aaf2.mobid module

Excerpt from SMPTE ST 330 (Focus on Basic UMID):

5 General Specification

A unique material identifier (UMID) provides for the globally unique identification of any audiovisual material.
This standard defines a dual approach through the specification of a basic UMID and an extended UMID. The basic UMID provides a globally unique identification for audiovisual material that comprises an integer
number of one or more contiguous material units. The basic UMID has no embedded mechanism to distinguish between individual material units within a single instance of audiovisual material. The data in the
basic UMID can be created through automatic generation.

The extended UMID comprises the basic UMID followed immediately by a source pack that provides a signature for material units. The source pack comprises a fixed length metadata pack of 32 bytes that provides sufficient metadata by which source ?when, where and who (or what)? information can be identified
regardless of current ownership or status. The extended UMID also provides a mechanism to distinguish between individual material units within a single instance of audiovisual material.

The basic UMID is 32 bytes long and the extended UMID is 64 bytes long.
Both UMID types use the key-length-value construct defined by SMPTE ST 336. The key is a 16-byte universal label truncated to 12 bytes.
In the case of the basic UMID, the length field has a value of 13h and the value is formed by the combination
of a material number and an instance number.
In the case of the extended UMID, the length field has a value of 33h and the value is formed by the combination of the material and the instance numbers followed by the source pack.
All components of the UMID have a defined byte order for consistent application in storage and streaming
environments.

The components of the basic UMID are:
1. A 12-byte universal label,
2. A 1-byte length value,
3. A 3-byte instance number, and
4. A 16-byte material number.

The combination of the instance and material numbers can be treated as a dumb number.
Note: The material number does not indicate the status of the material (such as copy number) or its representation
(such as the compression kind). The material number can be identical in copies and in different representations of
the material. The purpose of the instance number is to separately identify different representations or instances of
audiovisual material. Thus, for example, a high-resolution picture and a thumbnail can both have the same
material number because they both represent the same picture but, because they are different instances, they will
have different instance numbers for the different representations. Guidance for the consistent application of new
material numbers and instance numbers is given in SMPTE RP 205.

UMID universal label (SMPTELabel)

Byte No.	Description	Value (hex)	Meaning
1	Object identifier	06h	Universal label start
2	Label size	0Ah	12-byte Universal label
3	Designation: ISO	2Bh	ISO registered
4	Designation: SMPTE	34h	SMPTE registered
5	Registry category	01h	Dictionaries
6	Specific category	01h	Metadata dictionaries
7	Structure	01h	Dictionary standard (SMPTE ST 335)
8	Version number	05h	Version of the metadata dictionary (defined in SMPTE RP 210)

9	Class	01h	Identifiers and locators
10	Subclass	01h	Globally unique identifiers
11	Material type	XXh	See Section 6.1.2.1
12	Number creation method	YYh	See Section 6.1.2.2

6.1.2.1 - Material type identification

Byte 11 of the UL shall define the material type being identified using one of the values defined in Table 2.
The use of material types '01h', '02h', '03h' and '04h' shall be deprecated for use in implementations using
this revised standard. These values are preserved only for compatibility with systems implemented using
SMPTE ST 330:2000#

Table 2

Byte value	Meaning	Examples and notes
01h	picture material	Deprecated
02h	audio material	Deprecated
03h	data material	Deprecated
04h	other material	Deprecated (originally not only picture, audio, or data material, but also a combination of material types)
05h	single picture component	e.g. Y component
06h	Two or more picture components in a single container	e.g. interleaved Y, Cb and Cr components
08h	single audio component	e.g mono audio
09h	two or more audio components in a single container	e.g. AES3 audio pair
0Bh	single auxiliary (or data) component	e.g. sub-titles only
0Ch	two or more auxiliary (or data) components in a single container	e.g. multiple sub-titles streams in different languages
0Dh	mixed group of components in a single container	e.g. video & stereo audio pair
0Fh	material type is not identified	

6.1.2.2 Number creation method identification

Byte 12 of the UL shall define the method by which the material and instance numbers are created. This byte
is divided into top and bottom nibbles for the purpose of this definition.
The top nibble shall occupy the 4 most significant bits (MSBs) of the byte and the value shall be used to
define the method of material number creation. The values used by this nibble shall be limited to the range 0
to 7h so that byte 12 conforms to the ASN.1 BER short form coding rules used by SMPTE ST 298.
The methods of material number generation shall be as defined in table 3 and the specification of the each
method shall be as defined in Annex A.
Note: New material number generation methods can be added by amendment or revision of this document. Each
addition will provide the proposed value (within the range of values currently identified as "Reserved but not
defined") for inclusion in Table 3 together with the supporting definition to be added to Annex A.

Table 3 - Identification of material number generation method::

Value (hex)	Method
0	No defined method
1	SMPTE method
2	UUID/UL method
3	Masked method

```
4             IEEE 1394 network method
5~7           Reserved but not defined
```

Notes from Pixar 10/30/17

Final note of discussion with Avid engineers how the top nibble in the 12th byte in the SMPTELabel should be set. (In the past we always had it set to 00, i.e. “no defined method”, we had some confusion about how to set it when using a uuid for the material)

Avid Engineer:

“The specification of number creation identification is very clear about using the 4 MSBs for the material number so I am pretty sure that the numbers in the sub-titles of Annex A (e.g. 02h) should not be interpreted literally as values of byte 12. 20h is the correct value of the byte 12 for UL/UUID method/No defined method.

By the way, I don’t see anything wrong with setting byte 12 to 00h (No defined method / No defined method)”

We at Pixar decided to set the byte to 20h, since it (even if already very minimal) completely eliminates the possibility to collide with any MOB ID created by our old MOB ID generation algorithm.

aaf2.mobid.UniqueMobID()

class **aaf2.mobid.MobID**(*mobid=None*, *bytes_le=None*, *int=None*)

Bases: `object`

bytes_le

static **new**()

Static method for generating unique MobIDs. Uses uuid.uuid4() for generation.

material

MobID material representation as a UUID

SMPTELabel

length

instanceHigh

instanceMid

instanceLow

Data1

Data2

Data3

Data4

from_dict(*d*)

Set MobID from a dict

to_dict()

MobID representation as dict

int

MobID representation as a int

urn

MobID Uniform Resource Name representation.
https://en.wikipedia.org/wiki/Uniform_Resource_Name