

EBU – TECH 3350



EBU-TT
Part 1
Subtitling format definition

Status: Version 0.9 for comment

Geneva
January 2012

Conformance Notation

This document contains both normative text and informative text.

All text is normative except for that in the Introduction, Examples, any section explicitly labelled as 'Informative' or individual paragraphs which start with 'Note:'.

Normative text describes indispensable or mandatory elements. It contains the conformance keywords 'shall', 'should' or 'may', defined as follows:

- 'Shall' and 'shall not': Indicate requirements to be followed strictly and from which no deviation is permitted in order to conform to the document.
- 'Should' and 'should not': Indicate that, among several possibilities, one is recommended as particularly suitable, without mentioning or excluding others.
OR indicate that a certain course of action is preferred but not necessarily required.
OR indicate that (in the negative form) a certain possibility or course of action is deprecated but not prohibited.
- 'May' and 'need not': Indicate a course of action permissible within the limits of the document.

Default identifies mandatory (in phrases containing "shall") or recommended (in phrases containing "should") presets that can, optionally, be overwritten by user action or supplemented with other options in advanced applications. Mandatory defaults must be supported. The support of recommended defaults is preferred, but not necessarily required.

Informative text is potentially helpful to the user, but it is not indispensable and it does not affect the normative text. Informative text does not contain any conformance keywords.

A conformant implementation is one which 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.

Contents

Status of this document (Informative).....	7
Scope (Informative)	9
Definitions of terms	9
1. Introduction.....	10
1.1 Background	10
1.2 EBU-TT as exchange format	10
1.3 Relation to other specifications.....	10
2. Generic constraints.....	11
2.1 Namespaces	11
2.2 Extensibility	12
2.3 Initial values.....	12
2.4 Order of timecode values	12
2.5 Compatibility with TTML timing model	13
2.6 Unicode support	13
2.7 White space handling	13
3. Document Structure.....	13
3.1 Head	17
3.1.1 Metadata in tt:head.....	17
3.1.2 Styling in tt:head	24
3.1.3 Layout in tt:head	29
3.2 Body	31
3.2.1 Div	32
4. Datatypes	36
4.1 ebuttdt:cellResolutionType.....	36
4.2 ebuttdt:colorType.....	36
4.3 ebuttdt:extentType	37
4.4 ebuttdt:fontFamilyType.....	37
4.5 ebuttdt:fontSizeType	37
4.6 ebuttdt:framerateMultiplierType	38
4.7 ebuttdt:lengthType	38
4.8 ebuttdt:lineHeightType	38
4.9 ebuttdt:originType	38
4.10 ebuttdt:paddingType	38
4.11 ebuttdt:timingType	38

5. Bibliography	39
Annex A: Use of ebutts:multiRowAlign	40
Annex B: Time container with no specified duration	44
Annex C: Support of the STL GSI Block in EBU-TT	46
Annex D: Initial Values of TTML and EBU-TT attributes	48
Annex E: Index of elements, attributes and types	50

Status of this document (Informative)

This document is a working draft of the EBU-Timed Text (EBU-TT) format definition. The EBU publishes the 0.9 version of EBU-TT to indicate that the specification is believed to be stable and to encourage implementation by the developer community.

Interested parties are invited to review the document and send comments to subtitling@ebu.ch up until 31 March 2012.

This is a draft document and may be updated, replaced or made obsolete by other documents at any time. It is inappropriate to cite this document as anything other than work in progress

This document is the first part of a series of EBU-TT documents. The full list of planned EBU-TT documents is given below.

Part 1: EBU-TT subtitling format definition

Introduction to EBU-TT and definition of the XML based format.

Part 2: Archive (STL)

How EBU-TT provides backwards compatibility with EBU STL.

Part 3: Live subtitling

How to use EBU-TT for the production of live subtitles.

Part 4: Annotation

How EBU-TT can be used in future scenarios for 'authoring of intent'.

EBU-TT-4 is the more abstract form of a subtitle archive file and a single EBU-TT-4 file could be converted to a number of different EBU-TT-Part 1 files, each targeted at a specific use case.

Further, an EBU-TT-1 file might be used as a foundation to create an EBU-TT-4 file by the addition of extra metadata and the re-casting of the EBU-TT-Part 1 specific attribute data into a more abstract representation.

Part 5: User Guide

General guide ('How to use EBU-TT').

XML Schema (Informative)

An example of an XML Schema (ebu-tt-v0.9-xml-schema.zip) is available for free download from the following web page - <http://tech.ebu.ch/ebu-tt>.

Editor's note: Exceptionally, for consistency throughout this document the American English spellings 'color' and 'center' have been used.

EBU-TT Part 1 Subtitling Format Definition

<i>EBU Committee</i>	<i>First Issued</i>	<i>Revised</i>	<i>Re-issued</i>
TC	2012		

Keywords: subtitling, STL, XML, W3C, TTML, DFXP, captions

Scope (Informative)

Subtitles are created, edited, exchanged and archived in many different ways. At one extreme subtitles may be closely linked to the video, e.g. as burned-in (so-called open, or forced) subtitles in the video, at the other extreme they may be loosely coupled to the audio/video essence (e.g. stored on an external storage medium and associated with the video at the moment of playout/viewing).

The aim of this publication is to specify an XML based archiving and interchange format for subtitles as follow-up to the currently widely used EBU STL format (EBU Tech 3264) [1]. The EBU has developed a recommendation on transport of subtitles inside and outside MXF files (R 133) [2] and has been investigating implementations that are in use today. Harmonization is needed to obtain predictable and reliable results when interchanging subtitle files and when integrating new products into TV production environments.

Definitions of terms

Captions and subtitles

The term “captions” describes on screen text for use by deaf and hard of hearing audiences. Captions include indications of the speakers and relevant sound effects.

The term “subtitles” describes on screen text for translation purposes.

For easier reading only the term “subtitles” is used in this specification as the EBU-TT file representation for captions or subtitles is identical.

In this specification the term “captions” may be used interchangeably for the term “subtitles” (except where noted).

Active picture

The term “active picture” (known alias: Production Aperture) refers to the full image size, as specified in the relevant Standard concerned, regardless of any safe areas.

1. Introduction

1.1 Background

The introduction of higher resolution television formats (HDTV), user demands for improved presentation, the switch to file-based production workflows and the multiplication of web-based distribution mechanisms, require a new XML based subtitling format that can retain its timing characteristics during the creation and transport of subtitles.

The introduction of HDTV has created new expectations, including displaying subtitles in different and more user-friendly ways. This recommendation aims to support these new requirements.

To enhance the quality of the subtitler's work more efficient automation processes are needed, allowing a subtitler to add value to the subtitles by using his contextual knowledge, cultural awareness and special skills.

Reliable exchange mechanisms are especially important for the creation of subtitles, which often takes place in external production houses or at home using a wide variety of different platforms and applications.

1.2 EBU-TT as exchange format

EBU-TT is intended as general purpose exchange format for subtitles and supports Unicode characters. As an exchange format EBU-TT intrinsically also is an archiving format (see Figure 1).

