SMPTE Public Committee Draft

Extensible Time Label — Items

A list of all public CDs can be found on the SMPTE website

https://www.smpte.org/public-committee-drafts#listing

Table of Contents Page

Foreword

SMPTE (the Society of Motion Picture and Television Engineers) is an internationally-recognized standards developing organization. Headquartered and incorporated in the United States of America, SMPTE has members in over 80 countries on six continents. SMPTE"s Engineering Documents, including Standards, Recommended Practices, and Engineering Guidelines, are prepared by SMPTE"s Technology Committees. Participation in these Committees is open to all with a bona fide interest in their work. SMPTE cooperates closely with other standards-developing organizations, including ISO, IEC and ITU.

SMPTE Engineering Documents are drafted in accordance with the rules given in its Standards Operations Manual. This SMPTE Engineering Document was prepared by Technology Committee 32NF Network/Facilities Infrastructure.

Normative text is text that describes elements of the design that are indispensable or contains the conformance language keywords: "shall", "should", or "may". Informative text is text that is potentially helpful to the user, but not indispensable, and can be removed, changed, or added editorially without affecting interoperability. Informative text does not contain any conformance keywords.

All text in this document is, by default, normative, except: the Introduction, any section explicitly labeled as "Informative" or individual paragraphs that start with "Note:"

The keywords "shall" and "shall not" indicate requirements strictly to be followed in order to conform to the document and from which no deviation is permitted.

The keywords "should" and "should not" indicate that, among several possibilities, one is recommended as particularly suitable, without mentioning or excluding others; or that a certain course of action is preferred but not necessarily required; or that (in the negative form) a certain possibility or course of action is deprecated but not prohibited.

The keywords "may" and "need not" indicate courses of action permissible within the limits of the document.

The keyword "reserved" indicates a provision that is not defined at this time, shall not be used, and may be defined in the future. The keyword "forbidden" indicates "reserved" and in addition indicates that the provision will never be defined in the future.

A conformant implementation according to this document is one that includes all mandatory provisions ("shall") and, if implemented, all recommended provisions ("should") as described. A conformant implementation need not implement optional provisions ("may") and need not implement them as described.

Unless otherwise specified, the order of precedence of the types of normative information in this document shall be as follows: Normative prose shall be the authoritative definition; Tables shall be next; then formal languages; then figures; and then any other language forms.

Introduction

This section is entirely informative and does not form an integral part of this Engineering Document.

As media production workflows embrace non-linear processes and multiple sources, the ability to temporally align elements is essential for editorial processes and may prove to be useful in other contexts. Since its inception in the 1970s, the ST 12 family of standards has been updated to meet changing needs but does not provide all that is required for the future. Additionally, applications outside of media have become dependent on ST 12 making it desirable that any replacements offer similar cross-industry versatility.

The Extensible Time Label (TLX) is a data construct that can be adapted to multiple applications, not necessarily time-related or media-related. The TLX environment supports globally-unique identification, and the flexibility to create labels of varying content to meet the needs of many applications.

The structure of a TLX label as described in SMPTE 2120-1 specifies that a label is an object having one or more TLX items. SMPTE 2120-2 expands on SMTPE 2120-1 by specifying TLX items. It does not address specific configurations of TLX items that a given application can adopt or require as these are considered out of scope for this document.

This document enumerates the various TLX items and attributes that can be used in a TLX label. This list is not exhaustive, and it is intended that this document be amended as necessary to add definitions of other TLX items and attributes as needs arise. A formal JSON Schema for the TLX label, its items and attributes is included as an informative non-prose element.

[Editors notes: The following paragraph will be replaced with the appropriate patent information during the SMPTE Headquarters publication process.]

At the time of publication, no notice had been received by SMPTE claiming patent rights essential to the implementation of this Engineering Document. However, attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. SMPTE shall not be held responsible for identifying any or all such patent rights.

3 Scope

This document specifies a number of TLX items as defined by SMPTE ST 2120-1 that can be assembled within TLX labels. Each TLX item is an independent entity.

The following are not in scope for this document:

- Specific configurations of labels built to use for a particular application
- Procedures or methodologies for labeling of a media unit.

4 Normative References

The following standard contains provisions that, through reference in this text, constitute provisions of this standard. Dated references require that the specific edition cited shall be used as the reference. Undated citations refer to the edition of the referenced document (including any amendments) current at the date of publication of this document. All standards are subject to revision, and users of this engineering document are encouraged to investigate the possibility of applying the most recent edition of any undated reference.

