INTERNATIONAL STANDARD

ISO/IEC 14496-30

First edition
2014-03-15

Information technology — Coding of audio-visual objects —

Part 30: Timed text and other visual overlays in ISO base media file format

Technologies de l'information — Codage des objets audiovisuels —

Partie 30: Texte temporisé et autres recouvrements visuels dans le format ISO de base pour les fichiers médias

Reference number
ISO/IEC 14496-30:2014(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 14496-30 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

ISO/IEC 14496 consists of the following parts, under the general title *Information technology — Coding of audio-visual objects*:

— *Part 1: Systems*

— *Part 2: Visual*

— *Part 3: Audio*

— *Part 4: Conformance testing*

— *Part 5: Reference software*

— *Part 6: Delivery Multimedia Integration Framework (DMIF)*

— *Part 7: Optimized reference software for coding of audio-visual objects* [Technical Report]

— *Part 8: Carriage of* ISO/IEC 14496 *contents over IP networks*

— *Part 9: Reference hardware description* [Technical Report]

— *Part 10: Advanced Video Coding*

— *Part 11: Scene description and application engine*

— *Part 12: ISO base media file format*

— *Part 13: Intellectual Property Management and Protection (IPMP) extensions*

— *Part 14: MP4 file format*

— *Part 15: Advanced Video Coding (AVC) file format*

— *Part 16: Animation Framework eXtension (AFX)*

— *Part 17: Streaming text format*

— *Part 18: Font compression and streaming*

— *Part 19: Synthesized texture stream*

— *Part 20: Lightweight Application Scene Representation (LASeR) and Simple Aggregation Format (SAF)*

— *Part 21: MPEG-J Graphics Framework eXtensions (GFX)*

— *Part 22: Open Font Format*

— *Part 23: Symbolic Music Representation*

— *Part 24: Audio and systems interaction* [Technical Report]

— *Part 25: 3D Graphics Compression Model*

— *Part 26: Audio conformance*

— *Part 27: 3D Graphics conformance*

— *Part 28: Composite font representation*

— *Part 29: Web video coding*

— *Part 30: Timed text and other visual overlays in ISO base media file format*

Introduction

This part of ISO/IEC 14496 defines a storage format based on, and compatible with, the ISO Base Media File Format (ISO/IEC 14496-12 and ISO/IEC 15444-12), which is used by the MP4 file format (ISO/IEC 14496-14) and the Motion JPEG 2000 file format (ISO/IEC 15444-3) among others. This part of ISO/IEC 14496 enables timed text and subtitle streams to

— be used in conjunction with other media streams, such as audio or video,

— be used in an MPEG-4 systems environment, if desired,

— be formatted for delivery by a streaming server, using hint tracks, and

— inherit all the use cases and features of the ISO Base Media File Format on which MP4 and MJ2 are based.

Information technology — Coding of audio-visual objects —

Part 30:
Timed text and other visual overlays in ISO base media file format

1 Scope

This part of ISO/IEC 14496 describes the carriage of some forms of timed text and subtitle streams in files based on the ISO base media file format (ISO/IEC 14496-12). The documentation of these forms does not preclude other definition of carriage of timed text or subtitles; see, for example, 3GPP Timed Text (3GPP TS 26.245).

2 Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

W3C Recommendation, *Timed Text Markup Language 1.0, Second Edition*[1)]

ISO/IEC 14496-12, *Information technology — Coding of audio-visual objects — Part 12: ISO base media file format*[2)]

W3C Community Group Report, *WebVTT:The Web Video Text Tracks Format*[3)]

3GPP TS 26.245, *Transparent end-to-end Packet switched Streaming Service (PSS); Timed text format*

IETF RFC 3986, *Uniform Resource Identifier (URI): Generic Syntax*

3 Terms and definitions

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

3.1
timed text document
file-based representation of textual content, possibly XML, used to produce timed text streams and possibly representing timed text track samples

3.2
timed text stream
stream of content, which when decoded results in textual content, possibly containing internal timing values, to be presented at a given presentation time and for a certain duration

3.3
subtitle stream
timed text stream potentially also presenting images

1) http://www.w3.org/TR/ttaf1-dfxp/

2) ISO/IEC 14496-12 is technically identical to ISO/IEC 15444-12.

