PROPOSED SMPTE STANDARD

SMPTE 330M
Revision of
SMPTE 330M-2000

for Television — Unique Material Identifier (UMID)

Table of contents

1 Scope

This standard specifies the format of the unique material identifier (UMID). The UMID is a unique identifier for audiovisual material which is locally created and globally unique. It differs from many unique identifiers in that the number does not depend wholly upon a pre-registration process, but can be generated automatically at the point of material origination without reference to a central resource.

The UMID provides a method of identification for instances of audiovisual material and thus enables the material to be linked with its associated metadata. The UMID itself is neither intended for the identification of copyright nor the ownership of rights. Nor, for example, does it identify program content or works.

The UMID consists of an ordered group of components each providing a key aspect to the identification of the audiovisual material, be it picture, sound or data. A key property of a UMID generated in accordance with this standard is that it is possible to use the resulting UMID simply as a globally unique dumb number.

The UMID may exist in one of two forms:

- A basic UMID which contains the minimum components necessary for unique identification; and
- An extended UMID which attaches a packed metadata set (source pack) to the basic UMID.

This standard specifies the formats of each component in both the basic UMID and the extended UMID.

This standard also identifies and specifies methods by which the components of the identifier can be created.

A glossary is provided to define the terms used in this standard and this scope.

2 Normative references

The following standards contain provisions which, through reference in this text, constitute provisions of this standard. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this standard are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below.

ANSI/SMPTE 298M-1997, Television — Universal Labels for Unique Identification of Digital Data

SMPTE 12M-1999, Television, Audio and Film — Time and Control Code

SMPTE 309M-1999, Television — Transmission of Date and Time Zone Information in Binary Groups of Time and Control Code

SMPTE 336M-2000, Television — Data Encoding Protocol Using Key-Length-Value

SMPTE 359M-2002, for Television and Motion Pictures — Dynamic Documents

SMPTE RP 210.4-2002, Metadata Dictionary Registry of Metadata Element Decriptions

AES3-1992 (R1997, Amendment 4-1999), AES Recommended Practice for Digital Audio Engineering (1999) — Serial Transmission Format for Two-Channel Linearly Represented Digital Audio Data

IEEE 1394-1995, High Performance Serial Bus

IETF-RFC1321, The MD5 Message-Digest Algorithm, Internet Engineering Task Force, 1992

ISO 3166-1:1997, Codes for the Representation of Names of Countries and Their Subdivisions — Part 1: Country Codes

ISO/IEC 8859-1:1998, Information Processing — 8-Bit Single-Byte Coded Graphics Character Sets — Part 1: Latin Alphabets No. 1

ISO/IEC 11578-1:1996, Information Technology — Open Systems Interconnection — Remote Procedure Call (RPC), Annex A, Universal Unique Identifier

Department of Defense, World Geodetic System 1984. U.S. Defense Mapping Agency DMA TR 8350.2, Second Edition, 1 September 1991

3 Glossary of terms

audio-visual material: Any one or any combination of picture (or video) essences, sound (or audio) essences and data (or auxiliary) essences. This term is also frequently referred to simply as “material”.

dumb number: A number, the value of which has no intrinsic meaning.

essence: An abstract term that describes any data or signal necessary to represent any single type of visual, aural or other sensory experience independent of the method of coding. Essence excludes any form of metadata.

material unit: The entity that represents the quantum of the audiovisual material defined by its cyclic sampling structure. The material unit duration depends on the type of material; examples being an AES3 block and a video frame. Note that the duration of a material unit is defined by the requirements of the audiovisual material type and its application. The value of the material unit duration is not specified in the basic UMID. It may, however, be specified in the source pack.

3.5 metadata: Data which conveys information about the audio-visual material. For example, information about identification, essence coding, timelines, intellectual property, business operations, etc.

4 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 basic UMID is a combination of four components requiring a data space of 32 bytes:

- The first component is a 12-byte universal label which identifies the UMID.
- The second component is a 1-byte number that defines the length of the remaining part of the UMID.
- The third component is a 3-byte instance number which is a locally unique number used to differentiate between different representations of material which share the same material number. This specification provides several different methods to generate the instance number and the selected method is identified in the universal label. The value of this component shall be capable of being treated as a dumb number.
- The fourth component is a 16-byte material number which is globally unique for every item of material. The UMID allows several different methods to generate the material number and the selected method is identified in the universal label. The value of this component shall be capable of being treated as a dumb number.

The combination of the instance and material numbers shall be capable of being treated as a dumb number.

Annex A defines methods by which the material number can be generated and which are known to provide a high likelihood of being unique.

Annex B defines methods by which the instance number can be generated.

The extended UMID requires a total data space of 64 bytes and comprises the basic UMID followed immediately by a source pack which provides signatures 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.

NOTE – The source pack metadata only represents that information available at the time when the extended UMID is first applied. The source pack values may therefore refer to the “when, where and who (or what)” at the point of application of the source pack and this may differ from the originating device.

All metadata fields in the source pack may be automatically generated for each material unit using a timer, a global position calculator and pre-registered identification data. However, where automatic generation is not possible, it can be manually entered. Once a source pack is written into an extended UMID, it shall not be changed and shall always relate to its associated material unit. Source pack data field values may be zero filled where either no meaningful source data exists or where operational practice dictates.

Both UMID types operate using the key-length-value construct of SMPTE 336M. The key is a 12-byte universal label which is defined in the metadata dictionary, SMPTE RP 210.4.

In the case of the basic UMID, the length field has a value of 13_h and the value shall be formed by the combination of the material number and the instance number.

In the case of the extended UMID, the length field has a value of 33_h and the value shall be formed by the combination of the material and the instance numbers followed by the source pack.

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 may 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 may both have the same material number because they both represent the same thing 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.

5 UMID format specification

The basic UMID requires 32 bytes and the extended UMID an additional 32 bytes to make a total of 64 bytes. All components of the UMID have a defined byte order for consistent application in storage and streaming environments. Figure 1 defines the layout of the both basic UMID and the extended UMID.

