SMPTE STANDARD

SMPTE ST 428-7:2010
Revision of
SMPTE 428-7-2007

Digital Cinema Distribution Master — Subtitle

Table of Contents

Approved
December 15, 2010

Foreword

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

SMPTE Engineering Documents are drafted in accordance with the rules given in Part XIII of its Administrative Practices.

SMPTE ST 428-7 was prepared by Technology Committee 21DC.

Intellectual Property

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

1 Scope

This document specifies the format of a DCDM Subtitle file. A DCDM Subtitle file contains a set of instructions for placing rendered text or graphical overlays at precise locations on distinct groups of motion picture frames. A DCDM Subtitle File is an integral component of a D-Cinema composition and may be present in Digital Cinema Package (DCP) file sets. Consequently, its design reflects features of other DCP file formats. The DCDM Subtitle file format is not intended for use in streaming applications.

2 Conformance Notation

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

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

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

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

The keywords "may" and "need not" indicate courses of action permissible within the limits of the document.
The keyword “reserved” indicates a provision that is not defined at this time, shall not be used, and may be defined in the future. The keyword “forbidden” indicates “reserved” and in addition indicates that the provision will never be defined in the future.

3 Normative References

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

1. World Wide Web Consortium (W3C) (2004, February 4). Extensible Markup Language (XML) 1.0 (Third Edition)

2. World Wide Web Consortium (W3C) (2004, October 28). XML Schema Part 1: Structures (Second Edition)

3. World Wide Web Consortium (W3C) (2004, October 28). XML Schema Part 2: Datatypes (Second Edition)

4. Internet Engineering Task Force (IETF) (1996, November). RFC 2396 — Uniform Resource Identifiers (URI): Generic Syntax

5. Internet Engineering Task Force (IETF) (2005, July). RFC 4122 — A Universally Unique Identifier (UUID) URN Namespace

6. ISO/IEC 15948:2004 Information Technology — Computer Graphics and Image Processing — Portable Network Graphics (PNG): Functional Specification

7. ISO/IEC 14496, Part 18, 2004, Font Compression and Streaming

8. ISO/IEC 10646-1, 2000 Information Technology — Universal Multiple-Octet Coded Character Set (UCS)

4 Overview

A DCDM subtitle file, depicted in block form in Figure 1, is a representation of a series of *subtitle instances*: rendered text or graphical overlays on a primary picture in a d-cinema work, such as a motion picture, trailer, or advertisement. A subtitle track file contains a set of file-global metadata and a set of subtitle structures which encode the content and temporal and spatial locations of the subtitles to be displayed over the primary image.

A virtual timeline provides the temporal dimension of the DCDM Subtitle file. The timeline is a contiguous set of *editable units*. Spatial positions are expressed as percentages of the primary picture's frame size relative to a chosen border. The primary picture source must have constant frame size.

The DCDM Subtitle file may reference external font resources for rendered text, and external image resources for graphical overlays. The font and image resource formats are specified in this document's normative references.

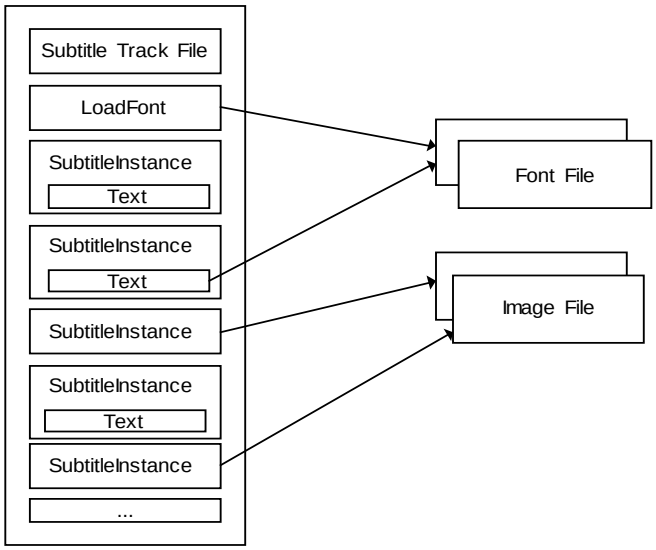


Figure 1 – Prototypical DCDM Subtitle File (Informative)

The structures defined in this document are represented using the *Extensible Markup Language* [XML 1.0], and specified using XML Schema [XML Schema Part 1: Structures] and [XML Schema Part 2: Datatypes].