3) http://www.w3.org/2013/07/webvtt.html

3.4
internal timing value
value contained in the payload of a timed text stream sample representing a time, e.g. a start time, an end time, or a duration, corresponding to a timed behaviour of a part or the whole of the sample

3.5
timed text track
ISOBMFF representation of a timed text stream

3.6
subtitle track
ISOBMFF representation of a subtitle stream

4 Abbreviated terms

For the purposes of this International Standard, the following abbreviated terms apply.

TTML Timed Text Markup Language

WebVTT Web Video Text Tracks

ISOBMFF ISO Base Media File Format

5 General Definitions

5.1 Layout

This subclause defines common layout behavior for processing of timed text or subtitle samples.

Unless specified by an embedding environment (e.g. an HTML page), the track header box information (i.e. width, height) shall be used to size the subtitle or timed text track content with respect to the video; otherwise, it may be ignored by the embedding environment. The width and height of the subtitle or timed text track should be appropriate for the width and height of the video track (as declared in the track header) it is intended to overlay, even if the video is not stored in an ISOBMFF file or stored as a track in a different ISOBMFF file. A typical usage is that the timed text or subtitle track has the same width and height as the underlying video, and no translation. For some timed text documents, the region thus defined corresponds to the visual area filled by the rendering of the timed text documents.

Additional region positioning using the translation values tx and ty from the track header matrix, as defined for 3GPP Timed Text tracks, may be used (see 3GPP TS 26.245, section 5.7, for the definition of the text track region using tx, ty, and the track width and height).

NOTE The 3GPP region is not the same as a WebVTT region.

Unless specified by an embedding environment (e.g. an HTML page), visually composed tracks including video, subtitle, and timed text shall be stacked or layered using the 'layer' value in the track header box. The layer field provides the same functionality as z-index in TTML.

NOTE Timed text and subtitle tracks are normally stacked in front of the video.

5.2 Timing

This subclause defines common timing behavior for processing of timed text or subtitle samples.

The general processing of timed text or subtitle tracks is that the text content of the sample is delivered to the decoder at the sample decode time, at the latest. The rendering of the sample happens at the composition time, taking into account edit lists if any, and for the whole sample duration, without timing behavior. However, timed text or subtitle sample data of specific formats may contain internal timing

values. Internal timing values may alter the rendering of the sample during its duration as specified by the timed text or subtitle format.

NOTE If an internal timing value does not fall in the time interval corresponding to the sample composition time and sample composition time plus sample duration, the rendering of the sample may be different from the rendering of the same sample data with a composition time such that the internal timing value lies in the associated composition interval.

The subclauses defining the storage of specific formats in the ISOBMFF specify how internal timing values relate to the track time or to the sample decode or composition time (see 6.3 and 7.3). For instance, start or end times may be relative to the start of the sample, or the start of the track.

For sections of the track timeline that have no associated subtitles or timed text content, 'empty' samples may be used, as defined for each format, or the duration of the preceding sample extended. Samples with a size of zero are not used.

The timescale field in the media header box should be set appropriately to achieve the desired timing accuracy; it is recommended to be set to the value of the timescale field in the corresponding video track's media header box.

5.3 Language

Timed text tracks should be marked with a suitable language in the media header box, indicating the audience for whom the track is appropriate. In the case where it is suitable for a single language, the media header must match that declared language. The value 'mul' may be used for a multi-lingual text.

5.4 Resources shared by multiple samples

Common resources, such as images and fonts, that are referred to by URLs, may be stored as items in a MetaBox as defined by ISO/IEC 14496-12. These items may be addressed by using the item_name as a relative URL in the timed text sample, as defined by 8.11.9 of ISO/IEC 14496-12.

NOTE A derived specification, with its applicable brand, may restrict this use of meta boxes for common items.

Fonts not supplied with the content may be already present on the target system(s), or supplied using any suitable supported mechanism (e.g. font streaming as defined in ISO/IEC 14496-18[1]).

6 Timed Text Markup Language (TTML)

6.1 Introduction

This subclause describes how documents based on TTML, as defined by the W3C, and derived specifications (for example SMPTE-TT[2]), are carried in files based on the ISO base media file format.