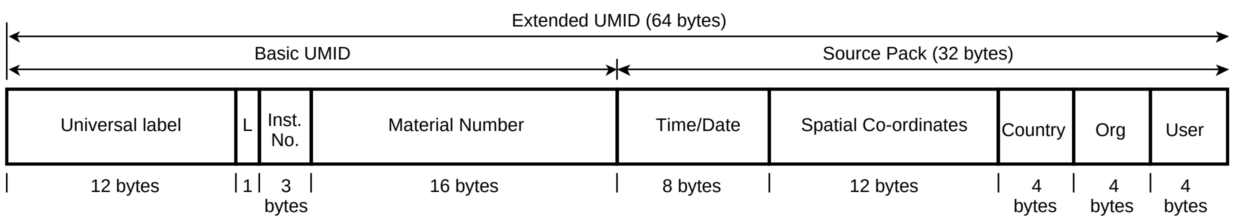

Figure 1 – Basic and extended UMID structures

NOTE – The universal label, together with text strings, are presented in character order, left to right, mapped directly onto the appropriate fields of figure 1. Number formats including the material and instance numbers together with the date/time and spatial co-ordinate fields are presented with the least significant byte leftmost in figure 1 unless otherwise defined. Randomly generated numbers defined in this standard may follow the byte order convention for numbers but by definition, the order is meaningless.

The basic UMID shall comprise the first group of 32 bytes and shall be fully implemented. The components of the basic UMID are:

- A 12-byte universal label to identify this as a SMPTE UMID. It defines the type of material which the UMID identifies and also defines the methods by which the material number and instance number are created;
- A 1-byte length value which shall define the length of the remaining parts of the UMID. The length value shall be 13_h for a basic UMID and 33_h for an extended UMID;
- A 3-byte instance number which shall be used to differentiate between different representations of material with the same material number;
- A 16-byte material number which is globally unique for every item of material. A given material number shall have the same value for related instances of the same audiovisual material only where the instance number is used to uniquely identify these related instances.

The extended UMID shall comprise a basic UMID as described above, followed immediately by a source pack. The source pack shall comprise a second group of 32 bytes and is optional. The components of the source pack are:

- an 8-byte date stamp and unit count component which shall identify the time and date of origination of the material unit with which the source pack is associated;
- a 12-byte value which defines the spatial coordinates at the time of origination of the material unit with which the source pack is associated;
- a 4-byte registered alpha-numeric code which identifies the registered country name of the originator of the material unit with which the source pack is associated;
- a 4-byte alpha-numeric code which identifies the organization name of the originator of the material unit with which the source pack is associated . The organization name is local to the country name, so organizations may use the same name provided the country name is different;
- a 4-byte alpha-numeric string which identifies the local name of the originator of the material unit with which the source pack is associated. This name is local to the organization for a given country name, so the same name may be used provided the country and organization names are different. It may be a device name or person name as determined by the organization.

NOTES

1 The source pack should be registered in the forthcoming Metadata Groups Registry.

2 The terms origination and originator above refer to physical origination and have no meaning in terms of intellectual property.

Section 5.1 defines the basic UMID and 5.2 defines the source pack.

5.1 Basic UMID

5.1.1 12-byte universal label

The first 12 bytes of the UMID shall provide identification of the UMID by the registered string value defined in table 1.

Table 1 – UMID univeral label

Byte No.	Description	Value (hex)	Meaning
1	Object identifier	06_h	Universal label start
2	Label size	$0A_h$	12-byte Universal label
3	Designation: ISO	$2B_h$	ISO registered
4	Designation: SMPTE	34_h	SMPTE registered
5	Registry category	01_h	Dictionaries
6	Specific category	01_h	Metadata dictionaries
7	Structure	01_h	Dictionary standard (SMPTE 335M)
8	Version number	VV_h	Version of the metadata dictionary (defined in SMPTE RP 210.4)
9	Class	01_h	Identifiers and locators
10	Subclass	01_h	Globally unique identifiers
11	Material type	XX_h	See 5.1.1.1
12	Number creation method	YY_h	See 5.1.1.2

NOTE – SMPTE 298M defines SMPTE labels as having a length of 16 bytes. The 12-byte UMID universal label is still a valid and unique ISO object identifier as defined in SMPTE 298M. When the UMID universal label is used in *isolation,* the 12-byte UMID universal label can be converted to a SMPTE label by padding with 4 bytes of null fill and changing the value of the label size in byte 2 from ‘$0A_h$’ to ‘$0E_h$’.

5.1.1.1 Material type identification

Byte 11 of the UL defines the material type being identified as shown in table 2.

Table 2 – Material type identification

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

NOTE – The use of material types ‘01_h’, ‘02_h’, ‘03_h’ and ‘04_h’ are deprecated for use in systems using this revised standard. These values are preserved only for compatibility with SMPTE 330M.

5.1.1.2 Number creation method identification

Byte 12 of the UL identifies the methods by which the material and instance numbers are created. This byte is divided into top and bottom nibbles.

The top nibble occupies the 4 MSBs 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 7_h in order that byte 12 conforms to ASN.1 BER short form coding rules used by SMPTE 298M.

The bottom nibble occupies the 4 LSBs and the value shall be used to define the method of instance number creation. The values used by this nibble shall be limited to the range 0 to F_h.

The methods of material number generation are defined in table 3 and the specifications of the defined methods are given in annex A.

New material number generation methods may be added as a type 1 entity as defined in SMPTE 359M. Each addition shall 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

The methods of instance number generation are defined in table 4 and the specifications of the defined methods are given in annex B.

New instance number generation methods may be added as a type 1 entity as defined in SMPTE 359M. Each addition shall provide the proposed value (within the range of values currently identified as “Reserved but not defined”) for inclusion in table 4 together with the supporting definition to be added to Annex b.

Table 4 – Identification of instance number generation method

Value (hex)	Method
0	No defined method
1	Local registration
2	24-bit PRS generator
3	Copy number and 16-bit PRS generator
4 ~ E	Reserved but not defined
F	Live stream

5.1.2 Length

The length shall be a 1-byte number with the value 13_h for a basic UMID and 33_h for an extended UMID.

5.1.3 Instance number

The instance number shall be used to separately identify instances of material where each instance shares a common material number. An instance number allows each instance to be linked to all metadata associated with that particular instance of the material. The instance number shall be set to zero whenever a new material number is created.

The 3-byte instance number shall be created by one of several means identified in table 4 and defined in annex B.

The origination of new audiovisual material requires the creation of a new material number together with an instance number which shall be set to zero as defined above.

5.1.4 Material number

The material number shall be entered for all UMIDs. The value shall be non-zero.

The 16-byte material number shall be created by one of several means identified in table 3 and defined in annex A.