This specification shall be associated with a unique XML namespace name [Namespaces in XML], The namespace name shall be the string value "`http://www.smpte-ra.org/schemas/428-7/2010/DCST`". Table 1 lists the XML namespace names used in this specification. Namespace names are represented as Uniform Resource Identifier (URI) values [RFC 2396][1].

The MIME type for a document containing a single SubtitleReel element as its root shall be `text/xml`.

[1] Readers unfamiliar with URI values as XML namespace names should be aware that although a URI value begins with a "method" element ("http" in this case), the value is designed primarily to be a unique string and does not necessarily correspond to an actual on-line resource. Applications implementing this standard should not attempt to resolve URI values on-line.

Table 1 – XML Namespaces (Normative)

Qualifier	*URI*
dcst	`http://www.smpte-ra.org/schemas/428-7/2010/DCST`
xs	`http://www.w3.org/2001/XMLSchema`

An XML document must be parsed in its entirety before it can be considered valid, thus the format defined by this document is not suitable for streaming applications.

The URIs found in Table 1 are normative. The namespace qualifier values (also called namespace prefixes in XML jargon) used in Table 1 and elsewhere in this document, namely "dcst" and "xs", are not normative. Specifically, they may be replaced in instance documents by any XML compliant namespace prefix. In other words, implementations shall expect any arbitrary XML compliant namespace prefix value that is associated with a URI from table 1.

Datatypes from other schemas that are used in this document will be prefixed with the appropriate namespace qualifier (e.g. `xs:dateTime`). See [XML Schema Part 2: Datatypes] and [XML-Signature Syntax and Processing] for further information about these types.

4.1 Deprecated Namespace Names

The namespace names listed below were defined by previous versions of this standard. Documents conforming to those versions shall be considered conformant to the Standard, but these namespace names are not intended for new documents.

http://www.smpte-ra.org/schemas/428-7/2007/DCST

5 DCDM Subtitle Structure (Normative)

A DCDM Subtitle file is an XML document consisting of a single SubtitleReel element as defined by this document.

5.1 SubtitleReel element

The `SubtitleReel` element (Figure 2) is the top level container of XML data in a DCDM Subtitle file. The `SubtitleReel` element contains a set of elements which define the global parameters of the track file, followed by the `SubtitleList` element which contains the set of elements which describe the individual subtitle instances.

5.2 Id element

The `Id` element uniquely identifies the reel for asset management purposes. It is encoded in the element body as a urn:uuid [RFC 4122].

5.3 ContentTitleText element

The `ContentTitleText` element contains an unbounded string value encoding of the subtitle file's title, e.g., "*Gone With the Wind*". It is meant strictly for display to the user.

5.3.1 Language attribute [optional]

Indicates the language represented by the text in the `ContentTitleText` element. Encoded as an `xs:language` type. The default value is `en` (English).

