

# SMPTE STANDARD

## for Television — SMPTE Labels Structure



### 1 Scope

The SMPTE labels structure defined by this standard covers the use of SMPTE universal labels as individual items where the meaning of each label is conveyed by the content of the universal label alone. This standard does not define the use of SMPTE universal labels as keys for the coding of data items or data groups as defined in SMPTE 336M.

The associated SMPTE labels registry defines individual label values together with their definition and description.

The labels structure and labels registry shall be used together as a pair. Neither shall be used in isolation. Applications of individual registry entries may vary but, when used, shall conform to the definitions used in this standard and the associated labels registry (RP 224).

### 2 Normative references

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

SMPTE 336M-2001, Television — Data Encoding Protocol using Key-Length-Value

SMPTE 359M-2001, Television and Motion Pictures — Dynamic Documents

### 3 SMPTE Labels Structure

The SMPTE labels structure provides a common structure for the cataloguing of SMPTE labels that shall be used by the labels registry. This structure ensures that individual SMPTE labels are arranged in an ordered hierarchical framework for consistency between different SMPTE label definitions. The use of SMPTE labels is defined in clause 6 of SMPTE 336M.

The labels registry defines an individual entry as SMPTE universal label that is numerical (and hence language independent) and unique. The first seven bytes of every SMPTE label are defined by this standard. The eighth byte is an incrementing number that identifies the version number of the SMPTE labels registry. The second group of eight bytes of the SMPTE label shall be used to provide the semantic definition of the label.

#### NOTES

1 As per SMPTE 298M, all SMPTE administered universal labels shall be 16 bytes. All individual words lie within the hexadecimal range of 00<sub>h</sub> to 7F<sub>h</sub>.

2 Any individual word with a value of 00<sub>h</sub> is assumed to act as a terminator for parsing the SMPTE label. Therefore, if present, all occurrences of this value shall be constrained to the least significant words of all SMPTE labels.

### 3.1 SMPTE label — First half

The individual fields of each 16-byte SMPTE label are defined in clause 3.1 of SMPTE 336M. Table 3 of SMPTE 336M defines the following field values:

**Table 1 – Definition of the first part of each SMPTE label**

Byte position	Description	Value	Meaning
1	Object identifier	06 <sub>h</sub>	Fixed start byte
2	UL length	0E <sub>h</sub>	Byte length of the following label
3	UL code	2B <sub>h</sub>	ISO organization
4	UL subcode	34 <sub>h</sub>	SMPTE
5	Registry category	04 <sub>h</sub>	Labels
6	Registry designator	01 <sub>h</sub>	Labels registries
7	Structure designator	01 <sub>h</sub>	Labels structure (this standard)
8	Version number	01 <sub>h</sub> –7F <sub>h</sub>	Incrementing labels registry version number
9-16	Item designator	Defined by labels registry	See the labels registry

### 3.2 SMPTE label — Second half

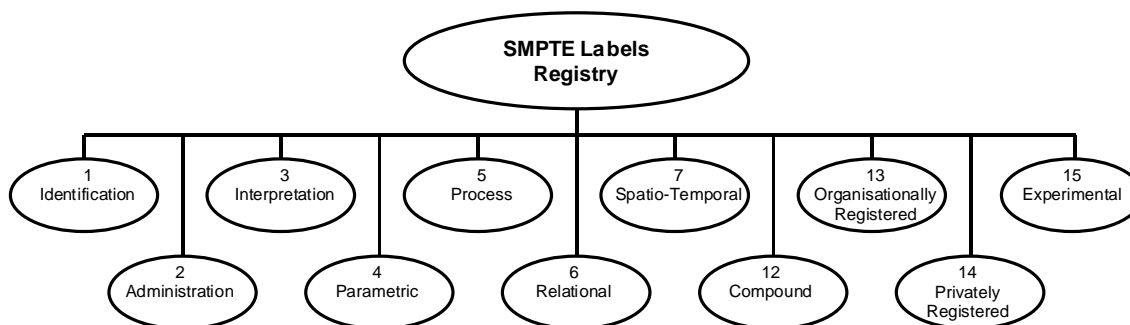
The SMPTE labels registry defines the organization of the second half of each label value and is organized into nodes and leaves. The labels registry is formed from class nodes that provide the registry structure and below each of these are further nodes at each subclass. To aid the management of the registry, these nodes and subnodes are assigned with no use as a label so as to give clear breaks in the registry structure. Entries within a subclass form leaves, which are the registered label values themselves.