IEEE Std 754™-2008 IEEE Standard for Floating-Point Arithmetic

IEEE Std 1588™-2019 IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems

IETF RFC 4122 A Universally Unique Identifier (UUID) URN Namespace

SMPTE ST 12-1:2014 Time and Control Code

SMPTE ST 12-2:2014 Transmission of Time Code in the Ancillary Data Space

SMPTE ST 12-3:2016 Time Code for High Frame Rate Signals and Formatting in the Ancillary Data Space

SMPTE ST 2059-2:202x SMPTE Profile for Use of IEEE-1588 Precision Time Protocol in Professional Broadcast Applications

SMPTE ST 2120-1:202x Extensible Time Label - Structure

5 Terms and Definitions

For the purposes of this document, the following terms and definitions apply.

local time

time (and date) as chosen by the facility's system administrator and implemented by the system

Note: This is typically the calendar date and time-of-day in common use in the locality. It usually includes an offset from UTC indicating the time zone, accumulated Leap Seconds by the rules of UTC, and, where observed, daylight saving time (summer time) by the rules of the responsible authority.

[SOURCE: SMPTE ST 12-1:2014, modified in format only]

time-like value

value having the dimension of time, but not necessarily representing real time

timecode

time code

time and control code

time address, flag bits, and binary groups, as specified in SMPTE ST 12-1

4 Mathematical operators

4.1. **floor(x)**

rounds the element 'x' to the nearest integer towards negative infinity

4.2. **ceiling(x)**

rounds the element 'x' to the nearest integer towards positive infinity

4.3. **%**

modulus operator (remainder from the division of two integers)

5 Derived data types

5.1. General

Primitive and structural data types are specified in ST 2120-1, which are consistent with the JSON data interchange format as described in IETF RFC 8259. Among these is the structural type of object, a data type representing an unordered collection of name-value pairs. A time label itself is an object: An unordered collection of one or more TLX items, each of which is a name-value pair. Each TLX item is also an object: An unordered collection of one or more TLX attributes, each of which is a name-value pair.

For the convenience and consistency of specification and implementation, section 5.2 specifies derived data types, which are compositions of and/or additionally constrained versions of the primitive and structural data type.

5.2. Derived data types

Primitive and structured data types were outlined in ST 2120-1. For ST 2120-2 the following definitions use the derived data types listed in Table 1

Table 1 - Derived data types

Data type	Definition	Constraints
integerType	number	Value shall be an integer in the range $[-2^{53} + 1, 2^{53} - 1]$
hexstringType	string	Value shall contain only characters representing hexadecimal digits [0-9a-f]. Unless otherwise specified, the first and last characters correspond to the four most and four least significant bits of a binary value, respectively.
rationalType	array of two integers, the first integer representing a numerator and the second a denominator. Both area numerator and a denominator are required.	The second value, the denominator, shall not equal zero.

NOTE: The constraint regarding the range of values for integerType is to ensure that numbers can be exactly represented in common programming languages that use IEEE 754 binary64 representation.

6 TLXptpTimestamp

6.1. Description

TLXptpTimestamp is an item that provides a media unit with a **ptpTime** representing the time of creation of the TLX label, measured in terms of duration since the PTP Epoch. The **localOffset** attribute allows for conversion to the local time. The **isLeapSecond** attribute provides information needed to present a correct representation of local time during the last second of a sixty-one second minute during a positive leap second, in accordance with ISO 8601-1.

6.2. Attributes

The attributes of TLXptpTimestamp shall be as specified in Table 2.

Table 2 - TLXptpTimestamp attributes

Name	Type	Constraints	Required/ Optional	Description
ptpTime	array of two integers	first integer s: $0 \le \mathbf{s} < 2^{48}$ second integer ns: $0 \le \mathbf{ns} < 10^9$	required	**ptpTime** represents seconds (**s**) and nanoseconds (**ns**), which is consistent with Struct Timestamp as specified in IEEE 1588-2019
localOffset	integerType	in the range [-2^{31}, $2^{31}-1$]	optional	add to **ptpTime** to produce local time on a local timescale
isLeapSecond	boolean		optional; default = false	where set to true, **s** is a positive leap second on the local timescale

If provided, adding **localOffset** to **ptpTime** determines the local time at which the label was generated (measured from 1970-01-01).