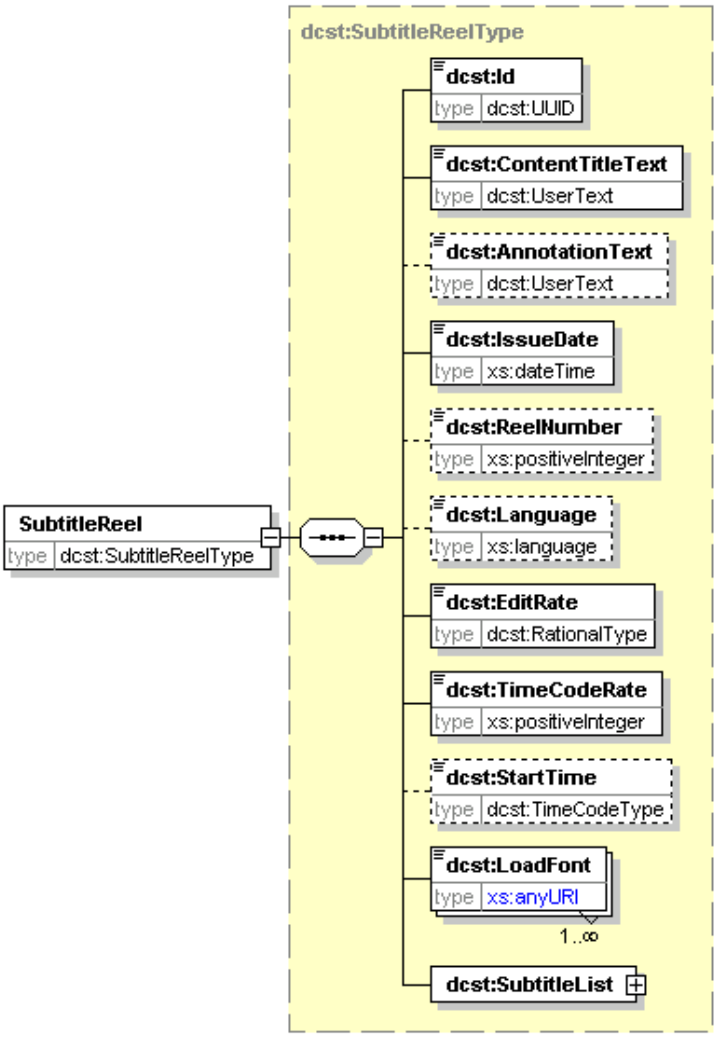


Figure 2 – DCDM Subtitle file structure. The dotted lines denote optional elements. (Informative)

5.4 AnnotationText element [optional]

The `AnnotationText` element contains an unbounded string value, which describes the subtitle reel essence. It is meant strictly for display to the user.

5.4.1 Language attribute [optional]

Indicates the language represented by the text in the `AnnotationText` element. Encoded as an `xs:language` type. The default value is `en` (English).

5.5 IssueDate element

The `IssueDate` parameter indicates the time and date at which the subtitle file was created. It may be displayed to the user. Encoded as an XML `xs:dateTime` type.

5.6 ReelNumber element [optional]

The `ReelNumber` element denotes the ordered placement of this file in a set of DCDM Subtitle files. The value shall only be used for informative purposes and shall not influence the reproduction of the subtitle instances. The value is a positive integer in the range *1 .. n*, where *n* is the total number of reels in a composition.

5.7 Language element [optional]

The `Language` element denotes the language used in the text elements and/or image resources. The value is encoded as an `xs:language` type. The default value is `en` (English).

5.8 EditRate

The EditRate element encodes the rate at which the virtual timeline progresses. It is expressed as a ratio of two integers giving the number of editable units per second of real time.

5.9 TimeCodeRate

The `TimeCodeRate` element records the count of editable units for each increment of the seconds field of a time code. It is expressed as a non-negative integer value. The editable units field of a time code shall have a value between 0 (zero) and `TimeCodeRate` – 1.

5.10 StartTime element [optional]

The `StartTime` element specifies the starting time of the DCDM Subtitle file's timeline. The timeline extends forward in time in discrete increments (editable units). The number of units per second is determined by the TimeCodeRate element (defined above).

The start time shall be encoded as a time code of the form `HH:MM:SS:EE`, or, hours, minutes, seconds and editable units, respectively. The maximum value that can be expressed is `23:59:59:ee`, where 'ee' is one number less than the value of `TimeCodeRate`. The default value is `01:00:00:00.`

5.11 LoadFont element

The `LoadFont` element is used to declare an OpenType [ISO/IEC 14496, Part 18] font resource for use within the DCDM Subtitle file. The font is identified by a urn:uuid in the element's body. The mapping of urn:uuid values to actual font resources is beyond the scope of this document.

The text rendering processes (with or without help from the font resource) shall be responsible for mapping characters outside the font's character set to the null glyph. At no time shall the presence of a character outside the font's character set be visible in the rendered subtitle instance. For the purpose of this standard, Unicode [ISO/IEC 10646-1] control codes (the 65 characters in the ranges U+0000..U+001F and U+007F..U+009F) shall be considered outside the character set of any font (control codes shall not be displayed).

When one or more `Text` elements are present, at least one `LoadFont` element shall also be present. The first `LoadFont` element shall be the default font for any text that is not explicitly styled with a `Font` element (see Section 6.4 below).

5.11.1 *ID attribute*

The `LoadFont` element accepts a single attribute, `ID`, which creates an internal (to the parent `SubtitleReel`) identifier to be referenced by `Font` elements in the `SubtitleList`. Each `LoadFont` element in a DCDM Subtitle file shall have a distinct `ID` attibute value.