The material number may be used to identify all instances of the same material. A material number allows the identified material to be linked to metadata that is common to instances of that material and vice-versa.

5.2 Source pack

The source pack is a metadata pack that identifies the source of a material unit by defining the “when, where and who (or what)” of the material unit with which it is associated.

The source pack comprises five components as illustrated in figure 1.

Any component of the source pack may be zero-filled where no meaningful value can be entered. Any zero-filled component shall be wholly zero-filled to clearly indicate to a downstream decoder that the component is not valid.

The next three clauses define each source pack component.

5.2.1 Date stamp and unit count component

This component of the source pack provides the temporal information (the “when”).

5.2.1.1 Date stamp and time-code unit count format for picture related material (deprecated)

NOTE – The use of this date stamp and time-code unit count method for picture related material is deprecated for use in systems using this revised standard. These values are preserved only for compatibility with SMPTE 330M.

In the case where the material unit is picture related and the material type identifier (byte 11 of the UL) has the value ‘01_h’, ‘03_h’ or ‘04_h’, the 8-byte date stamp and unit count shall be as defined in figure 2.

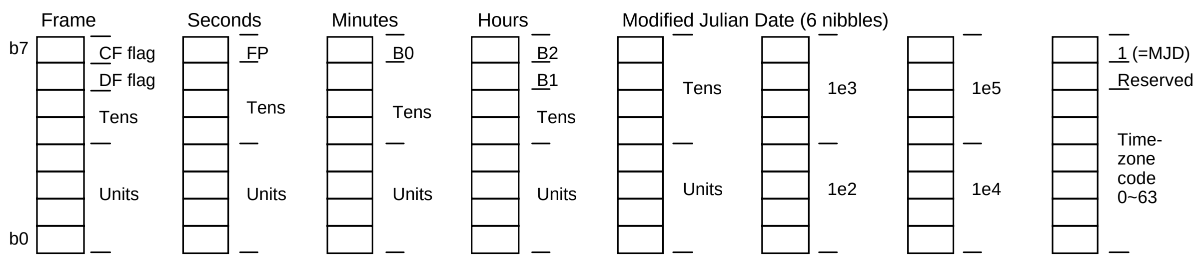

Figure 2 – Picture related date stamp and unit count data format

The first four bytes shall be as defined by SMPTE 12M and the last four bytes shall be as defined by SMPTE 309M.

A list of the abbreviated terms in figure 2 and their full names follows:

- CF flag: Color frame flag;
- DF flag: Drop frame flag;
- FP: Field phase (525/60), Binary group flag 0 (625/50);
- B0: Binary group flag 0 (525/60), Binary Group Flag 2 (625/50);
- B1: Binary group flag 1 (525/60 and 625/50).
- B2: Binary group flag 2 (525/60), Field phase (625/50);

The date format shall use the modified Julian date (MJD). The allocation of the MJD data fields and the values of the binary group flags shall be as defined in SMPTE 309M.

This format is deprecated for use with new implementations, but the information is provided for backwards compatibility.

5.2.1.2 Date stamp and unit count format for audio related material (deprecated)

NOTE – The use of this date stamp and unit count format for audio related material method is deprecated for use in systems using this revised standard. These values are preserved only for compatibility with SMPTE 330M.

In the case where the material unit is audio related and the material type identifier (byte 11 of the UL) has the value ‘02_h’,the data format for the 8-byte date stamp and unit count shall be as described in figure 3.

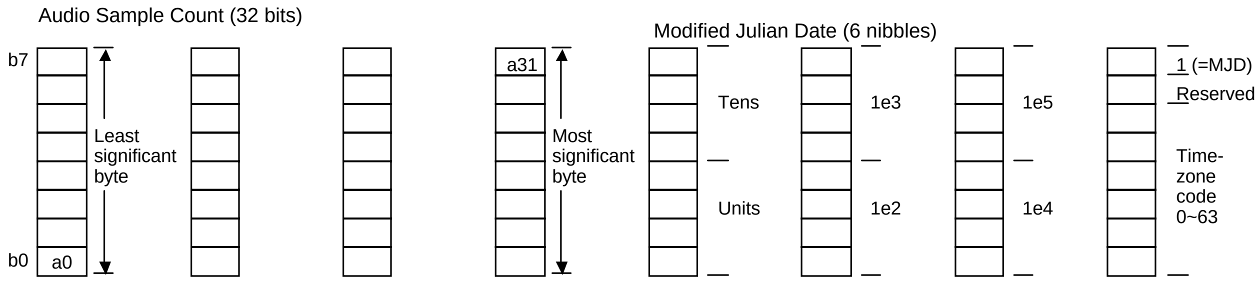

Figure 3 – Audio related date stamp and unit count data format

The first 4 bytes represent time by a unit count rate of 48 kHz with the count starting from midnight (as defined by AES3). The 32 bits for the time component allow a maximum count range of 4,294,967,296. The sample count is based on 48-kHz audio sampling which has 4,147,200,000 counts in a 24 hour period.

The date format shall use the modified Julian date (MJD). The allocation of the MJD data fields is defined in SMPTE 309M.

This format is deprecated for use with new implementations, but the information is provided for backwards compatibility.

5.2.1.3 Date stamp and unit count for all material types (single and in combination)

In the case where the material unit has a material type identifier (byte 11 of the UL) with any value of ‘05_h’ or higher, the data format for the 8-byte date stamp and unit count shall be as described in figure 4.

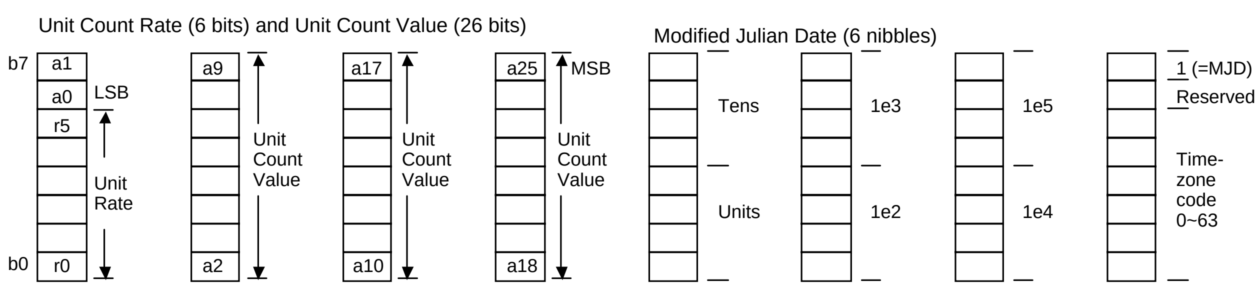

Figure 4 – Date stamp and unit count data format

Bits r0 to r5 of the leftmost byte in figure 4 shall be used to identify the unit count rate. Bits a0 to a25 of the leftmost 4 bytes in figure 4 shall be used as the unit count value within a 24-hour period starting at midnight.