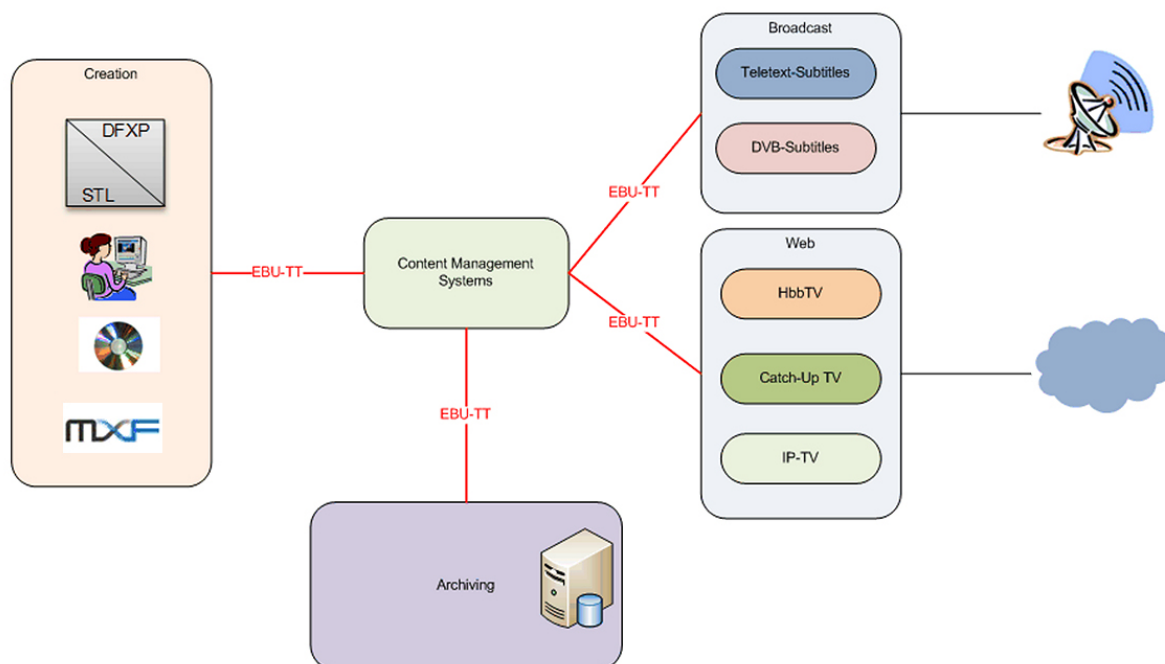


Figure 1: Subtitle workflow with EBU-TT

1.3 Relation to other specifications

EBU-TT uses a subset of the vocabulary given by the W3C Timed Text Markup Language (TTML) 1.0 [3]. This W3C standard was formerly known as "Timed Text (TT) Authoring Format 1.0 - Distribution Format Exchange Profile (DFXP)".

The EBU-TT format is intended to constrain the features provided by TTML, especially to make EBU-TT more suitable for the use with broadcast video and web video applications.

Valid EBU-TT documents are, by definition of the W3C TTML Specification, valid W3C TTML documents. Note however that it is possible to construct valid W3C TTML documents that are not valid EBU-TT documents.

Like the EBU-TT specification, the SMPTE Standard ST 2052-1:2010 (SMPTE-TT) [4] is derived from TTML. Valid EBU-TT documents are, by the definition of the SMPTE-Standard, valid SMPTE-TT documents but it is possible to construct valid SMPTE-TT documents that are not valid EBU-TT documents.

2. Generic constraints

The EBU-TT format defines constraints for an XML document instance. A valid EBU-TT XML document has to comply with the generic constraints in § 2 and the document structure defined in § 3.

Note: To facilitate the implementation of EBU-TT, a W3C XML Schema (in the following XML Schema) will be published together with the EBU-TT specification.

Although the XML Schema reflects the generic constraints and the document structure detailed in this spec, it is not normative but informative. This is not an error in design but a result of the limitation of XML Schema 1.0 in the expression of XML document constraints. Because of this limitation the EBU-TT XML Schema can produce 'false positive validations' which means that an XML document can validate against the EBU-TT XML Schema although it does not comply with a certain constraint in the EBU-TT specification.

One example of a constraint that cannot be expressed in an XML Schema is the dependency between a value that uses a pixel-based length expression and the requirement of the extent attribute on the `tt:tt` root element (see `ttp:cellResolution` (attribute) in § 3).

2.1 Namespaces

The following external namespaces from the W3C TTML specification shall be used for the TTML elements and attributes in EBU-TT:

Name	Prefix	Value
TT	tt:	http://www.w3.org/ns/ttml
TT Parameter	ttp:	http://www.w3.org/ns/ttml#parameter
TT Style	tts:	http://www.w3.org/ns/ttml#styling

The following namespaces shall be used for the assignment of XML Schema datatypes:

Name	Prefix	Value
XML Schema	xs:	http://www.w3.org/2001/XMLSchema

The following namespaces shall be used for the EBU-TT specific vocabulary:

Name	Prefix	Value
EBU-TT Metadata	ebuttm:	urn:ebu:tt:metadata
EBU-TT Styling	ebutts:	urn:ebu:tt:style
EBU-TT Datatypes	ebuttdt:	urn:ebu:tt:datatypes

2.2 Extensibility

Any Element defined in § 3 “Document Structure” may contain one `tt:metadata` child element. Exceptions are:

- the `tt:tt` element (root element)
- the `tt:metadata` element itself
- all elements in the EBU-TT Metadata namespace

If an element has a `tt:metadata` as child element `tt:metadata` shall appear before all other child elements that are defined for this element by EBU-TT (see § 3 “Document Structure”).

Every `tt:metadata` element may have one or more elements as children. With the exception of the elements included in this specification, these elements shall not be in a namespace defined by the W3C TTML or EBU-TT specification.

2.3 Initial values

TTML defines initial values for certain attributes that act as fallback values in case a value cannot be computed from a specified value in the document.

Note: One example is the attribute `tts:textAlignment`. If an EBU-TT document does not specify the text-alignment for a subtitle text then the initial value shall be “left”.

For most of the attributes in EBU-TT, the initial values from TTML are adopted. In case EBU-TT overwrites the initial value of a certain attribute this is highlighted in the description of the corresponding attribute in § 3 “Document Structure”.

For completion all initial values that are used by EBU-TT are listed in Annex D “Initial Values of TTML and EBU-TT attributes”.

Note: To clarify the intention of the author of an EBU-TT document it is recommended that attributes and their values be explicitly specified rather than relying on their initial values.

2.4 Order of timecode values

The order in which the timecode values of the `begin` and `end` attributes of the `tt:p` and `tt:span` elements appear in an EBU-TT document should be linear and monotonically increasing (see definition of `ttp:markerMode` attribute in § 3 “Document Structure”).

Note: Timecode in SMPTE 12M-1-2008 [5] can only represent a maximum of a single 24 hour period. In certain circumstances, for example when recording ‘live’ subtitle input, it is anticipated that timecode values may cross the 24 hour ‘boundary’ timecode value (wrap point). The next logical contiguous timecode value after this 24 hour ‘wrap point’ will be 00:00:00:00. The precise value of the timecode frame count value at the wrap point will depend on the framerate used to record timing, but the hour’s value,

minutes and seconds' values will be 23, 59 and 59 respectively.

It is anticipated that the timecode values for each frame of the video media that is associated with the subtitle content will increase by one frame count for each subsequent frame. That is, it is anticipated that there will be no timecode discontinuities in the associated media for an EBU-TT file (with the possible exception of the timecode 24 hour wrap point).

2.5 Compatibility with TTML timing model

With the constraints defined in EBU-TT the time container `tt:body` and `tt:div` have, according to the TTML event based timing model, no specified duration. However in EBU-TT the timecode values in the `begin` and `end` attributes of their children elements (`tt:p` and `tt:span`) may activate and deactivate the enclosed subtitle content nonetheless (see **Annex B** "Time container with no specified duration").

2.6 Unicode support

EBU-TT processing and transformation engines should support Unicode characters and the Unicode bidirectional algorithm (UAX9) [6].

2.7 White space handling

To indicate the authors' intention of the use of white space (spaces, tabs, and blank lines) the `xml:space` attribute may be added to a `tt:tt`, `tt:p` or `tt:span` element.

In accordance with W3C XML 1.0 Specification [7] the value "default" signals that the default white-space processing modes of the processing application are acceptable for this element; the value "preserve" indicates the intent that applications preserve all the white space.

This declared intent is considered to apply to all elements within the content of the element where it is specified, unless overridden with another instance of the `xml:space` attribute.

3. Document Structure

The order of content in this specification of the EBU-TT format follows the structure of a typical EBU-TT document instance. The levels within this specification document reflect the nested structure of an EBU-TT document.

The formal definition of how the EBU-TT specification uses EBU-TT-, TTML- and XML- vocabulary is presented in tabular form. When using this specification, the definition of the use of an element or attribute shall be interpreted relative to the position in the document instance.

Example:

The definition of the use of the `xml:id` attribute in § 3.1.2.1 "Style" specifies only the use of the `xml:id` attribute on the `tt:style` element.

Definitions used within this specification:

- Type:* Constraints of the Information structure of an XML element or XML attribute.
- Enumeration:* Enumerated values that shall be used for certain elements or attributes of type `xs:string`.
- Cardinality:* How often an element or attribute may be used inside the corresponding parent element. If the lower bound is greater than 0 (e.g. "1..1" or "1..*") the element or attribute is mandatory at this position of the document structure. If the lower bound is equal to 0 (e.g. "0..1" or "0..*") the element or attribute is optional at this position of the document structure.

Every EBU-TT document instance shall start with the `tt:tt` element. In XML terms this element is the root element of the document.

tt:tt (element)

Type	XML Content
Cardinality	1..1
Description	Root element.

The following attributes may be specified on the `tt:tt` element.