5.12 SubtitleList element

The `SubtitleList` element contains the set of structures that define subtitle instances. Any combination of `Font` and `Subtitle` elements may be present. `Subtitle` elements may contain `Text` or `Image` elements, and optional `Font` elements may be present at various levels to control text rendering. Figure 3 illustrates this structure.

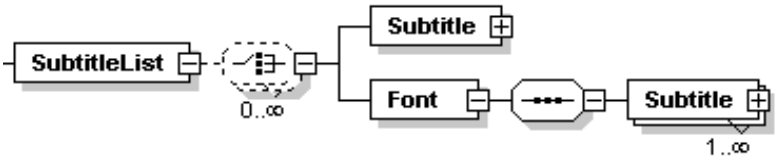


Figure 3 – SubtitleList contents. The dotted lines denote optional elements (Informative)

5.12.1 *Ordering of Subtitle Elements*

`Subtitle` elements shall be ordered within the `SubtitleList` element in ascending temporal order (least recent first). The `TimeIn` attribute of a given `Subtitle` element shall be equal to or greater than that of the preceding element. The first `Subtitle` element in the `SubtitleList` shall have a `TimeIn` value equal to or greater than the value of the `StartTime` element (or the default start time if the `StartTime` element is not present. This ordering shall not be affected by the presence of enclosing `Font` elements.

5.12.2 *Display Precedence*

Subtitle instances having overlapping temporal and spatial regions shall be rendered in ascending temporal order. The instance having the most recent `TimeIn` value will occlude those in the same spatial region having earlier `TimeIn` values. In the case of two or more instances having the same `TimeIn` value, the instance which occurs latest in the track file shall be considered most recent.

5.13 *Display Type [optional]*

The `DisplayType` element indicates whether the content of the subtitle document has been constrained to accommodate a particular type of display (e.g., on-screen or off-screen.) The element value is a text string that is meant to be both human and machine-readable. An optional “scope” attribute with default URI value `“http://www.smpte-ra.org/schemas/428-7/2010/DCST#display-type”` determines the permissible values of the element. If the scope attribute is absent, or set to its default value, the content of the element shall be `“MainSubtitle”`.

6 Subtitle Instances (Normative)

A subtitle instance is an auxiliary image to be displayed over a series of identically sized primary images. Subtitle instance images are given either directly in externally referenced Portable Network Graphics [PNG] image resources, or are rendered from integral text using externally referenced OpenType font resources.

Subtitle instances in a DCDM Subtitle file are contained within the `SubtitleList` XML element (see Section 5.12). The `Subtitle` element is the XML container of a subtitle instance. Each `Subtitle` element contains either one or more `Image` elements or one or more `Text` or `Font` elements. `Font` elements may appear at any level within the `SubtitleList` hierarchy, except within `Subtitle` elements which contain an `Image` element. Figure 4 illustrates this structure.