6.2 Layout

Subclause 5.1 defines the general layout behaviour for timed text and subtitle tracks. In particular, this means for TTML tracks that the track width and height provide the spatial extent of the root container, as defined in the TTML Recommendation. Any 'extent' attribute declared on the 'tt' element in the contained TTML document shall match the track width and height.

6.3 Timing

The top-level internal timing values in the timed text samples based on TTML express times on the track presentation timeline – that is, the track media time as optionally modified by the edit list. For example, the begin and end attributes of the <body> element, if used are relative to the start of the track, not relative to the start of the sample. This is shown in the figure below, using W3C TTML syntax.

In Figure 1, the sample composition time of each samples are 0, 30 minutes, and 1 hour, which correspond to the time at which the decoder will present the TTML content. The first sample, as per the TTML Recommendation, will not display any content in the first minute or after 2 minutes, and again, per TTML, will remain as such until the next sample is processed. The second sample contains a document describing the rendering between composition time 0 and 32 minutes. However, since it is provided to the decoder after 30 minutes and since internal timing values are relative to the start of the track, the TTML decoder will display the text as if it seeked to the 30 minutes into the document. It will not render anything for the first minute from the beginning of the sample, and then render some text for 1 other minute, and then again no rendering until the next sample is processed. The processing of sample 3 is similar, where the top level internal timings on the div elements are handled as relative to the start of the track.

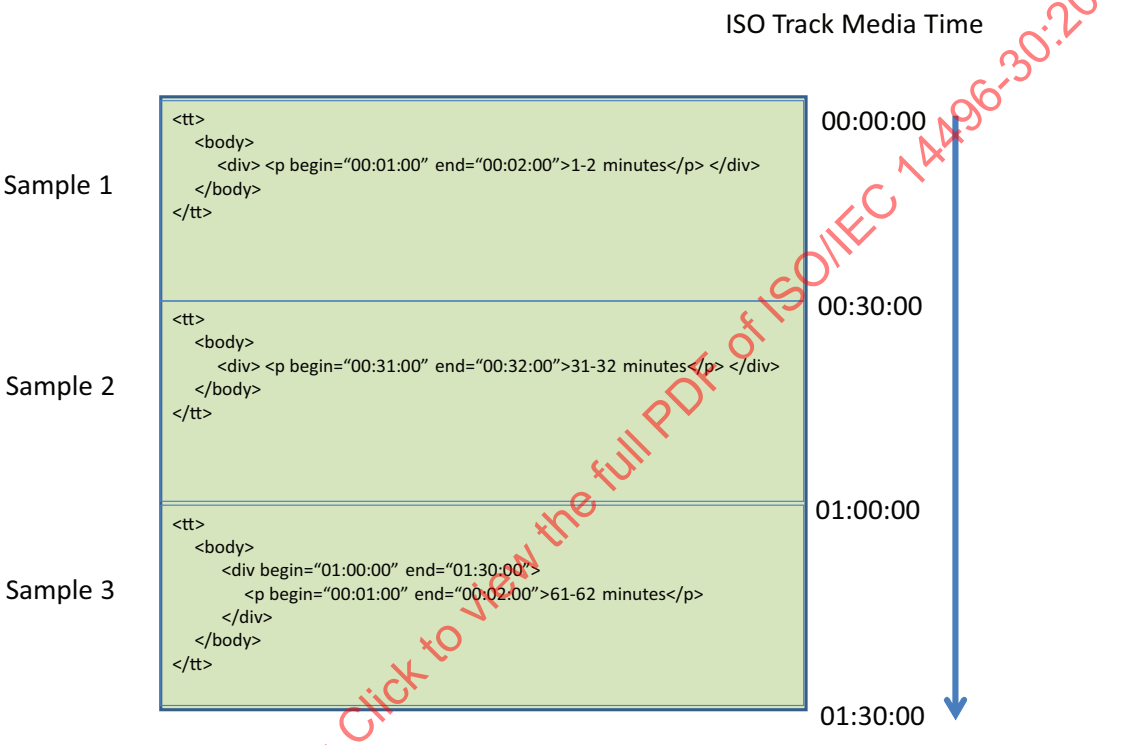

Figure 1 — Example of a TTML track with three samples

No transport layer buffer or timing model is defined to guarantee that subtitle content can be read and processed in time to be synchronously presented with audio and video. It is assumed that users of this track format will define timed text content profiles and hypothetical render models that will constrain content parameters so that compatible decoders may identify and decode those profiles for synchronous presentation.