The unit count rate values are defined in table 5.

New unit count rate values may be added to table 5 as a type 1 entity as defined in SMPTE 359M. Each addition shall define the proposed decimal value, the unit count rate value (within the range of values currently identified as “Reserved but not defined”) and a description of the intended application for inclusion within the table.

Table 5 – Identification of unit count rate

Decimal Value of bits r5~r0	Unit Count Rate (Hz)	Meaning
0 (default)	750	Default operation. This is equivalent to an audio frame rate of 48 kHz which, using the 192 sample AES3 block interval, results in a constant value for the 6 LSBs. In this case the 6 LSBs (r5~r0) are effectively set to zero. This rate is intended to support AES3 block rates for an audio frame rate of 48 kHz. In this case, the count will increment in steps of 3 for each new AES3 block duration. This rate is also intended to provide a default count method for all other material types and can be used as a method of calculating time within a 24 hour period.
1	500	This rate is intended to support AES3 block rates for an audio frame rate of 32 kHz and will increment in steps of 3 for each new AES3 block duration.
2	24	This material unit rate is intended to support film rates at 24 fps (frames per second).
3	24/1.001	This material unit rate is intended to support film-for-television rates at 24/1.001 fps.
4	25	This material unit rate is intended to support film and television rates at 25 fps.
6	30	This material unit rate is intended to support film and television rates at 30 fps.
7	30/1.001	This material unit rate is intended to support film-for-television and television rates at 30/1.001 fps.
8	48	This material unit rate is intended to support film rates.
9	48/1.001	This material unit rate is intended to support film-for-television rates.
10	50	This material unit rate is intended to support film and television rates.
12	60	This material unit rate is intended to support film and television rates.
13	60/1.001	This material unit rate is intended to support film-for-television and television rates.
14	72	This material unit rate is intended to support film and television rates.
16	75	This material unit rate is intended to support film and television rates.
18	90	This material unit rate is intended to support film and television rates.
20	96	This material unit rate is intended to support film and television rates.
22	100	This material unit rate is intended to support film and television rates.
24	120	This material unit rate is intended to support film and television rates.
60	44100/64 (~699)	This rate is intended to support AES3 block rates for an audio frame rate of 44.1 kHz (CD rate) and will increment in steps of 3 for each new AES3 block.
61	44100/64.064 (~688)	This rate is intended to support AES3 block rates for an audio frame rate of 44.1/1.001 kHz and will increment in steps of 3 for each new AES3 block.
63	Unspecified	The unit rate is not specified and the increments may be irregular. Its value may be determined by other means. The time of day may be incalculable using this value.
All other values	Reserved but not defined	All values not defined in the rows above are reserved but not defined.

NOTE – The 26 bits for the unit count value allow a maximum count range of 67,108,864 over a 24 hour period. This allows for a minimum unit duration of approximately 1.29 msec which is equivalent to a maximum unit rate of 775 units per second.

The rightmost four bytes define the date format which shall use the modified Julian date (MJD). The allocation of the MJD data fields shall be as defined in SMPTE 309M.

5.2.2 Spatial co-ordinate component

This component of the source pack provides the location information (the “where”).

The spatial co-ordinate component defines the location in latitude, longitude and altitude for the recording device, sensor device or the target object. The special co-ordinate value consists of three parts, each of 4 bytes, defined as follows:

- Altitude: 8 decimal numbers specifying up to 99,999,999 metres;
- Longitude: 8 decimal numbers specifying east/west 180.00000 degrees (5 decimal places active);
- Latitude: 8 decimal numbers specifying north/south 90.00000 degrees (5 decimal places active).

The reference World Geodetic System for the spatial co-ordinate component shall be OWGS84 which is the datum used by the NAVSTAR Global Positioning System.

It should be noted that although spatial co-ordinates are static for most material, this is not true for all cases. Material captured from a moving source such as a camera mounted on a vehicle may show changing spatial coordinate values.

Figure 5 illustrates the format of the spatial coordinate component.

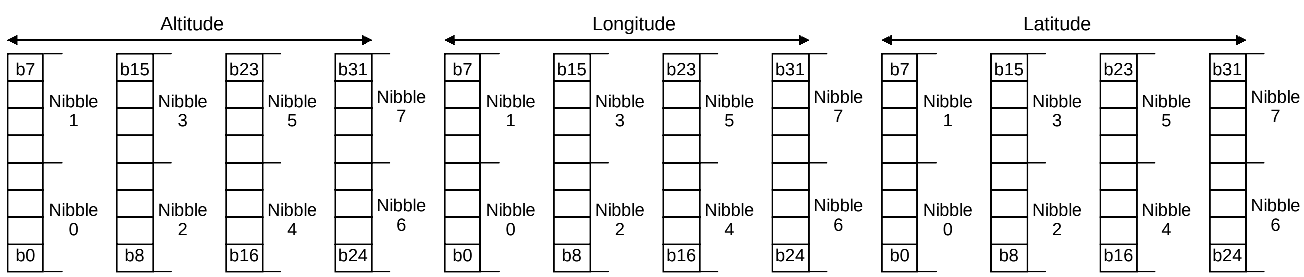

Figure 5 – 12-byte spatial coordinate format

Each nibble has a value which can represent a decimal number in the range 0~9. Values in the range A_h~F_h are reserved for special purposes as defined next.

Note that the format of each 4-byte part is little-endian which means that the least significant decimal number occupies the least significant 4 bits of the first byte (nibble 0) and the most significant decimal number occupies the most significant 4 bits of the fourth byte (nibble 7).

5.2.2.1 Altitude part

The altitude value is expressed either as a value from the centre of the earth or as a signed value relative to the sea level of the local geoid.

If nibbles 0 to 7 all lie in the range 0 to 9, the altitude value shall be measured from earth center using the full range of nibbles 0 to 7. This allows an altitude of 99,999,999 meters from earth center.

If nibble 7 lies in the range A_h to F_h, the altitude shall be measured relative to the local geoid. In this case, nibbles 7, 6 and 5 shall be interpreted as follows.