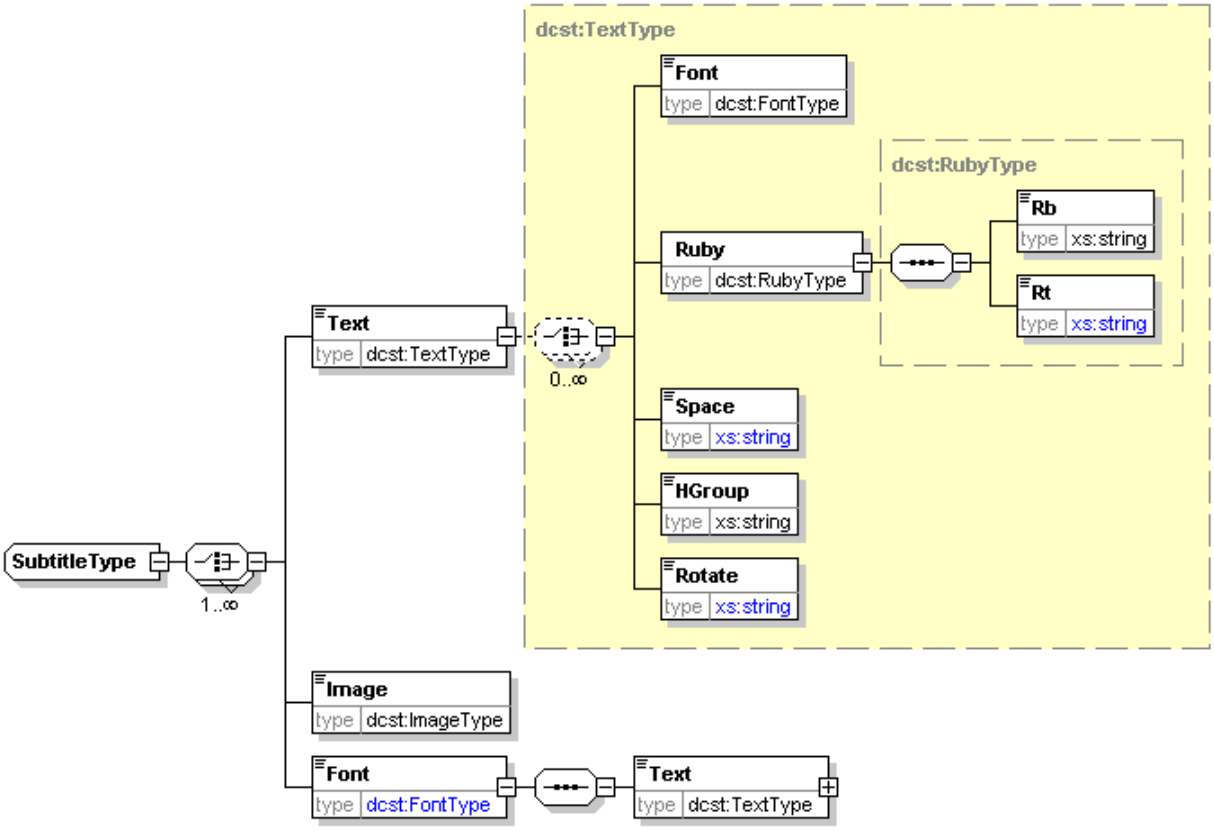


Figure 4 – Subtitle instance structure. The dotted lines denote optional elements. (Informative)

6.1 Subtitle element

The Subtitle element has attributes which specify the start time, duration, and fade characteristics of the subtitle instance. Subtitle elements must be ordered in ascending temporal order within the SubtitleList element, and may have overlapping temporal regions.

6.1.1 *SpotNumber attribute [optional]*

An unbounded string value, which identifies the instance in some other notation system, such as a spotting list.

6.1.2 *TimeIn attribute*

The time, expressed as hours, minutes, seconds and editable units (`HH:MM:SS:EE`) since the start of the timeline (see Sections 5.8 and 5.9), at which the subtitle instance shall begin to fade in.

6.1.3 *TimeOut attribute*

The time, expressed as hours, minutes, seconds and editable units (`HH:MM:SS:EE`) since the start of the timeline, at which the subtitle instance shall complete the fade out. This value shall be greater than the `TimeIn` value of the same element.

6.1.4 *FadeUpTime attribute [optional]*

The duration, expressed as hours, minutes, seconds and editable units (`HH:MM:SS:EE`), of the fade of the subtitle instance from 100% transparent to the full specified opacity. The default value shall be `00:00:00:02`.

6.1.5 *FadeDownTime attribute [optional]*

The duration, expressed as hours, minutes, seconds and editable units (`HH:MM:SS:EE`), of the fade of the subtitle instance from full specified opacity to 100% transparent. The default value shall be `00:00:00:02`.

6.2 Image element

The `Image` element identifies an external image resource to be displayed over the primary image. The image resource is identified by a urn:uuid in the element's body. The mapping of urn:uuid values to actual image resources is beyond the scope of this document. Images are encoded in Portable Network Graphics [PNG] format, and shall have a frame size no larger than the primary picture frame size. The image position over the primary picture shall be determined by the `Image` element attribute values.

6.2.1 *Halign attribute [optional]*

Horizontal alignment mode. Determines the side of the image to be used when calculating position with `Hposition`. One of [`left`, `center`, `right`]. The default value is `center`.

6.2.2 *Hposition attribute [optional]*

Specifies the distance of the subtitle instance from the side of the primary picture's frame. The distance is expressed as a percentage of primary picture width between the reference points chosen by the `Halign` attribute value. Table 2 lists the values of `Halign` and the resulting interpretation of `Hposition`. The value is encoded as a signed decimal number having one or more optional decimal places. The default value is 0 (zero).