The following document constraints may need to be specified to define a timed text profile that will guarantee synchronous decoding of conforming content on conforming decoders:

— Maximum allowed document size

— Number of document buffers in the hypothetical render model

— Video overlay timing of the hypothetical render model

— Maximum total compressed image size in megabytes per sample

— Maximum total decoded image size in megapixels per sample

— Maximum decoded image dimensions

— Maximum text rendering rate required by a document

— Maximum image rendering rate required by a document

— Maximum number of simultaneously displayed characters

— Maximum font size

— Maximum number of simultaneously displayed images

NOTE Defining timed text content profiles is outside the scope of this part of the standard, but providing a method to signal an externally defined timed text profile in the subtitle sample description is possible using the sample entry description.

An 'empty' sample is defined as containing a TTML document that has no content.

The duration of the TTML document carried in a sample may be less than the sample duration, but should not be greater.

6.4 Track format

TTML streams shall be carried in subtitle tracks, and as a consequence according to ISOBMFF, the media handler type is 'subt', and the track uses a subtitle media header, and associated sample entry and sample group base class.

6.5 Sample entry format

TTML streams shall use the XMLSubtitleSampleEntry format.

The namespace field shall be set to at least one unique namespace. It should be set to indicate the primary TTML-based namespace of the document, and should be set to all namespaces in use in the document (e.g. TTML + TTML-Styling + SMPTE-TT),

The schema_location field should be set to schema pathnames that uniquely identify the profile or constraint set of the namespaces included in the namespace field.

When sub-samples are present (see 6.6), then the auxiliary_mime_types field shall be set to the mime types used in the sub-samples – e.g. "image/png".

6.6 Sample format

A TTML subtitle sample shall consist of an XML document, optionally with resources such as images referenced by the XML document. Every sample is therefore a sync sample in this format; hence, the sync sample table is not present.

Other resources such as images are optional. Resources referenced by an XML document may be stored in the same subtitle sample as the document that references them, in which case they shall be stored contiguously following the XML document that references those resources. Resources should be stored in presentation time order.

When resources are stored in a sample, the Track Fragment Box ('traf') shall contain a Sub-Sample Information Box ('subs') constrained as follows:

`entry_count` and `sample_delta` shall be set to 1 since each subtitle track fragment contains a single subtitle sample.

`subsample_count` shall be set to the number of resources plus 1.

`subsample_priority` and `discardable` have no meaning; they shall be set to zero on encoding and may be ignored by decoders.

If sub-samples are used, the XML document shall be the first sub-sample entry. Each resource the document references shall be defined as a subsequent sub-sample in the same table.

The XML document shall reference each sub-sample object using a URI, as per RFC3986. When a URN is used, it shall be of the form:

urn:<nid>:.....:<index>[.<ext>]

Where:

<nid> is the registered URN namespace ID per RFC 2141.

<index> is the sub-sample index "j" in the 'subs' referring to the object in question.

<ext> is an optional file extension - e.g. "png".

The first resource in the sample will have a sub-sample index value of 1 in the 'subs' and that will be the index used to form the URI.

Reference the same object can be made multiple times within an XML document. In such cases, there will be only one sub-sample entry in the Sub-Sample Information Box for that object, and the URN's used to reference the object each time will be identical.

An example construction of the sample with images is shown in Figure 2.

NOTE The text in the images is just an example and not meant to constrain or imply anything about what is encoded in the images.