Lower levels of the labels structure are defined by SMPTE RP 224 and, as far as practical, follows a similar structure to the metadata dictionary node structure indicated in SMPTE EG 37.

#### 3.2.1 Individual label classes

The SMPTE labels registry is organized into a hierarchical structure, where each registry node is assigned to an initial list of labels classes as described below. This list is summarized in figure 1.

- Class 1: Identification and location labels
- Class 2: Administration labels
- Class 3: Interpretive labels
- Class 4: Parametric labels
- Class 5: Process labels
- Class 6: Relational labels
- Class 7: Spatio-temporal labels
- Class 12: Compound labels
- Class 13: Organizationally registered labels for public use
- Class 14: Organizationally registered labels as private
- Class 15: Experimental labels



**Figure 1 – SMPTE Labels Registry Class Structure**

This list mirrors the classes defined by SMPTE 335M and adds one more compound class to cover labels that tag entities with a diverse meaning.

The number of labels registry classes can be extended in the future to a maximum of 127. Although labels registry classes can be populated with any value, additional new classes may be created up to the limit to deal with specific labels characteristics or attributes.

### **3.3 Labels registry class descriptions**

#### **3.3.1 Class 1: Identification and location**

SMPTE labels in this class shall primarily carry information about identification and location. Labels in this class shall include information about identifiers and locators with both global and local scope. Examples of label kinds in this class are:

- Object identifiers
- Device identifiers
- Media locations

#### **3.3.2 Class 2: Administration**

SMPTE labels in this class shall primarily identify administrative or business information including authorization usage and encryption. Examples of label kinds in this class are:

- Security classification
- Encryption types

#### **3.3.3 Class 3: Interpretive**

SMPTE labels in this class shall primarily carry information that defines how data is interpreted. Labels in class 3 shall be principally used for identifying catalogues, thesauri and other items which can be used to provide terms for either human or machine application. Examples of label kinds in this class are:

- Fundamental (such as length and time systems)
- Descriptive (human-assigned collections of key-words)
- Categorization (identifying a category set)
- Descriptors (machine-assigned or computed)

### 3.3.4 Class 4: Parametric

SMPTE labels in this class shall primarily carry information that describes the technical characteristics of devices or systems. Examples of label kinds in this class are:

- Essence encoding characteristics
- Container encoding characteristics
- Test parameter sets
- Compression coding characteristics

### 3.3.5 Class 5: Process

SMPTE labels in this class shall primarily carry information that describes how the essence was originated or processed. Examples of label kinds in this class are:

- Process indicators
- Manipulation kinds
- Downstream processing history types
- Enhancement or modification processes
- Processor settings (device-specific)

### 3.3.6 Class 6: Relational

SMPTE labels in this class shall primarily carry information that describes a class of relationships between objects. Examples of label kinds in this class are:

- Generic relationships
- Relative relationships
- Object to object relationships
- Relationship structures

### 3.3.7 Class 7: Spatio-temporal

SMPTE labels in this class shall primarily carry information about aspects of time, place, or space. Examples of label kinds in this class are:

- Position and space vector systems
- Image positional information
- Distance and angle specifications
- Delay and latency kinds

### 3.3.8 Class 12: Compound

SMPTE labels in this class shall primarily carry information that combines information from multiple classes.

This class is intended for SMPTE labels whose role is to identify entities that cannot be accommodated in a single class.

### 3.3.9 Class 13: Organizationally registered for public use

SMPTE labels in this class shall be defined and registered by a specific user organization or individual and are, therefore, reserved and managed separately from the other classes of labels. The definition may allow or require these labels to carry standard information from single classes defined here, or from a combination of classes.

Information about publicly registered labels shall appear in the appropriate sections of the published labels registry. Class 13 labels shall be managed by the SMPTE Registration Authority and their approval shall be consistent with SMPTE 359M.

### **3.3.10 Class 14: Organizationally registered as private**

SMPTE labels in this class shall be defined and registered by a specific user organization or individual and are, therefore, reserved and managed separately from the other classes of labels. The definition may allow or require these labels to carry information from the other primary classes, or publicly registered labels (class 13), or organizationally registered labels (class 14), or any other combination.