Table 2 – Halign and Hposition interaction (Normative)

Halign Value	*Hposition Interpretation*
`Left`	The distance between the left primary image border and the left subtitle instance border. No negative values allowed.
`Center`	The distance between the center of the primary image and the center of the subtitle instance. Negative values indicate left offset, positive indicate right.
`Right`	The distance between the right primary image border and the right subtitle instance border. No negative values allowed.

6.2.3 *Valign attribute [optional]*

Vertical alignment mode. Determines the side of the image to be used when calculating position with `Vposition`. One of [`top, center, bottom`]. The default value is `center`.

6.2.4 *Vposition attribute [optional]*

Specifies the distance of the subtitle instance from the side of the primary picture's frame. The distance is expressed as a percentage of primary picture height between the reference points chosen by the `Valign` attribute value. Table 3 lists the values of `Valign` and the resulting interpretation of `Vposition`. The value is encoded as a signed decimal number having one or more optional decimal places. The default value is 0 (zero).

Table 3 – `Valign` and `Vposition` interaction (Normative)

Valign Value	*Vposition Interpretation*
`top`	The distance between the top primary image border and the top subtitle instance border. No negative values allowed.
`center`	The distance between the center of the primary image and the center of the subtitle instance. Negative values indicate top offset, positive indicate bottom.
`bottom`	The distance between the bottom primary image border and the bottom subtitle instance border. No negative values allowed.

6.3 Text element

The `Text` element encapsulates text to be rendered into the subtitle instance. The text string should be short enough to fit in a single line on screen using the given font and position parameters. Multiple lines of text to be displayed on screen simultaneously shall be encoded as sibling `Text` elements under the same `Subtitle` element.

The position and direction of the rendered text are controlled by attributes of the `Text` element.

6.3.1 *Halign attribute [optional]*

Horizontal alignment mode. Determines the side of the image to be used when calculating position with `Hposition`. One of [`left`, `center`, `right`]. The default value is `center`.

6.3.2 *Hposition attribute [optional]*

Specifies the distance of the subtitle instance from the side of the primary picture's frame. The distance is expressed as a percentage of primary picture width between the reference points chosen by the `Halign` attribute value. Table 2 lists the values of `Halign` and the resulting interpretation of `Hposition`. The value is encoded as a signed decimal number having one or more optional decimal places. The default value is 0 (zero).

6.3.3 *Valign attribute [optional]*

Vertical alignment mode. Determines the side of the text area to be used when calculating position with `Vposition`. One of [`top`, `center`, `bottom`]. The default value is `center`.

6.3.4 *Vposition attribute [optional]*

Specifies the distance of the subtitle instance from the side of the primary picture's frame. The distance is expressed as a percentage of primary picture height between the reference points chosen by the `Valign` attribute value. Table 3 lists the values of `Valign` and the resulting interpretation of `Vposition`. The value is encoded as a signed decimal number having one or more optional decimal places. The default value is 0 (zero).

6.3.5 *Direction attribute [optional]*

Specifies the direction in which the text will be rendered. One of the values in Table 4 shall be used. The default value is `ltr`.

Table 4 – Direction attribute values (Normative)

Direction Value	*Direction of Rendered Text*
`ltr`	left-to-right
`rtl`	right-to-left
`ttb`	top-to-bottom
`btt`	bottom-to-top

6.4 Font element

The `Font` element identifies a previously loaded font (see `LoadFont`, Section 5.11 above) to be used for rendering all text content enclosed within the `Font` element. When rendering text, the `Font` element closest in the `SubtitleReel` hierarchy (nearest parent) has precedence. Attributes given with the `Font` element further define font properties. Attributes not given in a particular `Font` element instance will be inherited from the nearest parent.

The `Font` element may contain `Subtitle` or `Text` elements. `Font` elements may also contain text data, in the case where the particular `Font` element is a child of a `Text` element.

6.4.1 *ID attribute [optional]*

Name of the font to use. This value must match one of the values defined by the `LoadFont` elements. If this attribute is missing or contains an unknown value, the current font shall be used.

6.4.2 *Script attribute [optional]*

Instructs the processor to render text offset from the centerline. The value of the attribute determines the direction of the offset. One of the values in Table 5 shall be used. The default value is `normal`.