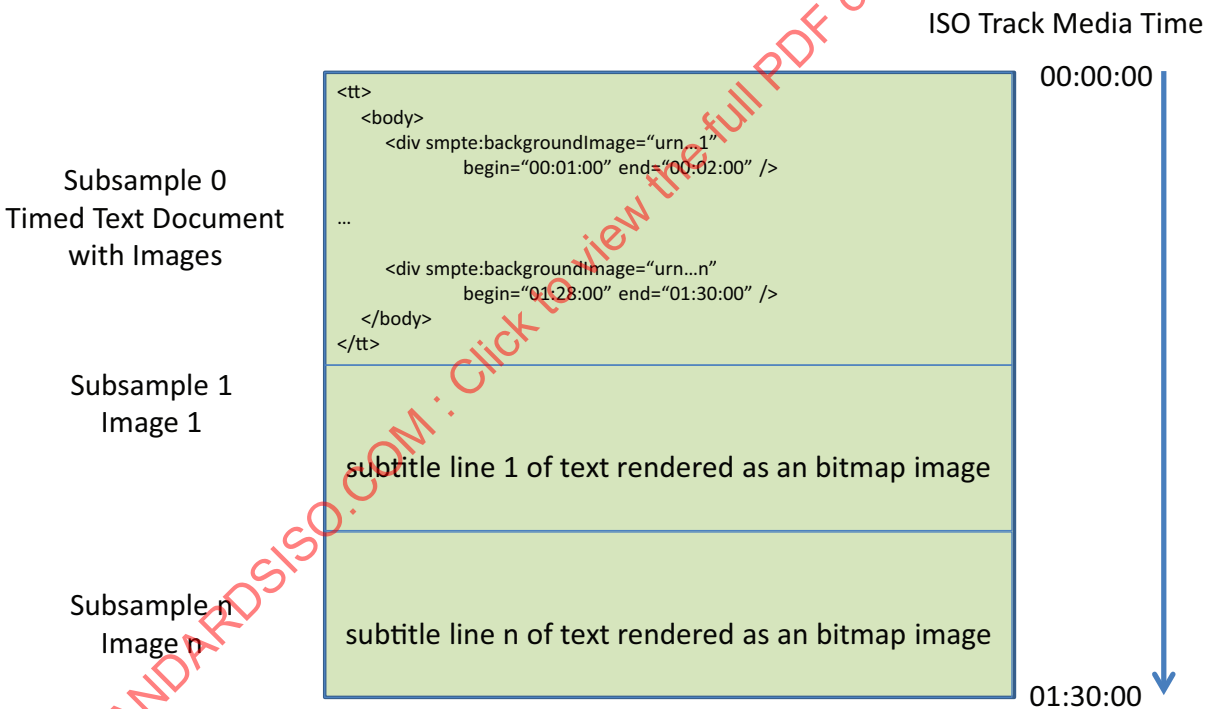

Figure 2 — Subtitle sample structure when using subsamples

6.7 Additional Considerations

The following additional considerations should be addressed by users of this carriage:

— Unicode character codes allowed

— Fonts and styles allowed

7 Web Video Text Tracks (WebVTT)

7.1 Introduction

This section defines how documents based on the WebVTT specification are carried in files based on the ISO base media file format.

WebVTT text content in tracks is encoded using UTF-8, and the data-type `boxstring` indicates an array of UTF-8 bytes, to fill the enclosing box, with neither a leading character count nor a trailing null terminator.

Each WebVTT cue, as defined in the WebVTT specification, is stored de-constructed, partly to emphasize that the textual timestamps one would normally find in a WebVTT file do *not* determine presentation timing; the ISO file structures do. It also separates the text of the actual cue from the structural information that the WebVTT file carries (positioning, timing, and so on). WebVTT cues are stored in a typical ISO boxed structured approach to enable interfacing an ISO file reader with a WebVTT renderer without the need to serialize the sample content as WebVTT text and to parse it again.

Boxes shall not contain trailing CR or LF characters, or trailing CRLF sequences (where 'trailing' means that they occur last in the payload of the box).

7.2 Layout

Subclause 5.1 defines the general layout behaviour for timed text and subtitle tracks, which is applicable to WebVTT tracks.

7.3 Timing

Following the general timing processing defined in 5.2, each cue shall be passed to the WebVTT renderer at the time from the time-to-sample table, as mapped by the edit list (if any). The times derived for a sample from the durations in the time-to-sample table reflect the start and end-time of all cues in that sample. All samples are sync samples; the sync sample table is not used.

If there is internal timing value in a cue, each sample must be labelled with the VTT time that corresponds to the sample start time on the VTT time line..

NOTE 1 This enables reconstructing a correct internal timing value, when the time-to-sample table is edited.

NOTE 2 Internal timestamps within the cue that precede this current time would be already ":past" at the start of the sample, for example.