To indicate the authors' intention of the use of white space (spaces, tabs, and blank lines) the `xml:space` attribute may be added.

xml:space (attribute)

Type	<code>xs:string</code>
Enumeration	"default" "preserve"
Cardinality	0..1
Description	Indicates the author's intention of white space handling within the content of the EBU-TT document.

EBU-TT uses the following parameters from TTML to give information on how the timing information in an EBU-TT document should be interpreted. These attributes shall be specified on the `tt:tt` element.

ttp:timeBase (attribute)

Type	xs:string
Enumeration	"smpte"
Cardinality	1..1
Description	<p>The timebase defines the time coordinate system for all time expressions in EBU-TT. All time expressions in EBU-TT shall be interpreted in the time coordinate system of SMPTE 12M-1-2008. Accordingly the value for the attribute <code>ttp:timebase</code> shall always be "smpte".</p> <p>The format for the SMPTE timebased expressions is hh:mm:ss:ff</p> <p>Because SMPTE time based expressions use frames as a measurement unit, the attributes <code>ttp:frameRate</code> and <code>ttp:frameRateMultiplier</code> shall be specified for every EBU-TT document.</p>

ttp:frameRate (attribute)

Type	xs:positiveInteger
Cardinality	1..1
Description	<p>The frame rate used to interpret time expressions of the <code>tt:begin</code> and <code>tt:end</code> attributes and other time expression that are SMPTE based. The frame rate applies to the entire document instance.</p> <p>Sample Value: "25"</p>

ttp:frameRateMultiplier (attribute)

Type	ebuttdt:frameRateMultiplierType
Cardinality	1..1
Description	<p>Multiplier that shall be applied to the frame rate specified by a <code>ttp:frameRate</code> attribute in order to compute the effective frame rate.</p> <p><u>Example:</u></p> <p>The frame rate multiplier used for synchronizing with video objects at 30 frames per second is nominally 1000:1001.</p> <p>The value for the <code>ttp:frameRateMultiplier</code> attribute would accordingly be "1000 1001".</p> <p>If the frame rate is an integral number of frames per second the value for the <code>ttp:frameRateMultiplier</code> attribute is "1 1".</p>

ttp:markerMode (attribute)

Type	xs:string
Enumeration	"discontinuous"
Cardinality	1..1
Description	<p>ttp:markerMode shall always have the value "discontinuous". This is necessary to enable event-based timecode handling.</p> <p>As specified in §2.5 "Compatibility with TTML timing model" and as a further restriction the order in which the timecode values of the <code>begin</code> attributes of the <code>tt:p</code> and <code>tt:span</code> elements appear in an EBU-TT document should be linear and monotonically increasing.</p>

The attributes `ttp:cellResolution` or `tts:extent` may be specified on the `tt:tt` element to indicate the size of the active picture the subtitles are authored for.

If an EBU-TT document instance uses the 'cell' measurement unit (e.g. as part of a `tts:fontsize` attribute value) then the `ttp:cellResolution` attribute shall be specified on the `tt:tt` element.

ttp:cellResolution (attribute)

Type	ebuttdt:cellResolutionType
Cardinality	0..1
Description	<p>Expresses a virtual visual grid composed of horizontal and vertical cells. This virtual grid shall divide the active picture in rows and columns. The first value defines the number of columns and the second value defines the number of rows.</p> <p>In contrast to TTML the initial value of this attribute shall not be "32 15", but "40 24" (ETS 300 706) [8].</p>

If an EBU-TT document instance uses the 'pixel' measurement unit (e.g. as part of a `tts:fontsize` attribute value) then the `tts:extent` attribute shall be specified on the `tt:tt` element.

tts:extent (attribute)

Type	ebuttdt:extentType
Cardinality	0..1
Description	<p>Defines the width and height of the active picture the subtitles were authored for. Only length expressions in pixels shall be used.</p> <p>Sample Value: "1280px 720px"</p>

To identify the language that the subtitles are prepared for, the `xml:lang` attribute shall be specified on the `tt:tt` element.

xml:lang (attribute)

Type	xs:language
Cardinality	1..1
STL mapping	Language Code (LC)
Description	The language for which the EBU-TT document is prepared. Sample Values: "en", "en-US" or "de"

3.1 Head

The head section of an EBU-TT document carries information needed by a subtitling device to correctly present or render the contained subtitles. Specific layout and styling information shall be placed in the head of an EBU-TT document. The subtitle blocks in the body reference this information.

Furthermore, the head section shall contain metadata information.

tt:head (element)

Type	XML Content
Cardinality	1..1
Description	Container element that groups styling, layout and metadata information.

3.1.1 Metadata in tt:head

The `tt:metadata` element inside the `tt:head` element is used as a generic container for metadata information that applies to the whole document.

The content model defined in this section shall apply only to the use of the `tt:metadata` element inside the `tt:head` element.

tt:metadata (element)

Type	XML Content
Cardinality	1..1
Description	Generic metadata container.

3.1.1.1 Document Metadata

The `ebuttm:documentMetadata` element is used for EBU-TT specific metadata that applies to the whole EBU-TT document. The `ebuttm:documentMetadata` element and all the children elements of `ebuttm:documentMetadata` defined in this section shall be used only in the head section of an EBU-TT document.

ebuttm:documentMetadata (element)

Type	XML Content
Cardinality	1..1
Description	Container of the EBU-TT metadata.

New Metadata Elements

As children of the `ebuttm:documentMetadata` element, the following new metadata elements are defined by EBU-TT:

ebuttm:documentEbuttvVersion (element)

Type	<code>xs:string</code>
Enumeration	"v0.9"
Cardinality	1..1
Description	The version of the EBU-TT standard used by the document instance. Only the enumerated values published by an EBU-TT spec. shall be used.

ebuttm:documentIdentifier (element)

Type	<code>xs:string</code>
Cardinality	0..1
Description	Identifier for an EBU-TT document that may be used as external reference to an EBU-TT document. The format of the identifier may be an URI.

ebuttm:documentCopyright (element)

Type	<code>xs:string</code>
Cardinality	0..1
Description	The copyright of the document. Sample Value: "© EBU 2011"

ebuttm:documentReadingSpeed (element)

Type	xs:positiveInteger
Cardinality	0..1
Description	The intended reading speed for the subtitles in words per minute.

STL-Metadata

Note: Most of the information that is present in the GSI block of the EBU-STL specification (EBU Tech 3264) is incorporated by EBU-TT as legacy metadata. EBU-TT adds no further meaning to this metadata. If these metadata elements are used in an EBU-TT document that was not generated by an STL to EBU-TT transformation, then that use is at the discretion of the implementation or document author.

ebuttm:documentOriginalProgrammeTitle (element)

Type	xs:string
Cardinality	0..1
STL mapping	Original Programme Title (OPT)
Description	STL meaning: "The programme title in the original language."

ebuttm:documentOriginalEpisodeTitle (element)

Type	xs:string
Cardinality	0..1
STL mapping	Original Episode Title (OET)
Description	STL meaning: "The title of the episode of the programme in the original language."

ebuttm:documentTranslatedProgrammeTitle (element)

Type	xs:string
Cardinality	0..1
STL mapping	Translated Programme Title (TPT)
Description	STL meaning: "The programme title in the local language."

ebuttm:documentTranslatedEpisodeTitle (element)

Type	xs:string
Cardinality	0..1
STL mapping	Translated Episode Title (TET)
Description	STL meaning: "The title of the episode of the programme in the local language."

ebuttm:documentTranslatorsName (element)

Type	xs:string
Multiplicity	0..1
STL mapping	Translator's Name (TN)
Description	STL meaning: "Name of the translator."

ebuttm:documentTranslatorsContactDetails (element)

Type	xs:string
Cardinality	0..1
STL mapping	Translator's Contact Details (TCD)
Description	STL meaning: "The translator's contact details."

ebuttm:documentSubtitleListReferenceCode (element)

Type	xs:string
Cardinality	0..1
STL mapping	Subtitle List Reference Code (SLR)
Description	STL meaning: "Free-format character string which may be used to provide a unique reference for the subtitle list."

ebuttm:documentCreationDate (element)

Type	xs:date
Cardinality	0..1
STL mapping	Creation Date (CD)
Description	STL meaning: "The date of creation of the subtitle datafile."

ebuttm:documentRevisionDate (element)

Type	xs:date
Cardinality	0..1
STL mapping	Revision Date (RD)
Description	STL meaning: "The date of the most-recent modifications to the datafile."