Table 5 – Script attribute values (Normative)

Script Value	*Centerline Offset Mode*
`super`	offset above centerline of text
`sub`	offset below centerline of text
`normal`	on centerline

6.4.3 *Effect attribute [optional]*

Enables a special effect to alter the contrast of the rendered text against the primary picture. One of the values in Table 6 shall be used. The default value is `none`.

Table 6 – `Effect` attribute values (Normative)

Effect Value	*Effect Description*
`border`	Draw colored border around each character. The color is determined by EffectColor attribute.
`shadow`	Draw colored drop shadow around each character. The color is determined by EffectColor attribute.
`none`	No effect is applied to the rendered text.

6.4.4 *Italic attribute [optional]*

Enables the italic font characters. The value shall be one of `yes` or `no`. The default value is `no`.

6.4.5 *Underline attribute [optional]*

Enables underlining of characters. The value shall be one of `yes` or `no`. The default value is `no`.

6.4.6 *Weight attribute [optional]*

Specifies the weight of the characters. The value shall be one of `bold` or `normal`. The default value is `normal`.

6.4.7 *Color attribute [optional]*

The color to be used for rendering text characters. The color is encoded as a string of 8 (eight) hexadecimal characters in the following format: `AARRGGBB`, where `AA` is an 8 bit alpha-blend value, and `RR`, `GG`, and `BB` represent 8 bits of red, green and blue, respectively. The default value is `FFFFFFFF` (opaque white).

6.4.8 *EffectColor attribute [optional]*

The color to be use for rendering the effect chosen by the `Effect` attribute. The format is identical to that used by the `Color` attribute of this element. The default value is `FF000000` (opaque black).

6.4.9 *Size attribute [optional]*

The size of the characters, expressed in points. Fonts are rendered as if the primary picture frame size is 11 inches in height, so a 72pt font will be 1/11 frame height. The default value is 42.

6.4.10 *AspectAdjust attribute [optional]*

Used to adjust the aspect ratio of the font. This attribute can be used to increase or decrease the width of each rendered character. This can be used to lengthen or shorten the character string slightly. The value shall be encoded as a decimal number. Values greater than 1.0 will widen each character and result in a longer string. Values less than 1.0 will narrow each character and result in a shorter string. Values shall be limited to not less than 0.25 and not more than 4.0. The default value shall be “1.0”.

6.4.11 *Spacing attribute [optional]*

Provides additional spacing between the rendered characters. The spacing is specified in units of em. 1 em is equivalent to the current font size and 0.5 em is equivalent to half the current font size. The value shall be encoded as a decimal number. This attribute can be used to increase or decrease the amount of space between adjacent characters. This can be used to lengthen or shorten the character string slightly. Typical usage would have values less than 1 em. Negative values are allowed but should be used with care as characters could overlap. Negative spacing shall be limited to no more than -1.0 em. The default value shall be “0.0”.

6.5 Ruby Element

The `Ruby` element is a container used to specify ruby characters that are to be associated with a set of base characters in Asian text, specifically Japanese. The `Ruby` element shall contain exactly one (1) `Rb` element, and one (1) `Rt` element. No other text or elements shall be present inside the `Ruby` element. The `Ruby` element shall be present only inside a `Text` element.

6.6 Rb Element

The `Rb` or RubyBase element is a container used to specify the set of base characters that ruby text will be associated with in Asian text, specifically Japanese. The `Rb` element shall be present only inside a `Ruby` element. There shall be one (1) `Rb` element inside a `Ruby` element. No elements shall be present inside an `Rb` element.

6.7 Rt Element

The `Rt` or RubyText element is a container used to specify the actual ruby characters that will be associated with the base characters specified in a corresponding `Rb` element in Asian text, specifically Japanese. The `Rt` tag shall be present only inside a `Ruby` element. No elements shall be present inside an `Rt` element. There shall be one (1) `Rt` element inside a `Ruby` element. The `Rt` element shall have five attributes, `Size`, `Position`, `Offset`, `Spacing`, and `AspectAdjust`. These are described in the following sub sections. The ruby characters shall be centered with respect to the base characters in all cases.

6.7.1 *Size attribute [optional]*

Indicates the size of the rendered ruby characters. Character sizes shall be specified in units of em. 1 em shall be equivalent to the current font size for the base characters. 2 em shall be equivalent to twice the size of the current font size, and 0.5 em shall equivalent to half the size of the current font size. Values for `size` shall be positive numbers (greater than 0). Default `size` = 0.5.