ST 2059-2 specifies SM TLV metadata having the integer components of *currentLocalOffset*, *timeOfNextJump*, *jumpSeconds*, and the boolean component *leapSecondJump*. Where derived from ST 2059-2 SM TLV metadata, **localOffset** and **isLeapSecond** shall be computed so as to produce the same result as this pseudocode:

```
if s is equal to timeOfNextJump then:
     localOffset = currentLocalOffset + jumpSeconds
else:
     localOffset = currentLocalOffset

if s is equal to timeOfNextJump then:
     isLeapSecond = leapSecondJump
else:
     isLeapSecond = false
```

NOTE 1. When local time corresponds to any time zone having an offset from UTC0 that is an integer multiple of 15 minutes and the magnitude of **localOffset** is less than 86400 (i.e., < 24 hours), then (**localOffset** % 900) added to **ptpTime** faciltates determining the UTC0 time at which the label was generated.

NOTE 2. **ptpTime** can coincide with the capture, generation, retrieval time, or rendering of the media unit. In cases where **ptpTime** represents live capture time, it could be suitable for synchronizing the labelled media unit with other live-captured material and metadata.

7 TLXmediaCount

7.1. Description

TLXmediaCount is an item that provides a **count** value for a unit of media. Where provided, **rate** allows **count** to be interpreted as a time-like value.

7.2. Attributes

The attributes of TLXmediaCount shall be as specified in Table 3.

Table 3 - TLXmediaCount attributes

Name	Type	Constraints	Required/ Optional	Description
count	integerType	≥ 0	required	index of a media unit in a sequence
rate	rationalType	numerator ≥ 0	optional	divisor, in media units per second, to be used when converting **count** to a time or time-like representation

The **count** attribute is an index of a media unit in a sequence. The first media unit corresponds to an initial index and does not need to be any particular value. The media units proceed as an ordinal sequence to termination, with each increment increasing in value by one.

The **rate** attribute in the TLXmediaCount item shall be constant within a sequence of media samples. This **rate** attribute need not be the same as the media sampling rate.

When generating TLX labels for a sequence of media units, the **count** value shall increment by one for each consecutive media unit; and the **rate** value, if present, shall be constant. Where TLX labels are sparse and/or TLXmediaCount is omitted for some TLX labels, the **count** value shall be generated as if all intervening labels had been generated, each with a TLXmediaCount.

NOTE: This ensures that TLXmediaCount represents an index for each media unit within the sequence for which the TLX labels were generated.

7.3. Time-like Values

A time-like value t, having the dimension of time, can be produced by dividing **count** by **rate**:

$$t = \mathbf{count} \,/ \mathbf{rate}$$

In the case where a time-like value representing a count of whole seconds w and whole media units r (in excess of those media units wholly contained in the seconds w) is being produced, the values for w and r should be produced by:

$$w = floor\left(\mathbf{count} \,/ ceiling(\mathbf{rate})\right)$$

$$r = \mathbf{count} \,\%\, ceiling(\mathbf{rate})$$

The properties of t, w and r are dependent on the conditions under which they are determined, as shown in Table 4.

Table 4 – Properties of t, w, and r

Conditions	t	w & r
fixed, integer media rate	time in s	time in s and media units
fixed, non-integer media rate	time in s	time-like non-SI seconds and media units
variable rate media	time-like	time-like

NOTE 1. In cases where **TLXmediaCount.rate** corresponds to an actual, fixed rate of the media, then t will be a time. If the media rate is further an integer media rate (e.g., 50/1), then when w and r are computed, the result is also a time. For other conditions, the results t or w and r merely resemble time and are said to be time-like. Such cases include **TLXmediaCount.rate** not corresponding to the actual media rate; use with variable rate media. Additional conditions that produce time-like values for w and r include non-integer media rates (e.g., 30000/1001) or any media rate less than one media unit per second.

NOTE 2. Where converting time-like seconds to a time of day, a zero value for the count attribute corresponds to a beginning of day.

NOTE 3. A time-like value of the form “HH:MM:SS:FF” can be derived by populating the “FF” field with digits representing r and the count of whole seconds w can be broken out as needed for the other fields of such a time-like value. The actual format for such a time-like value or string and the number of digits allocated to each of its fields is out of scope for this document.

For one example of producing time-like values using a count value of 86400 and a rate value of 24/1, results would be 3600.00 seconds for t, and 3600 seconds and 0 media units for w and r, respectively. With 3600 seconds in one hour, the 3600 seconds and 0 media units can be represented as 01:00:00:00.