5.2.2.1.1 Nibble 7

The values of nibble 7 shall be interpreted as defined in table 6.

Table 6 – Values and definitions for nibble 7

Nibble 7 Value	Definition
A_h	Defines a positive altitude relative to the sea level of the local geoid and that the spatial coordinates are those of the sensor device location
B_h	Defines a positive altitude relative to the sea level of the local geoid and that the spatial coordinates are those of the recording device location
C_h	Defines a positive altitude relative to the sea level of the local geoid and that the spatial coordinates are those of the target object location
D_h	Defines a negative altitude relative to the sea level of the local geoid and that the spatial coordinates are those of the sensor device location
E_h	Defines a negative altitude relative to the sea level of the local geoid and that the spatial coordinates are those of the recording device location
F_h	Defines a negative altitude relative to the sea level of the local geoid and that the spatial coordinates are those of the target object location

NOTE – The geo-spatial coordinates can be located in three places depending on the operational use:

• target object location: where the geo-spatial coordinates are the location of the target object, typically that of the image or sound source. This location would be used in the case where, for example, a sensor device was capturing images at long range, but where the location of the image object was known by some means.

• sensor device location: where the geo-spatial coordinates are the location of the sensor device, typically that of the camera or microphone. This location would be used in the case of an integrated camcorder device where the geospatial coordinate sensor is located in the camcorder.

• recording device location: where the geo-spatial coordinates are the location of the recording device, typically that of a video or audio recorder which is remotely located from the camera or microphone. This location can be also used where spatial coordinates of the target object or the sensor are not available at the time of recording.

5.2.2.1.2 Nibble 6

In the nibble 6, semantics are defined for each bit as:

- b27 and b26 indicate “the number of satellites – 1” used for the measurement.
- b25 indicates if the geospatial measurements have been aided with any supportive apparatus such as a gyroscope.
- b24 indicates whether the value in nibble 5 is used for altitude measurement or for PDOP (position dilution of precision) value (including the horizontal DOP or the vertical DOP). If this bit is ‘0’, then nibble 5 provides for an altitude range of +/-999,999 metres. If this bit is set to ‘1’, then nibble 5 shall be interpreted as the PDOP value and the altitude is limited to the range +/- 99,999 metres. Note that this bit is useful only if the measurement has been calculated using more than two satellites.

NOTE – PDOP is the value obtained from more than three satellites. If only three satellites are available, it should be called either Horizontal DOP or Vertical DOP. The relationship between these values *is:* $PDOP^2 = HDOP^2 + VDOP^2$.

The values of nibble 6 shall be interpreted as defined in table 7.

Table 7 – Definition for nibble 6 of the altitude part

Nibble 6 Value	Definition
0_h	All spatial coordinates have been manually input (+/-999,999m)
1_h	The measurement has been obtained by the GPS system, but that the result is not valid and may at best be that held over from the last successful capture (+/- 999,999m)
2_h	The measurement has been obtained by only the supportive apparatus because no or only one satellite has been captured (+/-999,999m)
3_h	Reserved for future use
4_h	The measurement has been captured from two satellites only, without any supportive apparatus (+/-999,999m)
5_h	Reserved for future use
6_h	The measurement has been captured from two satellites with supportive apparatus (+/-999,999m)
7_h	Reserved for future use
8_h	The measurement has been captured from three satellites, without any supportive apparatus (+/-999,999m)
9_h	The measurement has been captured from three satellites, without any supportive apparatus (+/-99,999m)
A_h	The measurement has been captured from three satellites with supportive apparatus. (+/-999,999m)
B_h	The measurement has been captured from more than or equal to three satellites with supportive apparatus (+/-99,999m)
C_h	The measurement has been captured from more than or equal to four satellites, without any supportive apparatus (+/-999,999m)
D_h	The measurement has been captured from more than or equal to four satellites, without any supportive apparatus (+/-99,999m)
E_h	The measurement has been captured from more than or equal to four satellites with supportive apparatus. (+/-999,999m)
F_h	The measurement has been captured from more than or equal to four satellites with supportive apparatus (+/-99,999m)

5.2.2.1.3 Nibble 5

The interpretation of nibble 5 shall depend on the b24 in the nibble 6.

If the value of nibble 6 is 9_h, B_h, D_h or F_h, then the altitude value shall be limited to +/-99,999m and the value of nibble 5 shall be the PDOP (position dilution of position) value. The PDOP value defines a rounded integer number from ‘0’ upwards where a higher value indicates lower positional accuracy.

Otherwise, the altitude value shall have the range +/-999,999m and the value of nibble 5 shall be the most significant decimal value of the altitude.

NOTE – A useful reference can be found at http://www.montana.edu/places/gps/. There are many other sources of such information that may be found from searches on the World-Wide Web.

5.2.2.2 Longitude part

Nibble 7 of the longitude part shall be used to represent the combination of the east/west parameter and the most significant longitude number which may be a 0 or a 1. The LSB of nibble 7 shall be used to represent the longitude number and the remaining bits shall represent the east/west parameter. This results in the following values defined in table 8.

Table 8 – Definitions for nibble 7 of the longitude part

Value	Definition
0_h	Longitude West 0
1_h	Longitude West 1
E_h	Longitude East 0
F_h	Longitude East 1

5.2.2.3 Latitude part

Nibble 7 of the latitude part shall specify the north/south parameter as follows in table 9.

Table 9 – Definitions for nibble 7 of the latitude part

Value	Definition
0_h	Latitude North
F_h	Latitude South

Where the spatial coordinate metadata field is not used, the 12 bytes shall all be set to zero. In this case, the first 4 bytes indicate that the altitude is at the centre of the earth and clearly not a valid value.

5.2.3 Country, organization and user components

This component of the source pack provides the originator information (the “who” or “what”).

5.2.3.1 Country code component

The country code component is a code which shall be set either to the country code of the legal organization or person owning or operating the device, or to zero (per 5.2).

The 4-byte country code is an abbreviated alpha-numeric string according to the values defined in ISO 3166-1. Where the country code is less than 4-bytes, the active part of the code shall occupy the first part of the 4-bytes and the remainder shall be filled with the space character (20_h).

ISO 3166-1 3-byte alpha codes should be used, but 2-byte alpha codes or numeric codes may be used where other policies prescribe.