ebuttm:documentRevisionNumber (element)

Type	xs:string
Cardinality	0..1
STL mapping	Revision Number (RN)
Description	STL meaning: "The revision number of the subtitle datafile may be used to specify a particular version of the subtitle list."

ebuttm:documentTotalNumbersOfSubtitles (element)

Type	xs:int
Cardinality	0..1
STL mapping	Total Number of Subtitles (TNS)
Description	STL meaning: "The number of subtitles."

ebuttm:documentMaximumNumberOfDisplayableCharacterInAnyRow (element)

Type	xs:int
Cardinality	0..1
STL mapping	Maximum Number of Displayable Characters in any text row (MNC)
Description	STL meaning: "Maximum number of displayable rows per television frame which could be occupied at any one time by the subtitles defined in the TTI blocks."

ebuttm:documentStartOfProgramme (element)

Type	ebuttdt:timingType
Cardinality	0..1
STL mapping	Timecode: Start-of-Programme (TCP)
Description	<p>STL meaning: "The time code of the first frame of the recorded video signal which is intended for transmission."</p> <p>Note: When the referenced start timecode of the video material the subtitles were authored for is greater than 00:00:00:00 (e.g. 10:00:00:00) it is recommended to specify the attribute ebuttm:documentStartOfProgramme.</p>

ebuttm:documentCountryOfOrigin (element)

Type	xs:string
Cardinality	0..1
STL mapping	Country of Origin (CO)
Description	<p>STL meaning: "The country of origin of the subtitle list."</p> <p>Note: Although three letter country codes must be supported to be compatible with STL the use of two letter country codes is recommended.</p> <p>The ebuttm:documentCountryOfOrigin element shall not be used as a substitute for the xml:lang attribute of the tt:tt element.</p>

ebuttm:documentPublisher (element)

Type	xs:string
Cardinality	0..1
STL mapping	Publisher (PUB)
Description	<p>STL meaning: "Name of the publisher of the subtitle list."</p>

ebuttm:documentEditorsName (element)

Type	xs:string
Cardinality	0..1
STL mapping	Editor's Name (EN)
Description	<p>STL meaning: "Name of the editor of the subtitle list."</p>

ebuttm:documentEditorsContactDetails (element)

Type	xs:string
Cardinality	0..1
STL mapping	Editor's Contact Details (ECD)
Description	STL meaning: "Information about the editor named in the metadata element documentEditorsName."

ebuttm:documentUserDefinedArea (element)

Type	xs:string
Cardinality	0..1
STL mapping	User-Defined Area (UDA)
Description	STL meaning: "This field may be used to carry information about the programme or subtitle list, or other relevant details"

3.1.1.2 Binary Data Tunnelling

The `ebuttm:binaryData` element may be used to transport binary data of the input formats or associated documents used to generate an EBU-TT document.

Where a sequence of source documents has been used to generate an EBU-TT document, multiple `ebuttm:binaryData` elements should be in the same order as the sources were processed.

ebuttm:binaryData (element)

Type	xs:string
Cardinality	0..*
Description	Container for transporting binary data. The binary data is encoded in one text string.

The `ebuttm:binaryData` element shall have the following attributes:

ebuttm:textEncoding (attribute)

Type	xs:string
Enumeration	"BASE64"
Cardinality	1..1
Description	Text encoding of the binary data. The text-encoding shall have the value "BASE64".

ebuttm:binaryDataType (attribute)

Type	xs:string
Cardinality	1..1
Description	Internal format of the binary data. Any format that is not defined by this document or another EBU Specification shall be prefixed with "x-". To indicate that the binary data sent in a document is an STL file, the value "EBU Tech 3264" shall be used.

The ebuttm:binaryData element may have the following attribute:

ebuttm:fileName (attribute)

Type	xs:string
Cardinality	0..1
Description	A filename that may be used to identify the original filename of the tunnelled binary data.

3.1.2 Styling in tt:head

The head section of an EBU-TT document shall contain one tt:styling element. The tt:styling element is the container for styling information. It shall contain at least one tt:style child element.

tt:styling (element)

Type	XML Content
Cardinality	1..1
Description	Container for styling information.

3.1.2.1 Style

A tt:style element defines a set of style information through style attributes. The tt:div, tt:p and tt:span elements in the body section, that enclose subtitle content, shall only use references to these specific style definitions.

Note: EBU-TT uses referenced styling. Therefore it is not required that an EBU-TT processor supports the direct specification of style attributes in the subtitle blocks (also known as inline styling).

tt:style (element)

Type	XML Content
Cardinality	1..*
Description	Set of style information.

The `tt:style` element shall have an *ID* that is unique in the entire document instance. This *ID* is used by subtitle blocks and regions to reference the style element.

xml:id (attribute)

Type	<code>xs:ID</code>
Cardinality	1..1
Description	<i>ID</i> of a <code>tt:style</code> element that is used by other elements for reference.

Note: The XML attribute `xml:id` (type `xs:ID`) is not only used by the `tt:style` element, but also by the `tt:region` and `tt:p` element. By definition, a value of type `xs:ID` must be unique in the entire document. (See the W3C Specification Extensible Markup Language (XML) 1.0 and XML Schema Part 2: Datatypes [9]).

This means for example that a `tt:style` element and a `tt:region` element must not have the same `xml:id` attribute value (e.g. "id1").

To "inherit" style information the `tt:style` element may itself reference one or more other `tt:style` elements. To reference these elements, the *ID* values of the referenced `tt:style` elements should be listed in the `tt:style` attribute.

tt:style (attribute)

Type	<code>xs:IDREFS</code>
Cardinality	0..1
Description	<i>ID(s)</i> of other style element(s). Style information from the referenced styles shall be inherited. If multiple styles are referenced the <i>IDs</i> shall be delimited by space characters (e.g. "styleId1 styleId2 styleId3").

Style information is set using attribute values of the `tt:style` element.

Note: The style attributes used in EBU-TT are a limited subset of the TTML 1.0 W3C recommendation. An EBU-TT processor is not required to support TTML style attributes that are not used in the EBU-TT specification.

Style attributes that are not used in EBU-TT are amongst others `tts:display`, `tts:opacity`, `tts:visibility`, `tts:textOutline` and `tts:zIndex`.

In addition to the TTML style attributes listed below EBU-TT defines also style attributes for the `tt:region` element (see § 3.1.3.1 "Region"). The style attributes of the `tt:style` element as well as the style attributes of `tt:region` element shall only appear inside the parent element they are defined for. This means that a style attribute that is defined for the `tt:style` element shall not appear in a `tt:region` element and vice versa.

EBU-TT supports the following style information attributes in the `tt:style` element.

tt:direction (attribute)

Type	xs:string
Enumeration	"ltr" "rtl"
Cardinality	0..1
Description	<p>Directionality if bi-directional text is used.</p> <p>Note: Bi-directional text is text containing text in both text directions, right-to-left ("rtl") and left-to-right ("ltr").</p> <p>The Arabic and Hebrew scripts, notably, are written in a form known as right-to-left ("rtl"), in which writing begins at the right-hand side of a page and concludes at the left-hand side. This is different from the left-to-right ("ltr") direction used by most languages in the world.</p>

tts:fontFamily (attribute)

Type	ebuttdt:fontFamilyType
Cardinality	0..1
Description	Font-family from which glyphs are selected.

tts:fontSize (attribute)

Type	ebuttdt:fontSizeType
Cardinality	0..1
Description	The font-size of a glyph.

tts:lineHeight (attribute)

Type	ebuttdt:lineHeightType
Cardinality	0..1
Description	<p>Inter-baseline separation between line areas.</p> <p>If the value is "normal" than the line height shall be the same as the largest font size that applies to any descendant element.</p>

tts:textAlign (attribute)

Type	xs:string
Enumeration	"left" "center" "right" "start" "end"
Cardinality	0..1
Description	<p>Alignment of inline areas in a containing block.</p> <p>The alignment values "start" and "end" depend on the writing direction of the text which may be specified on a <code>tt:region</code> element with the attribute <code>tts:writingMode</code>.</p> <p><u>Example:</u></p> <p>In a left-to-right inline writing direction "start" has the same meaning as "left" alignment while in the top-to-bottom inline writing direction the alignment value "start" expresses "top-alignment".</p>

tts:color (attribute)

Type	ebuttdt:colorType
Cardinality	0..1
Description	Foreground color of an area.

tts:backgroundColor (attribute)

Type	ebuttdt:colorType
Cardinality	0..1
Description	Background color of a subtitle or a region.

tts:fontStyle (attribute)

Type	xs:string
Enumeration	"normal" "italic"
Cardinality	0..1
Description	Font style that applies to glyphs.

tts:fontWeight (attribute)

Type	xs:string
Enumeration	"normal" "bold"
Cardinality	0..1
Description	Font weight that applies to glyphs.

tts:textDecoration (attribute)

Type	xs:string
Enumeration	"none" "underline"
Cardinality	0..1
Description	Whether a glyph is underlined.

tts:unicodeBidi (attribute)

Type	xs:string
Enumeration	"normal" "embed" "bidiOverride"
Cardinality	0..1
Description	Directional embedding or override according to the Unicode bidirectional algorithm. (see [6])

In addition to the style attributes from TTML the new style attribute `ebutts:multiRowAlign` is defined by EBU-TT. The `ebutts:multiRowAlign` attribute defines how multiple 'rows' of inline areas are aligned within a containing block area. See [Annex A](#) for a detailed description of how the attribute can be used.

ebutts:multiRowAlign (attribute)

Type	xs:string
Enumeration	"start" "center" "end" "auto"
Cardinality	0..1
Description	Alignment of multiple 'rows' of inline areas within a containing block area.