For another example of producing time-like values using a count value of 86400 and a rate value of 24000/1001, results would be 3603.60 seconds for t, and 3600 seconds and 0 media units for w and r, respectively. Likewise, with 3600 seconds in one hour, the 3600 seconds and 0 media units can be represented as 01:00:00:00.

8 TLXmediaUnitInterval

8.1. Description

TLXmediaUnitInterval represents **an interval** in seconds between a media unit instant and the following media unit instant (if any). This is attribute is expressed as a rational, e.g., 1/25, 1001/30000 (i.e., numerator, denominator) to enable representation of fractional seconds.

8.2. Attributes

The attributes of TLXmediaUnitInterval shall be as specified in Table 5

Table 1 - TLXmediaUnitInterval attributes

Name	Type	Constraints	Required/Optional	Description
interval	rationalType		required	interval in seconds for this media unit

For a fixed-rate media sequence, **interval** is constant and the reciprocal of **interval** is the sample rate of the media.

For a variable-rate media sequence, **interval** need not be constant.

9 TLXuniqueSourceID

9.1. Description

TLXuniqueSourceID is an item that provides a **sourceID** for a media unit. The **sourceID** is a unique identifier created by or for a source of media labels.

9.2. Attributes

The attributes shall be as specified in Table 6.

Table 2 - TLXuniqueSourceID attributes

Name	Type	Constraints	Required/Optional	Description
sourceID	string	exactly 36 characters	required	string representation of UUID as specified in RFC 4122

The **sourceID** shall be a UUID, as specified in RFC 4122, and should be a version 1 (MAC address & time) or version 4 (random) UUID.

A given **sourceID** may be used by a source indefinitely, or a different **sourceID** may be used for each sequence or individual media unit of a sequence.

For a source that simultaneously labels multiple sequences of media units, the **sourceID** used for each simultaneously generated sequence shall be distinct and should use a different value in the clock sequence field, as specified in RFC 4122, whereby the **sourceID** for each such sequence can differ only by the clock sequence field.

NOTE: TLXuniqueSourceID might provide traceability to a specific physical device to support diagnostic investigations (e.g., tracking a camera with a dead pixel). Where **sourceID** is a version 1 UUID, traceability is directly enabled by the MAC address used to generate the UUID. Where **sourceID** is a version 4 UUID, determining what device produced the UUID is outside the scope of this standard.

10 TLXsourceName

10.1. Description

TLXsourceName is an item that provides an user-specified **name** for a source or sequence of media samples

10.2. Attributes

The attributes of TLXsourceName shall be as specified in Table 7.

Table 3 - TLXsourceName attributes

Name	Type	Constraints	Required/Optional	Description
name	string	up to 40 characters	required	

NOTE 1. TLXsourceName is available to support naming of sources or sequences from or in legacy workflows and is expected to find use in conjunction with the TLXst12 item. For example, **name** can represent a tape name, file name, camera name, etc. While **name** is associated with a sequence of media units, **name** can also be associated with other sequences, files, devices, or roles. Management of the uniqueness of **name** within a context (e.g., a production) or across different contexts, is outside the scope of this standard.

NOTE 2. A string, **name** can comprise Unicode characters whose UTF-8 representation requires more than one byte per character.

11 TLXst12

11.1. Description

TLXst12 is an item able to provide a representation of all the data in an SMPTE ST 12-1, ST 12-2, or ST 12-3 timecode plus, optionally, a count **modulus**. The attributes **timeAddress**, **binaryGroups**, and **st12Flags** are required for all ST 12 timecode representations, while the optional **DBB1** and **DBB2** attributes are only needed for use with ST 12-2.