Information about organizationally registered labels shall not be made public, but the nodes and subnodes shall be publicly identified in label registry and shall be reserved for use by the registered organization. Class 14 metadata shall be managed by the SMPTE Registration Authority and its approval shall be consistent with SMPTE 359M.

### **3.3.11 Class 15: Experimental**

SMPTE labels in this class shall carry information whose definition and use does not need to conform to the definitions in the labels registry. Class 15 labels are intended for use in research or other, limited access, experimental environments where experimentation with new labels and applications does not depend on strict conformance to approved standards and which remain private within a test or laboratory environment.

## **4 SMPTE labels registry management and compatibility requirements**

To ensure the reliable and correct interpretation of existing labels registry values, only additions shall be permitted. Deletions or changes to leaf entries (other than purely editorial) are not permitted. This addition process shall be speedily carried out and documented in accordance with SMPTE 359M by the SMPTE Registration Authority. It shall occur immediately on request from the appropriate SMPTE technical committee and shall be administered and scrutinized with a light but formal touch to ensure minimal delay in the availability for use of a newly required SMPTE label. The version number of the registry shall be incremented each and every time an addition (or groups of additions) is made since this is critical to ensuring the operational compatibility of existing decoders. The incrementing of the version number shall not prevent use of existing labels by a decoder that conforms to the prior version.

It is inevitable given the above addition process that eventually the SMPTE labels registry will become cluttered with legacy entries to the point where the responsible SMPTE technical committee determines it has reached the limit of its usefulness. At this stage, or when other changes to the labels registry, to an existing approved labels structure, or to relationships between labels classes that prevent backward compatibility, are necessary, new documents consisting of both a labels structure and labels registry shall be created. The structure and contents of the new labels registry shall be made readily accessible on-line by the SMPTE Registration Authority to allow upgrades to decoders. The superceded standard shall then undergo no further revision unless essential under the SMPTE five year rule.

### **4.1 SMPTE labels registry format**

The SMPTE labels registry shall comprise at least the following fields. Additional fields may be added to the registry at any time for any purpose such as verification or cross-checking. Any language sensitive fields shall use International English.

#### **4.1.1 Registry type**

This field shall be used to define whether the registry entry is a node or a leaf. Node entries are informative and are used to provide structure to the registry. Leaf entries are normative and shall be used to define the individual SMPTE label values.

#### 4.1.2 Registry designator

This field shall be used to define the first 8 bytes of the SMPTE label. The value of the first 7 bytes shall be as defined in table 1. Byte 8 shall define the version of the registry at the point of entry and shall lie in the range 01<sub>h</sub> to 7F<sub>h</sub>.

#### 4.1.3 Registry key

This field shall comprise the last eight bytes of the SMPTE label in the form:

xx.xx.xx.xx.xx.xx.xx.xx

where the value xx shall be the hexadecimal value of the byte. All 8 bytes shall be filled with a value between 00<sub>h</sub> to 7F<sub>h</sub>.

#### 4.1.4 Reference name

This field shall comprise the reference name values for each SMPTE label. Each reference name value should be unique in order to avoid confusion when searching this registry. Reference names are required for each registry leaf entry and are optional for each registry node entry.

#### 4.1.5 Description

This field shall comprise a human-readable description of the SMPTE label whether as a node or a leaf.

#### 4.1.6 Defining standard

This field shall define, where applicable, the standard or recommended practice that the SMPTE label values identify. In many cases, it may be defined in the node entry that precedes the enumerated SMPTE label values for the purpose of clarity.

### 4.2 SMPTE labels registry availability

The SMPTE labels registry shall be available in an on-line electronic form in a defined electronic publishing format, such as XML, with an accompanying document type definition.

The latest version of the SMPTE labels registry shall be made available on the SMPTE Web site in a non-editable format (PDF or equivalent). It is preferred that a minimum of the two immediate previous versions also be available in a clearly indicated archive.

#### Annex A (informative)

##### Bibliography

SMPTE 210.4-2002, Metadata Dictionary Registry of Metadata Element Descriptions

SMPTE RP 224, SMPTE Labels Registry

SMPTE EG 37-2001, Node Structure for the SMPTE Metadata Dictionary