Note: Although the TTML styling attributes `tts:wrapOption`, `tts:showBackground` and `tts:overflow` are not part of the EBU-TT specification, some clarification is needed as to how a processor ought to handle the presentation characteristics that are targeted by these attributes.

The `tts:wrapOption` attribute is used in TTML to specify whether or not automatic line wrapping (breaking) applies. In TTML the initial value for `tts:wrapOption` is "wrap". This means that, by default, automatic line wrapping must be applied. In contrast to TTML, EBU-TT recommends that an EBU-TT processor does not rely upon automatic line wrapping by default. If line wrapping is required, an explicit linebreak ought to be inserted by using the `tt:br` element.

The `tts:overflow` attribute is used in TTML to specify if text in a region area must be clipped if it overflows the extent of the region. The initial value in TTML for `tts:overflow` is "hidden". In contrast to TTML, EBU-TT recommends that an EBU-TT processor presents as much active content as possible even if the bounds of the active region are exceeded.

The `tts:showBackground` attribute is used in TTML to specify when the background color of a region is present. The initial value of `tts:showBackground` is "always". In

contrast to TTML, EBU-TT recommends that an EBU-TT processor only presents the background color when active content is present in a region.

3.1.3 Layout in `tt:head`

The EBU-TT head section shall contain one `tt:layout` element. The `tt:layout` element is a container element for layout information and shall contain at least one `tt:region` child element.

tt:layout (element)

Type	XML Content
Cardinality	1..1
Description	Container for region elements.

3.1.3.1 Region

A `tt:region` element defines a space or area where subtitle content is to be placed. It specifies a set of layout information through attributes. To apply this layout information, `tt:div` and `tt:p` elements may reference a region.

tt:region (element)

Type	XML Content
Cardinality	1..*
Description	Defines a space or area for the display of subtitle content.

The `tt:region` element shall have an *ID* that is unique in the entire document instance and it is used by subtitle blocks (`tt:div` element or `tt:p` element) to refer to a particular layout.

xml:id (attribute)

Type	<code>xs:ID</code>
Cardinality	1..1
Description	<i>ID</i> of a region. This <i>ID</i> is used by <code>tt:div</code> and <code>tt:p</code> elements to reference a region. Layout and style information of the referenced region shall be applied to these elements.

The position and size of the region shall be set through the attributes `tts:extent` and `tts:origin`. The reference for `tts:extent` and `tts:origin` shall be the active picture.

Note: TTML defines with the term "root container region" a region that establishes a coordinate system into which content regions are placed. In EBU-TT the root container region is the active picture.

tts:origin (attribute)

Type	ebuttdt:originType
Cardinality	1..1
Description	The x and y coordinates of the origin of a region with respect to the active picture the document was authored for. Values in percentage shall be relative to the width and height of the active picture.

tts:extent (attribute)

Type	ebuttdt:extentType
Cardinality	1..1
Description	Width and height of a region area. Values in percentage shall be relative to the width and height of the active picture.

The `tt:region` element may also be used to apply style information to subtitles presented in this region. Therefore the `tt:region` element may reference zero or more `tt:style` elements from the styling section. The *IDs* of the referenced `tt:style` elements are specified in the `tt:style` attribute of the `tt:region` element.

tt:style (attribute)

Type	xs:IDREFS
Cardinality	0..1
Description	<i>ID(s)</i> of one or more style element(s). The style information shall be applied to <code>tt:div</code> or <code>tt:p</code> elements that reference the region. If multiple styles are referenced the <i>IDs</i> shall be delimited by space characters (e.g. "styleId1 styleId2 styleId3").

The `tt:region` element may also specify some layout specific style information:

tts:displayAlign (attribute)

Type	xs:string
Enumeration	"before" "center" "after"
Cardinality	0..1
Description	Alignment in the block progression direction. Note: In the writing mode "Left to Right Top to Bottom" this would result in the vertical alignment of lines of text. The value "before" would result in "top" alignment and the value "after" would result in "bottom" alignment.

tts:padding (attribute)

Type	ebuttdt:paddingType
Cardinality	0..1
Description	Padding (or inset) space on all sides of a region area.

tts:writingMode (attribute)

Type	xs:string
Enumeration	"lrtb" "rltb" "tblr" "tblr" "lr" "rl" "tb"
Cardinality	0..1
Description	<p>Writing mode of subtitle content.</p> <ul style="list-style-type: none"> • "lrtp": "Left to Right Top to Bottom" • "rltb": "Right to Left Top to Bottom" • "tblr": "Top to Bottom Right to Left" • "tblr": "Top to Bottom Left to Right" • "lr": Shorthand for "Left to Right Top to Bottom" • "rl": Shorthand for "Right to Left Top to Bottom" • "tb": Shorthand for "Top to Bottom Right to Left"

3.2 Body

The body section of an EBU-TT document carries the content of the subtitle and the timing information. Styling and layout shall be applied through references to specific `tt:style` and `tt:region` elements in the header section.

The `tt:body` element shall be the container for subtitle and timing information.

tt:body (element)

Type	XML content
Cardinality	1..1
Description	Container for subtitle and timing information.

Subtitle content shall be contained within the following elements of the body-section:

- `tt:p` (logical paragraph)
- `tt:span` (inline element to apply local style information or metadata)

Timing information shall be set using the following attributes:

- `tt:begin`
- `tt:end`

Timing information shall be specified on the `tt:p` element and may be specified on a `tt:span` element.

3.2.1 Div

The `tt:div` element shall be a logical container of textual content. A `tt:div` element may contain `tt:div` children elements to create a nested structure of `tt:div` elements.

tt:div (element)

Type	XML content
Cardinality	1..*
Description	Container for textual content.

To apply layout- and style-information the `tt:div` element may reference a `tt:region` element using a `tt:region` attribute.

tt:region (attribute)

Type	<code>xs:IDREF</code>
Cardinality	0..1
Description	Application of layout and style information through reference of a region.

To apply style information a `tt:div` element may also reference a `tt:style` element directly with a `tt:style` attribute.

tt:style (attribute)

Type	<code>xs:IDREFS</code>
Cardinality	0..1
Description	<i>ID(s)</i> of one or more style element(s). The style information shall be applied to the enclosed content of the <code>tt:div</code> element. If multiple styles are referenced the <i>IDs</i> shall be delimited by space characters (e.g. "styleId1 styleId2 styleId3").

3.2.1.1 Paragraph

A `tt:p` element shall represent a logical paragraph. The `tt:p` element may have textual content and zero or more `tt:span` elements.

tt:p (element)

Type	Mixed content.
Cardinality	1..*
Description	Logical paragraph.

The `tt:p` element shall have an *ID* that is unique in the entire document. This *ID* shall represent the unique *ID* of a subtitle. No meaningful subtitle sequence should be inferred from the value of this *ID*.

Note: Typically this *ID* will be a monotonically (logically) increasing value through the EBU-TT document (e.g. sub1, sub2, sub3 or sub1, sub2, sub2a, sub2b, sub3).

xml:id (attribute)

Type	xs:ID
Cardinality	1..1
Description	Unique <i>ID</i> of a subtitle.

To indicate the authors intention of the use of white space (spaces, tabs, and blank lines) the `xml:space` attribute may be added.

xml:space (attribute)

Type	xs:string
Enumeration	"default" "preserve"
Cardinality	0..1
Description	Indicates the author's intention of white space handling within the content of the <code>tt:p</code> element.

