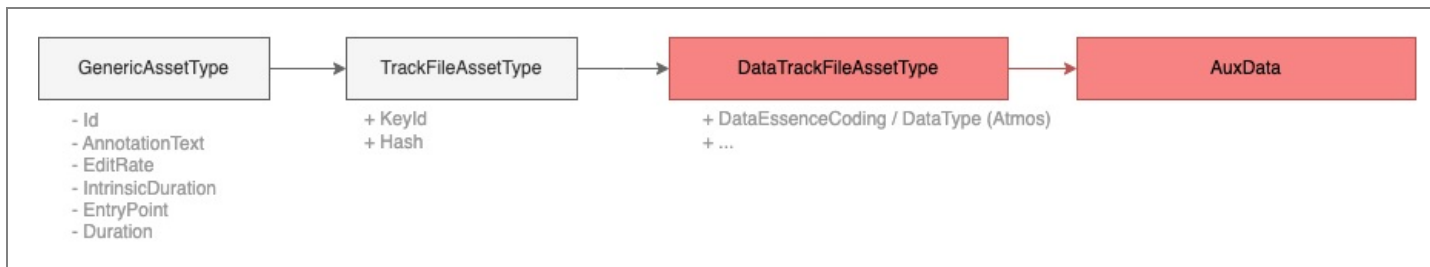




PRÉFACE

Le type d'asset **DataTrackFile** est utilisé par **AuxData**.



RÉFÉRENCES SMPTE / DCI

- **SMPTE 429-14** - Aux Data Track File
- **SMPTE 429-18** - Immersive Audio Track File
- **SMPTE 429-19** - DCP Operational Constraints for Immersive Audio

DESCRIPTION

Un exemple d'**AuxData** ¹ orienté Dolby (le plus souvent) :

```
<AuxData xmlns="http://www.dolby.com/schemas/2012/AD">
  <Id>urn:uuid:25b1eaeb-f034-480c-8edc-248d1d7ec2f2</Id>
  <EditRate>24 1</EditRate>
  <IntrinsicDuration>720</IntrinsicDuration>
  <EntryPoint>0</EntryPoint>
  <Duration>720</Duration>
  <KeyId>urn:uuid:ec7377b6-e69c-48ef-9396-d4e9349d57dd</KeyId>
  <Hash>RMiUP0Er7koH3eUWuva4/uLDVl8=</Hash>
  <DataType>urn:smpte:ul:060e2b34.04010105.0e090604.00000000</axd:DataType>
</AuxData>
```

----- Generic

|

----- TrackFile

----- DataTrackFile

Le tag AuxData est utilisé pour deux assets différents :

- Les assets **AuxData**
- les assets **Immersive Audio Bitstream (IAB)**

A la base, l'asset AuxData permettait de stocker des éléments autres, mais depuis récemment, avec l'arrivée de la norme IAB, cette dernière réutilise la syntaxe AuxData pour se l'approprier.

ASSET AUXDATA

DataTrackFile possède un tag **DataEssenceCoding** optionnel et est un UUID qui va déterminer le type de l'essence. Cet UUID commence par `urn:smpte:url:`

Voici un [exemple](#) de **DTS-X** ² :

```
<DataEssenceCoding>urn:smpte:ul:060e2b34.0401010e.0e1d0101.01010100</DataEssenceCoding>
```

La valeur dans **DataEssenceCoding** est la valeur se trouvant dans le KLV **Aux Data Essence Descriptor** à l'item **Data Essence Coding**.

ASSET IMMERSIVE AUDIO BITSTREAM (IAB)

Pour l'[Immersive Audio \(IAB\)](#), la norme IAB indique l'utilisation du tag **DataType** à la place de DataEssenceCoding. C'est donc le tag [DataType](#) que vous verrez pour un MXF IAB :

```
<DataType>urn:smpte:ul:060e2b34.04010105.0e090604.00000000</DataType>
```

Ici, nous avons l'identifiant **Immersive Audio Data Essence Coding**, donc l'utilisation d'un asset compatible [Immersive Audio Bitstream \(IAB\)](#) comme [Atmos](#), [DTS-X](#), [Auro](#).

La valeur dans **DataType** est la valeur se trouvant dans le KLV **Immersive Audio Data Essence Descriptor** à l'item **Data Essence Coding**. Cependant la norme indique qu'il faut toujours utiliser la valeur `060E2B34.04010105.0E090604.00000000` que ce soit dans la CPL et aussi dans le MXF.

CONCLUSION

Si vous trouvez un **DataType** dans <AuxData>, vous aurez un **MXF Immersive Audio Bitstream (IAB)**, soit un Dolby Atmos, soit un DTS-X, soit un Barco-Auro, tous sont maintenant compatible IAB. Retrouvez le chapitre [Immersive Audio Bitstream](#) avec ses sous-chapitres.

Si vous trouvez un **DataEssenceCoding** dans <AuxData>, vous aurez un **MXF AuxData pur**. Il faudra donc déterminer le type d'essence stocké dans le MXF à l'aide de l'identifiant inscrit. Retrouvez le chapitre [MXF AuxData](#)

CHAPITRE ANNEXES

- [AssetType Generic](#) : Le parent de tous
 - [AssetType TrackFile](#) : Le parent de Picture, Sound, Subtitle, Caption et Data.
 - [AssetType PictureTrackFile](#) : MainPicture et MainStereoscopicPicture
 - [AssetType SoundTrackFile](#) : MainSound
 - [AssetType SubtitleTrackFile](#) : MainSubtitle et ClosedSubtitle
 - [AssetType CaptionTrackFile](#) : MainCaption et ClosedCaption
 - [AssetType DataTrackFile](#) : AuxData ➡ [vous êtes ici](#)
 - [AssetType Marker](#) : MainMarkers
 - [AssetType CompositionMetadata](#) : CompositionMetadataAsset

NOTES

1. Pour **AuxData**, le namespace qualifier sera `axd` et son URI sera `http://www.smp-te-ra.org/schemas/429-14/2014/Aux-Data` (pour un AuxData classique) ou `http://www.dolby.com/schemas/2012/AD` pour du Dolby. Le namespace qualifier n'est pas obligatoire, l'URI, elle, si. ➡
2. N'ayant pas eu accès au MXF de ce DCP, je ne peux vous dire exactement si ce MXF "AuxData" est un AuxData-DTS-X d'avant Immersive Audio ou si c'est un AuxData-DTS-X intégrant la norme Immersive Audio. Cela serait étonnant, car il devrait utiliser le DataType en lieu et place de DataEssenceCoding. ➡