7.4 Track format

WebVTT streams shall be carried as timed text tracks, and as a consequence according to ISOBMFF, use the `'text'` media handler type, and the associated media header, sample entry, and sample group base class.

7.5 Sample entry format

WebVTT streams shall use the WVTTSampleEntry format.

In the sample entry, a WebVTT configuration box must occur, carrying exactly the lines of the WebVTT file header, i.e. all text lines up to but excluding the 'two or more line terminators' that end the header.

NOTE Other boxes may be defined for the sample entry in future revisions of this specification (e.g. carrying optional CSS style sheets, font information, and so on).

A WebVTT source label box should occur in the sample entry. It contains a suitable string identifier of the 'source' of this WebVTT content, such that if a file is made by editing together two pieces of content, the timed text track would need two sample entries because this source label differs. A URI is recommended

for the source label; however, the URI is not interpreted and it is not required there be a resource at the indicated location when a URL form is used.

```
class WebVTTConfigurationBox extends Box('vttC') {
   boxstring   config;
}
class WebVTTSourceLabelBox extends Box('vlab') {
   boxstring   source_label;
}
class WVTTSampleEntry() extends PlainTextSampleEntry ('wvtt'){
   WebVTTConfigurationBox   config;
   WebVTTSourceLabelBox     label;    // recommended
   MPEG4BitRateBox ();                // optional
}
```

7.6 Sample format

The character replacements as specified in step 1 of the WebVTT parsing algorithm may be applied before VTT data is stored in this format. Readers should be prepared to apply these replacements if integrated directly with a WebVTT renderer.

Each sample is either:

a) exactly one VTTEmptyCueBox box (representing a period of non-zero duration in which there is no cue data) or

b) one or more VTTCueBox boxes that share the same start time and end time, each containing the following boxes. Only the CuePayloadBox is mandatory, all others are optional. A sample containing cue boxes may also contain zero or more VTTAdditionalTextBox boxes, interleaved between VTTCueBox boxes and carrying any other text in between cues, in the order required by the processing of the additional text, if any.

The VTTCueBox boxes must be in presentation order, i.e. if imported from a WebVTT file, the cues in any given sample must be in the order they were in the WebVTT file.

It is recommended that the contents of the VTTCueBox boxes occur in the order shown in the syntax, but the order is not mandatory.

If a cue has WebVTT Cue Settings, they are placed into a CueSettingsBox without the leading space that separates timing and settings.

When a WebVTT source label box is present in the sample entry and a cue is written into multiple samples, it must be represented in a set of VTTCueBoxes all containing the same source_ID. All VTTCueBoxes that originate from the same VTT cue must have the same source_ID, and that source_ID must be unique within the set of cues that share the same source_label. This means that when stepping from one sample to another (possibly after a seek, as well as during sequential play), a match of source_ID under the same source_label is diagnostic that the same cue is still active. Cues with no CueSourceIDBox are independent from all other cues; a source ID may be assigned to all cues.

When there is no WebVTT source label in the sample entry, there must be no CueSourceIDBox in the associated samples. In this way the presence of the WebVTT source label indicates whether source IDs are assigned to cues split over several samples, or not.

When a cue has internal timing values (i.e. WebVTT cue timestamp as defined in the WebVTT specification) then each VTTCueBox must contain a CueTimeBox which gives the VTT timestamp associated with the start time of sample. When the cue content of a sample is passed to a VTT renderer, timestamps within the cues in the sample must be interpreted relative to the time given in this box, or adjusted considering this time and the sample start time.

The CuePayloadBox must contain exactly one WebVTT Cue. Other text, such as WebVTT Comments are placed into VTTAdditionalText boxes.

NOTE The sample entry code is 'vttC'; in contrast the VTTCueBox is 'vttc' and their container is also different.