To overwrite the language identification of the enclosed subtitle content the `xml:lang` attribute may be specified on the `tt:p` element.

xml:lang (attribute)

Type	xs:language
Cardinality	0..1
Description	Language identifier that overwrites the language identification of the enclosed subtitle content. Sample Values: "en", "en-US" or "de" Note: The type <code>xs:language</code> allows identifiers as defined in RFC 3066, which is based on ISO 639 "Code for the representation of names of languages." [10]

To apply layout and style information the `tt:p` element may reference a `tt:region` element using a `tt:region` attribute.

tt:region (attribute)

Type	xs:IDREF
Cardinality	0..1
Description	Application of layout information through reference of a region.

To apply style information a `tt:p` element may also reference a `tt:style` element directly with a `tt:style` attribute.

tt:style (attribute)

Type	xs:IDREFS
Cardinality	0..1
Description	<i>ID(s)</i> of one or more style element(s). The style information shall be applied to the enclosed content of the <code>tt:p</code> element. If multiple styles are referenced the <i>IDs</i> shall be delimited by space characters (e.g. "styleId1 styleId2 styleId3").

The timing information of a `tt:p` element is set through the attributes `tt:begin` and `tt:end`.

tt:begin (attribute)

Type	ebuttdt:timingType
Cardinality	1..1
Description	Start point of a temporal interval associated with a <code>tt:p</code> element.

tt:end (attribute)

Type	ebuttdt:timingType
Cardinality	1..1
Description	End point of a temporal interval associated with a <code>tt:p</code> element.

A `tt:br` element may be used to insert a forced line break.

tt:br (element)

Type	Empty element
Cardinality	0..1
Description	Forced line break.

3.2.1.1.1 Span

The `tt:p` element may have zero or more `tt:span` element(s). The `tt:span` element may be used to apply style information to the enclosed textual content. This style information is added to or overwrites style information from the currently active context.

A `tt:span` element may also have other `tt:span` elements as children (nested spans).

tt:span (element)

Type	Mixed content.
Cardinality	0..*
Description	Inline element to allow the application of local style information, annotation or metadata.

To indicate the author's intention of the use of white space (spaces, tabs, and blank lines) the `xml:space` attribute may be added to the `tt:span` element.

xml:space (attribute)

Type	<code>xs:string</code>
Enumeration	"default" "preserve"
Cardinality	0..1
Description	Indicates the authors' intention of white space handling within the content of the <code>tt:span</code> element.

To overwrite the language identification of the enclosed subtitle content the `xml:lang` attribute may be specified in the `tt:span` element.

xml:lang (attribute)

Type	<code>xs:language</code>
Cardinality	0..1
Description	Language identifier that overwrites the language identification of the enclosed subtitle content. Sample Values: "en", "en-US" or "de" Note: The type <code>xs:language</code> allows identifiers as defined in RFC 3066 which is based on ISO 639 "Code for the representation of names of languages."

To apply style information the `tt:span` element may reference style elements in the styling section of the document through the `tt:style` attribute.

tt:style (attribute)

Type	<code>xs:IDREFS</code>
Cardinality	0..1
Description	<i>ID(s)</i> of one or more style element(s). The style information shall be applied to the enclosed content of the span element. If multiple styles are referenced the <i>IDs</i> shall be delimited by space characters (e.g. "styleId1 styleId2 styleId3").

Timing information may be applied in a `tt:span` element through the attributes `tt:begin` and `tt:end`.

Note: Although it is not restricted by this specification, it is anticipated that the begin and end timecode values for a `tt:span` element will logically fall inside the timecode values defined by a parent `tt:p` element.

tt:begin (attribute)

Type	ebuttdt:timingType
Cardinality	0..1
Description	Start point of a temporal interval associated with a tt:span element.

tt:end (attribute)

Type	ebuttdt:timingType
Cardinality	0..1
Description	End point of a temporal interval associated with a tt:span element.

A tt:br element may be used to insert a forced line break.

tt:br (element)

Type	Empty element
Cardinality	0..1
Description	Forced line break.

4. Datatypes

EBU-TT defines specific datatypes to restrict the value space of certain scalar values.

Note: If a datatype is applied to an attribute or element that was taken from TTML the restriction of the datatype is equal to the definition in TTML or it is a further restriction of the value space as defined in TTML. Therefore all values that conform to the EBU-TT datatypes also conform to the values allowed in TTML. However it is possible that a value conforms to the TTML value space but does not conform to the EBU-TT datatypes.

4.1 ebuttdt:cellResolutionType

The value space shall be constrained to two numbers greater than zero delimited by a space. The first value shall define the number of columns and the second value shall define the number of rows.

4.2 ebuttdt:colorType

Note: The value space is the same as the <color> expression in TTML.

The value space shall be constrained to a named color string, a RGB color triple, RGBA color tuple, a hex notated RGB color triple or a hex notated RGBA color tuple.

Although different formats may be used in an EBU-TT document the value should be expressed in RGB or RGBA.

Named colors shall be:

- "transparent"
- "black"
- "silver"
- "gray"
- "white"
- "maroon"
- "red"
- "purple"
- "fuchsia"
- "magenta"
- "green"
- "lime"
- "olive"
- "yellow"
- "navy"
- "blue"
- "teal"
- "aqua"
- "cyan"

Note: The color black can, for example, be expressed as:

- "black" (named color)
- "rgb(0, 0, 0)" (RGB color triple)
- "rgba(0, 0, 0, 255)" (RGBA color tuple)
- "#000000" (RGB color triple in hex notation)
- "#000000FF" (RGBA color tuple in hex notation)

4.3 *ebuttdt:extentType*

The value space shall be constrained to two values of type *ebuttdt:lengthType* delimited by a space. The first value shall express the width and the second value the height.

4.4 *ebuttdt:fontFamilyType*

Note: The value space of the *ebuttdt:fontFamilyType* is the same as the value space defined in the TTML style value expressions <familyName> and <genericFamilyName>.

The value space shall be constrained to one or more font family- and/or generic family-names.

The value space for a font family name (e.g. "Arial" or "Verdana") shall not be restricted.

Generic family names shall be:

- "default"
- "monospace"
- "sanserif"
- "serif"
- "monospaceSansSerif"
- "monospaceSerif"
- "proportionalSansSerif"
- "proportionalSerif"
- "monospace"

The typographic characteristics of the generic family name "default" may be implementation dependent; however the default generic font family should be mapped to a monospaced, sans-serif font.

4.5 *ebuttdt:fontSizeType*

The value space is constrained to one or two values of type *ebuttdt:lengthType*.

If a single value is specified, then this length applies equally to horizontal and vertical scaling of a glyph's square; if two values are specified, then the first expresses the horizontal scaling and the second expresses vertical scaling.

Note: Use of independent horizontal and vertical font sizes is expected to be used with cell based metrics in order to denote fonts that are two rows in height and one column in width.

4.6 *ebuttdt:framerateMultiplierType*

The value space shall be constrained to two numbers greater than zero and delimited by a space. The value shall represent a fraction. The first number shall be the numerator and the second number shall be the denominator.

4.7 *ebuttdt:lengthType*

Note: The *ebuttdt:lengthType* of TTML is equal to the <length> expression defined in TTML except that the measurement parameter *em* is not allowed.

The value space shall be constrained to a number appended either by the percentage sign “%” or the measurement units “px” (for video pixel) or “c” (for cell).

If an EBU-TT document instance uses the ‘cell’ measurement unit (e.g. as part of a *tts:fontsize* attribute value) then the *ttp:cellResolution* attribute shall be specified on the *tt:tt* element.

If an EBU-TT document instance uses the ‘pixel’ measurement unit (e.g. as part of a *tts:fontsize* attribute value) then the *tts:extent* attribute of the *tt:tt* element shall be specified.

4.8 *ebuttdt:lineHeightType*

The value shall be the string “normal” or shall be of *ebuttdt:lengthType*.

4.9 *ebuttdt:originType*

The value space shall be constrained to two values of type *ebuttdt:lengthType* delimited by a space. The first value shall express a x-coordinate and the second value a y-coordinate.

4.10 *ebuttdt:paddingType*

The value space shall be constrained to one or four values of type *ebuttdt:lengthType* delimited by a space. If four values are specified then the first value shall apply to the before edges, the second value to the end edges, the third value to the after edges and the fourth value to the start edges of a region.

If only one value is specified the value shall apply to all four edges of a region.