For non-zero values, each byte may use any alpha-numeric character from the Latin set 1 as defined by ISO/IEC 8859-1. Alpha-numeric character values shall only lie in the range 20_h to $7F_h$.

5.2.3.2 Organization code component

The organization code component is a code that represents the organisation owning or operating the device.

The organization code shall be set to either a SMPTE registered 4-byte alpha-numeric string or to zero (per 5.2).

Organizations are encouraged to obtain a registered alpha-numeric string from the SMPTE Registration Authority.

Organization codes shall not use the ~ symbol (ISO/IEC 8859-1 character number: $7E_h$) as the first character. This character is reserved for freelance operator registration.

Where a SMPTE registered organization code is less than 4 bytes, the active part of the code shall occupy the first part of the 4 bytes and the remainder shall be filled with the space character (20_h).

For all SMPTE registered values, each byte shall be an alpha-numeric character from the Latin set 1 as defined by ISO/IEC 8859-1. Alpha-numeric character values shall only lie in the range 20_h to $7F_h$.

5.2.3.3 User code component

User codes shall be assigned locally by each organization and are not SMPTE registered. User codes shall be used in conjunction with the organization code.

If the organization code is set to zero per 5.2, then the user code shall be set to zero.

If the organization code is set to a SMPTE registered value, then the user code shall be set to either a 4-byte alpha-numeric code which is determined by the organisation (and may, for example, relate to a department, person or to the device itself) or to zero (per 5.2).

For all non-zero user code values, each byte of the user code shall be an alpha-numeric character from the Latin set 1 as defined by ISO/IEC 8859-1. Alpha-numeric character values shall only lie in the range 20_h to $7F_h$.

Where the user code is less than 4 bytes, the active part of the code shall occupy the first part of the 4 bytes and the remainder should be filled with the space character (20_h).

5.2.3.4 Freelance operators

A freelance operator is an individual who is not, or does not wish to be, associated with an organization, but still wishes to be identified as an operator with a registered code.

The operator code shall be set to be either a SMPTE registered 8-byte alpha-numeric string or to zero (per 5.2).

When the country code is non-zero, freelance operators shall use their country of domicility for the country code together with the SMPTE registered operator code. Freelance operators are encouraged to obtain a registered alpha-numeric string from the SMPTE Registration Authority.

All SMPTE registered operator codes shall start with the ~ symbol (ISO/IEC 8859-1 character number, $7E_h$). The remaining 7 alpha-numeric characters shall all be filled with characters from the Latin set 1 as defined by ISO/ IEC 8859-1. Alpha-numeric character values shall only lie in the range 20_h to $7F_h$.

Where a SMPTE registered operator code is less than 8 bytes (including the ~ character), the active part of the code shall occupy the first part of the 8 bytes and the remainder shall be filled with the space character (20_h).

NOTE – If the operator code is set to zero, it will have the same value as setting the organization and user codes to zero.

Annex A (normative)
Generation of UMID material numbers

A.1 SMPTE method ('01h')

The material number generation is divided into 3 parts as shown in figure A.1.

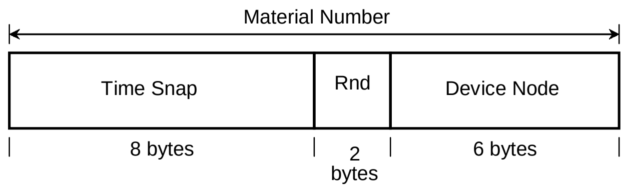

Figure A.1 – Format of the SMPTE material number

The three sub-components are:

- An 8-byte time snap value. This time snap value is referenced to the first associated unit of the material. However, it is important to note that the time snap is not a time-code or a time-stamp. It is simply a method of creating a globally unique material number in combination with the next two parts. The time snap value shall be treated as a dumb number.
- A 2-byte random number. This is included to prevent possible conflicts which might occur if the time snap value is set to the wrong time, creating the possibility of duplicate material numbers from the same device.
- A 6-byte node identifier that is a uniquely registered number for each device used for material creation.

Each of these three parts is now described in sequence.

A.1.1 Time snap

The 8-byte time snap format is divide up into a first 4-byte section used to enter a count which marks a point of time in one day and a second 4-byte section used to enter the modified Julian date. The allocation of the time snap into individual bytes is shown in figure A.2.

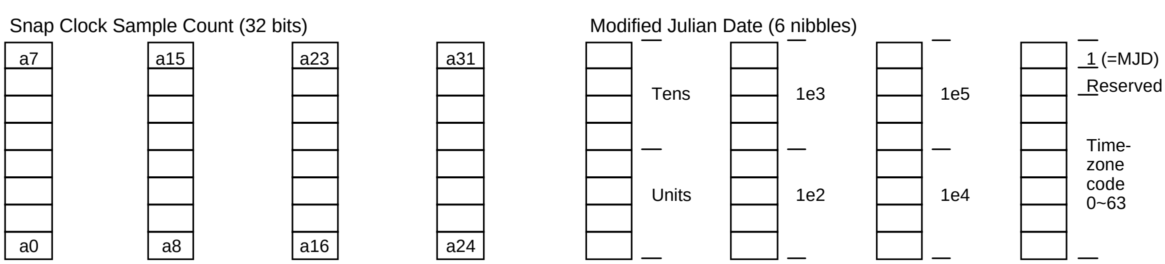

Figure A.2 – Format of the 8-byte time snap

The first 4 bytes, which define the time in a day period, are represented by the snap clock sample count. The 32 bits for the time-of-day allow a maximum count range of 4,294,967,296.

The snap clock may be any suitable clock that preferably has an integer number of clock cycles in a day. The minimum clock value shall be the material unit rate and the maximum value shall be 49.7 kHz. The minimum value is determined by the need to ensure that each material unit has a unique number and the maximum value is determined by the need to avoid overflow of the 32-bit counter in any day.

The 4-byte date format shall use the modified Julian date (MJD). The allocation of the MJD data fields is as defined in SMPTE 309M.

A.1.2 Random number

This 16-bit number is provided to help avoid duplicate numbers which may occur through incorrect setting of the clock or through a change in node identification.

The number is created by a 16-bit random number generator, whose creation method shall be linked neither to the time snap nor to the network node value.

A.1.3 Node Identifier

The node identification shall be an IEEE 6-byte address, normally defined by the IEEE 802 network host address. For systems with multiple network node addresses, only one address shall be used as the node identifier. Where a particular port is being used to output the defined material, the network node address of that port shall be used.