11.2. Attributes

The attributes of TLXst12 shall be as specified in Table 8

Table 4 - TLXst12 attributes

Name	Type	Constraints	Required/Optional	Description

timeAddress	array of four integers, representing in order, hours (H), minutes (M), seconds (S), and frames (F).	0 ≤ H < 24, 0 ≤ M < 59, 0 ≤ S < 59, 0 ≤ F < 30	Required.	The elements of this value correspond to the elements of a time address as specified in ST 12-1, given a conversion between integers and binary-coded decimal (BCD) digits. When used for ST 12-3, frames (F) represents super-frames.
dropFrame	boolean		Optional. Default is false.	This corresponds to the drop frame flag specified in ST 12-1
binaryGroups	hexstringType	length of exactly 8 characters	Optional. Default is "00000000".	The first character corresponds to the eighth binary group as specified in ST 12-1, the second character to the seventh binary group, and so on, for all eight binary groups.
bgFlags	hexstringType	length of exactly 1; in the range "0" to "7"	Optional. Used with ST 12-1; not used with ST 12-3; default is "0".	A 3-bit value representing the three binary group flags BGF2, BGF1, BGF0 specified in ST 12-1, in big-endian order.
colorFrame	boolean		Optional. Used with ST 12-1; not used with ST 12-3; default is false.	This corresponds to the color frame flag specified in ST 12-1
fieldMark	boolean		Optional. Used with VITC in ST 12-1; not used with LTC in ST 12-1 nor in ST 12-3.	This corresponds to the field mark flag specified in ST 12-1
subFrame	hexstringType	string in the range "00" to "1f"	Optional. Used with ST 12-3; not used with ST 12-1.	A 5-bit value representing the five frame identifier bits sub-frame_1 through sub-frame_5, specified in ST 12-3, in big-endian order. Per ST 12-3, sub-frame_1 is the most significant bit.

modulus	string	Value is one of: "24", "25", "30", "48", "50", "60", "72", "96", "100", "120 (24x5)", "120 (30x4)"	Optional.	A value from "24" to "60" indicates an associated Frames-Per-Second System specified in ST 12-1. The remaining values indicate an associated Frame Counting specified in ST 12-3.
DBB1	hexstringType	length of exactly 2 characters	Optional. Used with ST 12-2 and ST 12-3.	Value represents the 8-bit binary value for DBB1 as specified in ST 12-2 and ST 12-3.
DBB2	hexstringType	length of exactly 2 characters	Optional. Used with ST 12-2 and ST 12-3.	Value represents the 8-bit binary value for DBB2 as specified in ST 12-2 and ST 12-3.

A **timeAddress** represents hours, minutes, seconds, and frames as integer values. In ST 12-1, a pair of binary coded decimal (BCD) digits, conveying units and tens, represent each of hours, minutes, seconds, and frames. Thus, a conversion between an integer value and a pair of BCD digits is necessary when transferring between a TLXst12 item and an ST 12 representation.

The character representation of **binaryGroups** is defined to read from left-to-right when examined in a hexstringType representation for these encodings of the ST 12-1 binary groups:

- auxiliary time address data, as described in RP 169 and ST 262
- date formatted as YYMMDD, as described in ST 309
- modified Julian date (MJD), as described in ST 309

Note: Because the LTC polarity correction bit specified in ST 12-1 is determined computationally, it is not carried by the TLX.

12 TLX Schema

A formal JSON schema for the derived types and TLX items specified in this document is provided as a convenient test material usable to assist validation of JSON documents as valid TLX labels. This schema is compliant with the December 8, 2020 IETF Internet Draft, JSON Schema. In cases where a normative clause of this document and the schema are found to be in conflict, the prose governs. For the purposes of this document, the symbol `$(ROOT)` is defined to be the root directory in which the test materials are located.

This schema is presented in `$(ROOT)/schemas/smpte-tlx-items-2021.json` and is self-identified by the item `"$id": "http://smpte-ra.org/schemas/2120-2/2021/smpte-tlx-items"`, the value of which is the IETF RFC 3986 Universal Resource Identifier (URI) that is this schema's canonical identification per IETF RFC 6596.

The .json files located in the directory `$(ROOT)/tests/smpte-tlx-items/` provide many examples of TLX labels, both valid and otherwise.

This schema is not intended to validate labels having virtual elements, as described in ST 2120-1 (i.e., where labels, items, or attributes have been omitted). Accordingly, it can be valuable for any label, items, or attributes to be completely inferred (i.e., reconstructed) prior to validation.

Note: The JSON Schema language does not detect duplicate names within an object, thus this schema leaves unverified the requirement from ST2120-1 that names in an object be unique.

Bibliography (Informative)

ISO 8601-1 Date and time – Representations for information interchange – Part 1: Basic rules

Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, DOI 10.17487/RFC3986, January 2005, <https://www.rfc-editor.org/info/rfc3986>.

Ohye, M. and J. Kupke, "The Canonical Link Relation", RFC 6596, DOI 10.17487/RFC6596, April 2012, <https://www.rfc-editor.org/info/rfc6596>.

Wright, A., H. Andrews; B. Hutton, G. Dennis, "JSON Schema: A Media Type for Describing JSON Documents", IETF, December 8, 2020 <https://datatracker.ietf.org/doc/html/draft-bhutton-json-schema-00>