4.11 *ebuttdt:timingType*

A value of type *ebuttdt:timingType* shall conform with the time coordinate defined by SMPTE 12M-1-2008 and shall have the format

hh:mm:ss:ff

Where *hh* is hours, *mm* is minutes, *ss* is seconds and *ff* is frames.

5. Bibliography

- [1] EBU Tech 3264 Specification of the EBU Subtitling data exchange format.
<http://tech.ebu.ch/docs/tech/tech3264.pdf>

- [2] EBU R 133 Recommendation on transport of subtitles inside & outside MXF files (Draft).

- [3] TTML 1.0 Glenn Adams. Timed Text Markup Language (TTML) 1.0. W3C Rec.
18 November 2010. <http://www.w3.org/TR/ttaf1-dfxp/>

- [4] SMPTE ST 2052 Society of Motion Picture and Television Engineers, "SMPTE Standard for
Television - Timed Text Format (SMPTE-TT)", SMPTE ST 2052-1:2010.

- [5] SMPTE-12M-1-2008 Society of Motion Picture and Television Engineers, "SMPTE Standard for
Television - Time and Control Code", SMPTE 12M-1-2008.

- [6] UAX9 Mark Davis. Unicode Standard Annex #9. Unicode Bidirectional Algorithm.
<http://unicode.org/reports/tr9/>

- [7] XML 1.0 Tim Bray, et al. Extensible Markup Language (XML) 1.0 (Fifth Edition), W3C
Recommendation, 26 November 2008.
<http://www.w3.org/TR/2008/REC-xml-20081126/>

- [8] ETS 300 706 Enhanced Teletext specification.

- [9] XML Schema
Part 2 Paul Biron and Ashok Malhotra, XML Schema Part 2: Datatypes, W3C
Recommendation, 28 October 2004. <http://www.w3.org/TR/xmlschema-2/>

- [10] RFC 3066 H. Alvestrand, ed. RFC 3066: Tags for the Identification of Languages 1995.
<http://www.ietf.org/rfc/rfc3066.txt>

Annex A: Use of `ebutts:multiRowAlign`

The `ebutts:multiRowAlign` attribute may be used to specify a style property that defines how multiple 'rows' of inline are aligned within a containing block area. This attribute acts as a 'modifier' to the action defined by the `tts:textAlign` attribute value, whether that value is explicitly or implicitly defined. This attribute effectively creates additional alignment points for multiple rows of text, thus it has no effect if only a single row of text is present.

This modifier acts as follows: For multiple 'rows' of inline blocks, 3 additional multi-row alignment points ("start", "center", "end") are created by the rendered dimensions of the longest row within the `p` element. 'Rows' that are shorter than the longest row shall be each aligned against the longest row using the multi-row alignment point identified by the `ebutts:multiRowAlign` attribute value. The longest 'row' shall always be aligned within the region in accordance with the value of the `tts:textAlign` attribute.

Note: The combination of `tts:textAlign="start"` with `ebutts:multiRowAlign="start"` acts identically to the use of `tts:textAlign="start"` without the `ebutts:multiRowAlign` attribute.

Or more simply, if the `ebutts:multiRowAlign` attribute has the same value as `tts:textAlign`, the `ebutts:multiRowAlign` attribute has no effect.

If the term "auto" is used the basic behaviour of `tts:textAlign` is maintained unmodified (i.e. the presentation is as if `ebutts:multiRowAlign` would have the same computed value as `tts:textAlign`).

The use of `tts:textAlign` together with `ebutts:multiRowAlign` provides for combinations of text alignment as tabled below, where the highlighted combinations may be specified by the use of just the `tts:textAlign` attribute from the W3C TTML specification.

tts:textAlign	ebutts:multiRowAlign	Presentation
"start"	"start"	Start justified text. All 'rows' are aligned at start.
"start"	"left"	The longest 'row' is start aligned. Shorter 'rows' are left aligned against the left alignment point created by the longest 'row'.
"start"	"center"	The longest 'row' is start aligned. Shorter 'rows' are center aligned against the center alignment point created by the longest 'row'.
"start"	"end"	The longest 'row' is start aligned. Shorter 'rows' are end aligned against the end alignment point created by the longest 'row'.
"left"	"start"	The longest 'row' is left aligned. Shorter 'rows' are start aligned against the start alignment point created by the longest 'row'.
"left"	"center"	The longest 'row' is left aligned. Shorter 'rows' are center aligned against the center alignment point created by the longest 'row'.
"left"	"end"	The longest 'row' is left aligned. Shorter 'rows' are end aligned against the end alignment point created by the longest 'row'.
"center"	"start"	The longest 'row' is center aligned. Shorter 'rows' are start aligned against the start alignment point created by the longest 'row'.
"center"	"center"	Center justified text. All 'rows' are individually center aligned.
"center"	"end"	The longest 'row' is center aligned. Shorter 'rows' are end aligned against the end alignment point created by the longest 'row'.
"right"	"start"	The longest 'row' is right aligned. Shorter 'rows' are start aligned against the start alignment point created by the longest 'row'.
"right"	"center"	The longest 'row' is right aligned. Shorter 'rows' are center aligned against the center alignment point created by the longest 'row'.
"right"	"end"	The longest 'row' is right aligned. Shorter 'rows' are end aligned against the end alignment point created by the longest 'row'.

tts:textAlign	ebutts:multiRowAlign	Presentation
"end"	"start"	The longest 'row' is end aligned. Shorter 'rows' are start aligned against the start alignment point created by the longest 'row'.
"end"	"center"	The longest 'row' is end aligned. Shorter 'rows' are center aligned against the center alignment point created by the longest 'row'.
"end"	"end"	End justified text. All 'rows' are aligned at end.

If a specified value of this attribute is not supported, then a presentation processor shall interpret the attribute as if the attribute has the value "auto" (i.e. the basic behaviour of `tts:textAlign` is maintained unmodified).

The `ebutts:multiRowAlign` style is illustrated by the following example.

```

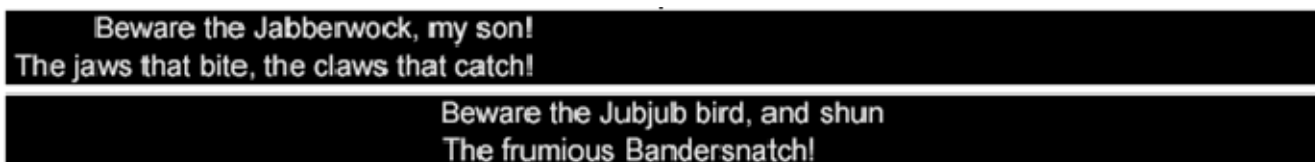
...
<tt:styling>
  <tt:style xml:id="baseStyle" tts:backgroundColor="black" tts:color="white" />
  <tt:style xml:id="startEnd" style="baseStyle" tts:textAlign="start" ebutts:multiRowAlign="end" />
  <tt:style xml:id="centerStart" style="baseStyle" tts:textAlign="center" ebutts:multiRowAlign="start" />
</tt:styling>

<tt:layout>
  <tt:region xml:id="regionTop" style="startEnd" tts:extent="355px 43px" tts:origin="0px 0px" />
  <tt:region xml:id="regionBottom" style="centerStart" tts:extent="355px 43px" tts:origin="0px 47px" />
</tt:layout>

...
<tt:p xml:id="subtitle1" region="regionTop" begin="00:00:00:00" end="00:00:03:00">
  Beware the Jabberwock, my son!<tt:br/>
  The jaws that bite, the claws that catch!
</tt:p>
<tt:p xml:id=" subtitle2" region=" regionBottom" begin="00:00:00:00" end="00:00:03:00">
  Beware the Jubjub bird, and shun<tt:br/>
  The frumious Bandersnatch!
</tt:p>
...

```

Produces:



* Page intentionally left blank. This document is paginated for two sided printing

Annex B: Time container with no specified duration

Note: The intent of the following section is to provide additional information for TTML implementations that take the SMIL semantics as a reference.

For a “par”- time container with no specified duration the implicit default value of the attribute “endsync” shall be “all”.

Note: TTML describes the behaviour of time containers such as the whole TTML timing model with reference to SMIL 2.1 semantics. In TTML not only are the elements `tt:p` and `tt:span` defined as time containers, but `xtt:body` and `tt:div` as well. Because EBU-TT `tt:body` and `tt:div` have no `begin` and `end` attributes, some clarification is needed as to how this fits in the SMIL event- based timing model of TTML.