NOTE – The node identifier bytes are placed left to right in figure A.1 according to network byte order.

A.2 UUID/UL method (‘02h’)

This method allows either a UUID according to ISO/IEC 11578 annex A or a SMPTE Universal label to be accommodated in the same space with a guarantee that the values are always different.

According to ISO/IEC 11578 annex A, the UUID “variant” bits (see below) ensure that the MSB of the 9th byte of every UUID is always set to a ‘1’. A UUID shall be mapped into the material number in its defined byte order.

According to SMPTE 298M, the MSB of 1st byte of every UL is always set to ‘0’. A UL shall be mapped into the Material Number with the last 8 bytes swapped with the first 8 bytes. Thus the 1st byte of the UL will be placed in 9th byte of the material number and vice-versa.

These two mappings ensure that the MSB of 9th byte in the material number is always a ‘0’ for a UL and a ‘1’ for a UUID. Thus, the value of a UL and a UUID in the Material Number will always be different.

The defining standard for creating a UUID is ISO/IEC 11578, annex A.

The following sub-sections provide an informative description of the UUID.

A.2.1 Time-snap (informative)

The first 60 bits of the 64-bit available space is specified according to the version identification given in the last 4 bits.

The last 4 bits define the version of UUID as follows:

Version	bit 0	bit 1	bit 2	bit 3
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0

Version 0 is not a valid number.

Versions not listed up to the value of 15 are reserved but not defined.

A.2.1.1 Version 1

The time snap is the output of a 60-bit binary counter clocked by a 10-MHz clock. The time snap is a count of 100 ns intervals since midnight (00:00:00.00) on 15 October 1582 (the date of Gregorian reform to the Christian calendar) based on the Greenwich meridian. This time snap is known as universal coordinated time (UTC). Local time may be used in exceptional cases provided the application of time is consistent and time-zone changes are not expected in normal operation.

NOTE – The least significant 8 bits of the counter are placed in the leftmost byte.

A.2.1.2 Version 2

This version is reserved for DCE security versions with embedded POSIX UIDs.

NOTE – This mode is not recommended.

A.2.1.3 Version 3

The 60 bits are created from an unspecified name.

A.2.1.4 Version 4

The 60 bits are created by a random number generator of an unspecified type.

A.2.2 Clock sequence and variant (informative)

The 2 MSBs of the first byte identify the UUID variant. The remaining 14 bits are used for a random number used to help avoid duplicate numbers which may occur through incorrect setting of the clock or through a change in node identification. The random number is created by a 14-bit random number generator which shall be created by a method linked neither to the 60-bit counter nor the node identification.

A.2.3 Node Identification (informative)

The node identification is an IEEE 6-byte address, normally defined by the IEEE 802 network host address. For systems with multiple network node addresses, any one address can be used.

A.3 Masked method ('03h')

In certain applications, the uniqueness of the material number is the *only* desired function. The material originator may not wish to reveal any information about the time, location, or equipment on which the material originated, for privacy or other security reasons. (investigative journalism provides an example.) The masked method creates a material number which is statistically unique to very high probability.

These masked material numbers are effectively random and do not support sorting or any other meaning-based processing.

A.3.1 Reference masking method

The reference masking method post-processes a material number created by either of the methods defined in A.1 or A.2 using additional pseudorandom data and the MD5 hash function (IETF-RFC1321).

The 128-bit "clear" material number should be computed in accordance with A.1. An additional 128 bits of local data are appended to the clear material number. The resulting 256-bit string is processed with the MD5 algorithm, producing a 128-bit result. This result is the masked material number.

The local data may be a fixed string of 128 "0" bits. Additional security implications are discussed below.

A.3.2 Alternative masking methods

The masked material number is an unpredictable number uniformly distributed over the range 0 thru 2^{128}-1. Its effectiveness as a unique identifier relies on this uniform random distribution, and the exact method of its generation is not important. Therefore, the use of the reference masking method is *not* normative, and any method providing an equivalent level of unpredictability and uniformity of distribution may be used with the "masked method" value in the "number generation method" field of the UMID universal label (reference table 1 in 5.1.1).

A.3.3 Security implications (informative)

The MD5 function is designed as a "secure hash function", for which it is computationally infeasible to derive the input value from the output value. Therefore, a single blinded material number does not reveal the IEEE 802 Node ID of the originator, nor does it reveal the time snap. However, as in many hash applications, a *brute-force* attack is possible in which an opponent tries all possible values of clear material number, computes the corresponding MD5 hash, and compares this result to the blinded material number he wishes to identify. If the opponent can independently guess likely values for the time snap and node ID, it may be quite practical to brute-force the blinded material number. This attack is especially powerful if another application, producing clear material numbers, also runs (concurrently or sequentially) on the same hardware as the sensitive application.

For this reason, particularly sensitive editorial content originators should employ additional unpredictable data (sometimes called *salt*) to resist brute-force attacks. In the reference masking method, 128 bits of salt are suggested, but alternative masking methods may use more. Methods of deriving unpredictable data are discussed in IETF RFC-2518 and in standard cryptography texts.

A.4 IEEE 1394 network method ('04h')

The material number generation is divided into 3 parts as shown in figure A.3.

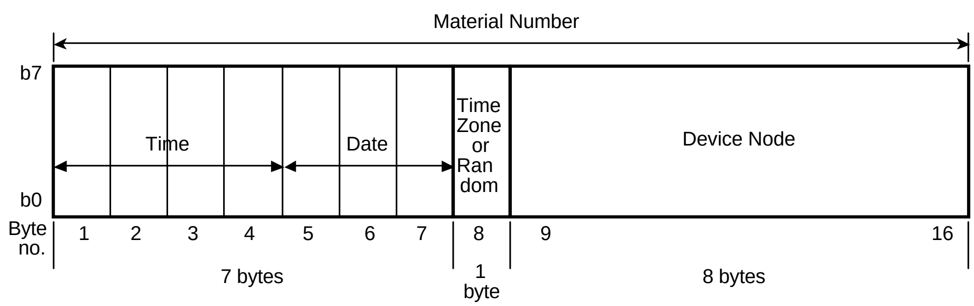

Figure A.3 – Format of the SMPTE material number

The byte values shall be defined as follows:

1) Bytes 1 to 4 define the time in a day period are represented by the snap clock sample count. The 32 bits for the time-of-day allow a maximum count range of 4,294,967,296. The snap clock may be any suitable clock which preferably has an integer number of clock cycles in a day. The minimum clock value shall be the material unit rate and the maximum value shall be 49.7 kHz. The minimum value is determined by the need to ensure that each material unit has a unique number and the maximum value is determined by the need to avoid overflow of the 32-bit counter in any day.

