique**

STATIQUE VS DYNAMIQUE

Cette partie est importante à connaître sinon vous ne comprendrez pas pourquoi les valeurs dans vos items dans vos KLV **Local Sets** seront "différentes" d'un MXF à l'autre. Vous allez comprendre pourquoi.

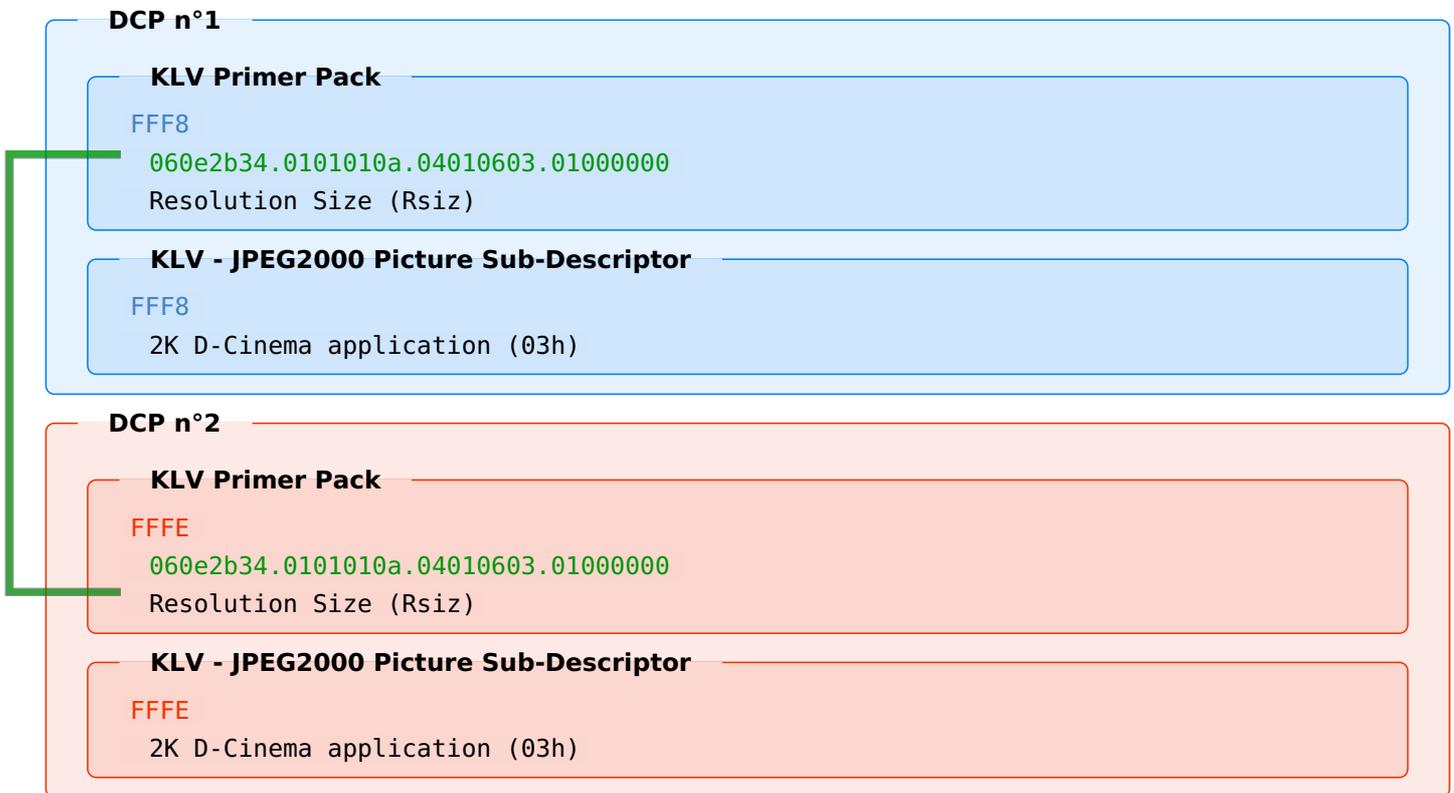
Le principe du **Local Tag Dynamique** est relativement simple mais complètement chiant :

- Un **Local Tag Statique** aura un identifiant unique, commun à l'ensemble des types de KLV et des MXF, il ne changera pas. Par exemple, `3C0A` sera un toujours un **Instance UID** au format UUID. Vous pouvez voir un listing de quelques Local Tag Statique juste au dessus.
- Un **Local Tag Dynamique** est exactement l'inverse : c'est un identifiant qui sera différent d'un MXF à l'autre et d'un KLV à l'autre. Par exemple, l'item **Resolution Size (Rsiz)** dans le KLV **JPEG2000 Picture Sub-Descriptor** aura un **Local Tag** différents à chaque fois.

Comment identifier un Local Tag dynamique ?

Un identifiant **Local Tag** égal ou supérieur à `0x8000` est **dynamique**.

Un exemple avec un **Resolution Size (Rsiz)** sur deux DCP différents :



Entre ces deux DCP, les **Local Tag** (FFF8 & FFFE) sont différents, cependant son **Universal Label** est toujours le même.

Pour être certain que le **Local Tag** manipulé soit le bon, il faut aller identifier son **Universal Label** qui lui est rattaché dans le KLV **Primer Pack**.

Ainsi, par exemple, un **item.key** avec un **Local Tag** statique à `3C0A`, nous savons que nous aurons affaire à un **Instance UID** donc avec une donnée d'une taille de 16 octets au format UUID. Et cela dans tous les MXF respectant la norme SMPTE.

Quand ils sont statiques, pas besoin de connaître l'**Universal Label**, il suffit de lire les normes SMPTE car ils sont entièrement décrits : on connaît leur code, leur taille et le type de valeur.

Cependant, dans les KLV utilisant les **Local Tags** dynamiques, ces derniers sont totalement "aléatoires" : Par

exemple, il est impossible d'être sûr qu'un **Local Tag** `FFFE`, utilisé dans un KLV, soit finalement un item **Cryptographic Key ID** ou un item **SubDescriptor** ou un item **Resolution Size** ou encore tout autre chose.

Ne considérez jamais un **Local Tag** supérieur à `0x8000` comme un acquis. Si dans un de vos MXF, le `FFFE` peut être un **Cryptographic Key ID**, il se peut que dans un autre il sera tout autre chose (en dehors d'un **Local Tag** statique déjà attribué).

Dès qu'on rencontre un **Local Tag** supérieur à `0x8000`, pour connaître sa véritable identité, il faut se reporter absolument au **Primer Pack** : C'est lui qui détient un registre entre les identifiants **Local Tag** utilisés dans le MXF et leurs **Universal Label** équivalents.

Autant un **Local Tag** supérieur à `0x8000` peut-être attribué à n'importe quoi et qui, autant un **Universal Label** est unique à un type particulier. Dès qu'on détecte un **Local Tag** dynamique dans un KLV, il faut aller lire son **Universal Label** dans **Primer Pack** et ainsi en déduire son type, son format et sa taille.

Statique est obsolète

Les **Local Tag statiques** sont considérés comme "deprecated", ils ne doivent donc ne plus être utilisés.

Malgré tout, ils restent encore utilisés dans les MXF, mais les nouveaux **Local Tag** doivent être maintenant dynamiques.

Vous retrouverez tous les **Local Tag statiques** et leurs équivalences **Universal Label** dans [ce listing](#) et tous les **Universal Label** spécifiques aux **Local Tag** et leurs descriptions dans [ce listing](#).