In terms of SMIL, the timing model of EBU-TT is event based. This timing model, combined with the marker mode chosen by EBU-TT, implements a “publish and subscribe”- pattern where the elements subscribe to timecode-marker events (e.g. “00:00:03:00”) and get activated (and deactivated) when a specific timecode-marker is “published”. As a result of a constraint in the SMIL event-based timing model `tt:p` and `tt:span` elements do not receive any timecode-marker events unless their parent time containers, `tt:body` and `tt:div`, are active. To be active, the duration of the parent time container must be greater than zero.

The default settings in TTML and SMIL imply that the duration of the time containers with no specified duration is calculated from the resolved duration of their children. If the duration of the children cannot be resolved than the duration of the parent time container is zero.

In terms of the SMIL semantics, the timecode-marker events in EBU-TT are regarded as any other unpredictable event. Therefore the duration of `tt:p` and `tt:span` cannot be resolved and their parent time container `tt:body` and `tt:div` have a duration of zero, which means that they never get active.

To “enable” the event reception of timecode-marker events for `tt:p` and `tt:span` EBU-TT overwrites a default value in the SMIL timing model as follows:

For a “par”- time container with no specified duration the implicit default value of the attribute “endsync” in TTML is “last”. In EBU-TT this implicit default value is “all”.

Through this change the calculated duration of a time container with no specified duration can be greater than zero even when the duration of their children cannot be resolved. In practice this simply means that `tt:p` and `tt:span` receive “their” timecode-marker events for `begin` and `end` although the parent time containers `tt:body` and `tt:div` have no specified duration.

* Page intentionally left blank. This document is paginated for two sided printing

Annex C: Support of the STL GSI Block in EBU-TT

The following characters are used to indicate if an element or attribute was defined for the STL information in EBU-TT:

- No element or attribute defined
- X Supported (Element or attribute defined for the STL Information)

STL Information	Mnemonic	EBU-TT Support
Code Page Number	CPN	-
Disk Format Code	DFC	-
Display Standard Code	DSC	-
Character Code Table Number	CCT	-
Language Code	LC	X
Original Programme Title	OPT	X
Original Episode Title	OET	X
Translated Programme Title	TPT	X
Translated Episode Title	TET	X
Translator's Name	TN	X
Translator's Contact Details	TCD	X
Subtitle List Reference Code	SLR	X
Creation Date	CD	X
Revision Date	RD	X
Revision Number	RN	X
Total Number of Text Timing Information (TTI) blocks	TNB	-
Total Numbers of Subtitles	TNS	X
Total Numbers of Subtitle Groups	TNG	-
Maximum Number of Displayable Character in any text Row	MNC	X
Maximum Number of Displayable Rows	MNR	-
Time Code: Status	TCS	-
Time Code: Start-of-Programme	TCP	X
Time Code: First In-Cue	TCF	-
Total Numbers of Disks	TND	-
Disk Sequence Number	DSN	-
Country of Origin	CO	X
Publisher	PUB	X
Editor's Name	EN	X
Editor's Contact Details	ECD	X
User-Defined Area	UDA	X

* Page intentionally left blank. This document is paginated for two sided printing

Annex D: Initial Values of TTML and EBU-TT attributes

The following table lists the TTML attributes where EBU-TT adopts the TTML initial values:

Attribute Name	TTML Initial Value
<code>tts:backgroundColor</code>	"transparent"
<code>tts:direction</code>	"ltr"
<code>tts:displayAlign</code>	"before"
<code>tts:fontFamily</code>	"default"
<code>tts:fontSize</code>	"1c" (The semantics of the unit of measure c [cell] are defined by the attribute <code>ttp:cellResolution</code> parameter.)
<code>tts:fontStyle</code>	"normal"
<code>tts:fontWeight</code>	"normal"
<code>tts:lineHeight</code>	"normal"
<code>tts:padding</code>	"0px"
<code>tts:textAlign</code>	"start"
<code>tts:textDecoration</code>	"none"
<code>tts:unicodeBidi</code>	"normal"
<code>tts:writingMode</code>	"lrbt"

The following table list the TTML attributes where EBU-TT overwrites the TTML initial values:

Attribute Name	TTML Initial Value	EBU-TT Initial Value
<code>tts:color</code>	Implementation dependent.	Implementation dependent. Note: It is recommended that in the absence of end-user preference information, a conformant presentation processor uses white or yellow.
<code>ttp:cellResolution</code>	"32 15"	"40 24"

Note: Some attributes in EBU-TT are mandatory and have a fixed enumerated value (e.g. `ttp:timebase` with the fixed value "smpte"). As the initial value of these attributes never applies they are not listed in this section.

The following table lists the initial values of the attribute defined in the EBU-TT Style namespace.

Attribute Name	EBU-TT Initial Value
<code>ebutts:multiRowAlign</code>	"auto"

* Page intentionally left blank. This document is paginated for two sided printing

Annex E: Index of elements, attributes and types

ebuttdt:cellResolutionType	16, 37	tt: styling (element).....	25
ebuttdt:colorType.....	27, 37	tt:begin (attribute)	35, 37
ebuttdt:extentType.....	16, 30, 38	tt:body (element).....	32
ebuttdt:fontFamilyType.....	26, 38	tt:br (element)	35, 37
ebuttdt:fontSizeType.....	27, 38	tt:direction (attribute).....	26
ebuttdt:framerateMultiplierType	15, 39	tt:div (element)	33
ebuttdt:lengthType.....	38, 39	tt:end (attribute).....	35, 37
ebuttdt:lineHeightType	27, 39	tt:head (element).....	17
ebuttdt:originType.....	30, 39	tt:layout (element).....	29
ebuttdt:paddingType	31, 39	tt:metadata (element)	12, 17
ebuttdt:timingType.....	22, 35, 37, 40	tt:p (element).....	33
ebuttm:binaryData (element).....	24	tt:region (attribute).....	33, 34
ebuttm:binaryDataType (attribute).....	24	tt:region (element).....	29
ebuttm:documentCopyright (element)	18	tt:span (attribute)	35
ebuttm:documentCountryOfOrigin (element)	22	tt:style (attribute)	26, 31, 33, 35, 36
ebuttm:documentCreationDate (element).....	21	tt:style (element)	25
ebuttm:documentEbuttVersion (element).....	18	tt:tt (element)	14
ebuttm:documentEditorsContactDetails (element)	23	ttp: frameRate (attribute)	15
ebuttm:documentEditorsName (element)	23	ttp: frameRateMultiplier (attribute)	15
ebuttm:documentIdentifier (element)	18	ttp:cellResolution (attribute)	16
ebuttm:documentMaximumNumberOfDisplayableCharacterIn AnyRow (element)	22	ttp:markerMode (attribute).....	16
ebuttm:documentMetadata (element)	18	ttp:timeBase (attribute).....	15
ebuttm:documentOriginalEpisodeTitle (element).....	19	tts: displayAlign (attribute)	31
ebuttm:documentOriginalProgrammeTitle (element)	19	tts:backgroundColor (attribute).....	27
ebuttm:documentPublisher (element).....	23	tts:color (attribute)	27
ebuttm:documentReadingSpeed (element).....	19	tts:extent (attribute)	16, 30
ebuttm:documentRevisionDate (element).....	21	tts:fontFamily (attribute)	26
ebuttm:documentRevisionNumber (element)	21	tts:fontSize (attribute)	27
ebuttm:documentStartOfProgramme (element)	22	tts:fontStyle (attribute).....	28
ebuttm:documentSubtitleListReferenceCode (element)	20	tts:fontWeight (attribute)	28
ebuttm:documentTotalNumbersOfSubtitles (element)	21	tts:lineHeight (attribute).....	27
ebuttm:documentTranslatedEpisodeTitle (element)	20	tts:origin (attribute)	30
ebuttm:documentTranslatedProgrammeTitle (element)	19	tts:padding (attribute)	31
ebuttm:documentTranslatorsContactDetails (element).....	20	tts:textAlign (attribute).....	27
ebuttm:documentTranslatorsName (element)	20	tts:textDecoration (attribute)	28
ebuttm:documentUserDefinedArea (element)	23	tts:unicodeBidi (attribute).....	28
ebuttm:fileName (attribute).....	24	tts:writingMode (attribute).....	32
ebuttm:textEncoding (attribute)	24	xml:id (attribute).....	25, 30, 34
ebutts:multiRowAlign (attribute).....	28, 41	xml:lang (attribute)	17, 34, 36
		xml:space (attribute).....	14, 34, 36