In the CuePayloadBox there must be no blank lines (but there may be multiple lines).

```
aligned(8) class VTTCueBox extends Box('vttc') {
   CueSourceIDBox()  // optional source ID
   CueIDBox();       // optional
   CueTimeBox();     // optional current time indication
   CueSettingsBox(); // optional, cue settings
   CuePayloadBox();  // the (mandatory) cue payload lines
};
class CueSourceIDBox extends Box('vsid') {
   int(32)   source_ID;    // when absent, takes a special 'always unique' value
}
class CueTimeBox extends Box('ctim') {
   boxstring   cue_current_time;
}
class CueIDBox extends Box('iden') {
   boxstring   cue_id;
}
class CueSettingsBox extends Box('sttg') {
   boxstring   settings;
}
class CuePayloadBox extends Box('payl') {
   boxstring   cue_text;
}
// These next two are peers to the VTTCueBox
aligned(8) class VTTEmptyCueBox extends Box('vtte') {
   // currently no defined contents, box must be empty
};
class VTTAdditionalTextBox extends Box('vtta') {
   boxstring   cue_additional_text;
}
```

Free space boxes and unrecognized boxes in any sample, or within the VTTCueBox or VTTEmptyCueBox may be present and should be ignored.

7.7 Converting to or from a WebVTT text file (Informative)

7.7.1 Introduction

This subclause connects the box structure to the parsing process for a WebVTT file as defined in section 5 of the WebVTT specification. Underlined terms here correspond to defined terms in that specification.

7.7.2 Importing a WebVTT file into the ISO base media file format

Prior to import, the character replacements as specified in step 1 of the WebVTT parsing algorithm, may be applied.

The initial part of the file, from the first characters (the string 'WEBVTT'), up to but not including the 'two or more line terminators' are placed into the WebVTTConfigurationBox.

The WebVTT Cue Timings of each WebVTT Cue are processed to form a set of samples that are contiguous and non-overlapping in time as follows:

a) The start time offset of the cue sets the sample decode time. The end time offset of the cue is used to set the sample duration.

b) If the start time offset of the cue is later than the decoding time plus the decoding duration of the previous sample, then a VTTEmptyCueBox is used to fill the empty time;

c) If the start time offset of the cue is earlier than the decoding time plus the decoding duration of the previous sample, then the cues overlap, and the earlier cue is split into a set of equivalent cues, such that for every start time offset and end time offset, there is a sample start or end time.

d) A source_ID is generated for each cue, and placed into the samples in at least all the cues that are represented by more than one VTTCueBox.

e) Each WebVTT Comment is placed in a VTTAdditionalTextBox. This should be placed either (1) in the sample containing the first occurrence of the following WebVTT Cue, before that cue or (2) in the last sample associated with a given source label, after the last cue.

Each split cue is then decomposed:

a) The WebVTT Cue Identifier, if it exists, is placed in a CueIDBox;

b) The WebVTT Cue Timings are removed, as they are now represented in the time to sample mapping of the file;

c) If the cue contains WebVTT cue timestamp, then a CueTimeBox is placed into the VTTCueBox;

d) The WebVTT Cue Settings, if they exist, are placed into a CueSettingsBox;

e) The Cue Payload, without the following two or more line terminators, is placed into the CuePayload box.

7.7.3 Exporting a WebVTT file from the ISO base media file format

To form a WebVTT file from the track, the file is started with the contents of the WebVTTConfigurationBox, followed by two line terminators.

The cue boxes of each sample are processed separately. Each cue box is marked with its start and end time, derived from the sample times (including the edit list).

The 'continuation cues' (adjacent samples that use the same sample entry and containing cues that have the same source_ID) may be merged with their preceding matching cue or continuation cue, and the end time offset set to the end time of the later continuation cue.

The internal timing values of each cue are adjusted to be relative to the start time offset.

Each cue is then written to the file: the optional WebVTT Cue Identifier from the CueIDBox, if it exists, followed by a line terminator; the WebVTT Timings, computed as above; the WebVTT Cue Settings from the CueSettings box, if it exists; and the Cue Payload from the CuePayload box. Each WebVTT Cue so formed are separated from the next by two or more line terminators. VTTEmptyCueBoxes are discarded.

7.8 Example (Informative)

7.8.1 Source File

```
WEBVTT

1
00:11.000 --> 00:12.500 align:start line:10
<v Roger Bingham>We are in New York City.
We are looking straight down 5th Avenue.

00:13.000 --> 00:18.000
<v Neil DeGrass Tyson>Didn't you already say that?

2
00:17.000 --> 00:20.000
Testing... <00:17.350>One... <00:18.125>Two...
```

7.8.2 Imported Format

For the example above, the cues might be represented by the following samples:

Sample 0, duration 11.000

```
[emptycue]
```