2) Bytes 5 to 7 provide the Modified Julian Date (MJD) information as a 3-byte date value per A.1.1.

3) Byte 8 may take one of two forms based on the value of bit 6. Bit 7 of byte 8 shall be set to '1' to identify that the date is the MJD (and not the DDMMYY) format. When bit 6 of byte 8 is '0' the remaining bits, b5 to b0, are assigned as per SMPTE 309M to give the time-zone information. When bit 6 of byte 8 is '1' bits b5 to b0 form a 6-bit random number serving to reduce the likelihood of the same number being created more than once when a device's clock is incorrectly set. In accordance with SMPTE 309M, the time shall be set to UTC.

4) The device node ID can use the EUI-64 network node ID value defined by IEEE 1394 to fill the available 8-byte space. If the network node number is EUI-48 rather than EUI-64, then the convention is to set the 4th and 5th bytes of the address (bytes 12 & 13) to the fixed value of 'FF_h' for easy detection. The first 3 bytes of the EUI-48 address are then mapped to bytes 9 to 11 and the last 3 bytes of the EUI-48 address are mapped to bytes 14 to 16.

Annex B (normative)
Generation of UMID instance numbers

B.1 Local registration ('1h')

The instance number for every instance of a given material number is obtained from a local registry which records all registered instance numbers for the material number in question.

Users should be aware that this method needs a carefully managed local environment - for instance, material may be taken off-site and copies made with duplicate instance numbers. Thus, this method should only be used where full managerial control of the local environment can be guaranteed.

B.2 24-bit PRS generator ('2h')

Any suitable psuedo-random sequence (PRS) generator polynomial may be used provided it has a maximal length of 16,777,215 clock cycles. At the point of creating a new instance of the material, the 24-bits from the PRS generator are sampled to gain a new instance value.

PRS generators shall not allow a zero value.

NOTES

1 Any suitable seed may be used to start the pseudo-random sequence (PRS) 24-bit generator.

2 The PRS generator should use a free-running clock having no time relationship with the clock used to generate the sampling strobe.

3 The PRS generator clock frequency should be greater than 10 kHz.

4 The number of feedback taps resulting from the PRS generator polynomial should be between 8 and 16 to ensure the random nature of the sequence.

B.3 Copy number and 16-bit PRS generator ('3h')

The first (leftmost) byte of the instance number shall be incremented for each new copy.

The remaining two bytes shall be created from a 16-bit pseudo-random sequence (PRS). The constraints on the 16-bit PRS generation shall be the same as that defined in clause B.2 with the exception that the PRS generator has a maximal length of 65535 clock cycles and that the recommended number of feedback taps lies between 4 and 12 the ensure the random nature of the sequence.

B.4 Live stream ('Fh')

This instance number method shall be used to identify that the material is a direct live signal source from a material creation device.

NOTES

1 This instance number method differentiates original material direct from a camera, microphone or source device from reproduced material that has a UMID with a new material number and a zero instance number.

2 It is possible that there may be several sources of the same live stream via parallel feeds and care needs to be exercised in its application; see SMPTE RP 205.

Annex C (normative)
Text representation of the UMID

Some applications do not support the use of byte-coded data such as used by the UMID. In these applications, failure can occur due to false interpretation of the byte values, either individually or in particular sequences. This annex provides for the representation of the UMID as a simple ISO 7-bit character text string both to aid human readability and to ensure that byte values are not falsely decoded in a textual format.

It is recommended that the UMID, whether as a basic UMID or as an extended UMID, be represented as a simple string of the hexadecimal representation of each byte of a UMID. Applications shall provide a means of identification of the hexadecimal representation.

By default, the hexadecimal representation should be preceded by the identifier '0x'. This default method of identification will result in 66 hexadecimal characters to represent a basic UMID as follows:

0x060A2B340101010105010101D13 … etc.

The hexadecimal letters should be encoded as upper case (A…F). However, decoders shall accept hexadecimal letter representations in both upper and lower case (A…F and a…f).

Annex D (informative)
Documentation of legacy UMID generation

This method of material number generation documents a commercial method (known as OMF, Open Media Framework) for generating a basic UMID in existing use in the field. This method is deprecated for new designs.

D.1 First 16 bytes

The 12-byte key for this identifier differs from that defined in this standard and thus provides ready differentiation from the specification in this standard. The value of the first 16 bytes, in period separated hexadecimal notation, is:

06.0C.2B.34.02.05.11.01.01.UT.10.00.13.00.00.00

NOTES

1 Byte 2 has a value of ‘0Ch’, This value is incorrect for a Label size of 12 bytes.
2 Byte 5 has a value of ‘02h’. This value is reserved for any forthcoming essence dictionary and is thus invalid.
3 Byte 10 (“UT”) means “UMIDType” and has an unspecified value.
4 Byte 13 has the correct value of ‘13h’
5 Bytes 14~16 define an instance number of zero.

NOTE – This 16-byte value must be precluded from any forthcoming essence dictionary in order to avoid potential duplication of meaning.

D.2 Second 16 bytes

The second part of the UMID is created as follows:

Byte number	Value
16 to 21	06.0E.2B.34.7F.7F (defined values)
22 & 23	Prefix (2 bytes, company specific)
24 to 27	Major (4 bytes, derived from time of day)
28 to 31	Minor (4 bytes, derived from processor tick count)

Annex E (informative)
Bibliography

SMPTE 335M-2001, Television — Metadata Dictionary Structure

SMPTE RP 205-2000, Application of Unique Identifiers in Production and Broadcast Environments

IEEE 802-1990, IEEE Standards for Local and Metropolitan Area Networks: Overview and Architecture

EBU/SMPTE Task Force for Harmonized Standards for the Exchange of Programme Material as Bit Streams, Final Report: Analyses and Results, July 1998, SMPTE J. 107(9):603-815; 1998 September

RFC 2518, HTTP Extensions for Distributed Authoring – WEBDAV, Internet Engineering Task Force, 1999

SMPTE Registration Authority may be found at: <www.smpte-ra.org>

XML Schema definitions may be found at: http://www.w3.org/TR/xmlschema-2/