6.7.2 *Position attribute [optional]*

Indicates the position of the rendered ruby characters with respect to the base characters. Valid values are `before`, and `after`. Specifying `before` shall indicate that the ruby characters shall be rendered above the base characters if the text direction is horizontal on screen. Specifying `before` shall indicate that the ruby characters are rendered to the right of the base characters if the text direction is vertical on screen. Specifying `after` shall indicate that the ruby characters are rendered below the base characters if the text direction is horizontal on screen. Specifying `after` shall indicate that the ruby characters are rendered to the left of the base characters if the text direction is vertical on screen. Default `position` = `before`.

6.7.3 *Offset attribute [optional]*

Indicates the offset or amount of space between the rendered ruby characters and the base characters. The offset shall be specified in units of em. 1 em shall be equivalent to the current font size for the base

characters. 2 em shall be equivalent to twice the size of the current font size, and 0.5 em shall be equivalent to half the size of the current font size. Negative offset shall not exceed -1em. Default `offset` = 0.

6.7.4 *Spacing attribute [optional]*

Indicates additional spacing between the rendered ruby characters. The spacing shall be specified in units of em. 1 em shall be equivalent to the current font size for the base characters, and 0.5 em shall be equivalent to half the size of the current font size. This attribute can be used to increase or decrease the amount of space between adjacent ruby characters. This can be used to lengthen or shorten the ruby character string slightly. Typical usage would have values less than 1em. Negative values are allowed but should be used with care as characters could overlap. Negative spacing shall be limited to no more than -1.0em. Note: spacing can be specified here as well as in the Font element. The effect of both shall be cumulative. Default `spacing` = 0.

6.7.5 *AspectAdjust attribute [optional]*

Used to adjust the aspect ratio of the rendered ruby characters. This attribute can be used to increase or decrease the width of each character. This can be used to lengthen or shorten the character string slightly. Values greater than 1.0 will widen each character and result in a longer string. Values less than 1.0 will narrow each character and result in a shorter string. Values shall be limited to not less than 0.25 and not more than 4.0. Default `AspectAdjust` = 1.0.

6.8 Ruby Examples

Example 1:

```
<Text Direction=”horizontal” HAlign=“left" HPosition=“11.4" VAlign="top"
VPosition="95.6">

<Ruby>

<Rb>新幹線</Rb>

<Rt Size=”0.5em” Position=“before” Offset=“0.5em” Spacing=”0em”>しんかんせん</Rt>

</Ruby>

</Text>
```

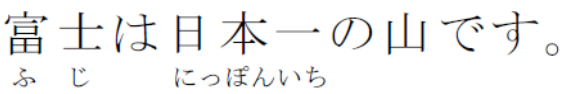
Should produce something like this:

しんかんせん
新幹線

Example 2:

```
<Text Direction = “horizontal” HAlign=“left" HPosition=“11.4" VAlign="top"
VPosition="85.6">

<Ruby>

<Rb>富士</Rb>

<Rt Size=”0.5em” Position =“After” Offset=“0.5em”>ふじ</Rt>

</Ruby>

は

<Ruby>

<Rb>日本一</Rb>

<Rt Size=”0.5em” Position =“After” Offset=“0.5em”>にっぽんいち</Rt>

</Ruby>

の山です。

</Text>
```

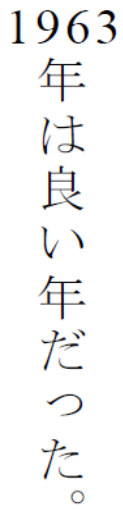
Should produce something like this:



Example 3:

```
<Text Direction=”vertical” HAlign=”right” HPosition=”10.0” VAlign=”top”
VPosition=”8.25”> <HGroup>1963</HGroup> 年は良い年だった。</Text>
```

Should produce something like this:



6.9 Space Element

The Space element provides a mechanism to insert a variable width amount of space in the middle of a rendered text string. The amount of space to insert is indicated by using the Size attribute described in the following section. The Space element shall be present only inside a Text element. No text or elements shall be present inside a Space element; the Space element has only attributes. The Space element may be encoded as an empty element.

6.9.1 *Size attribute [optional]*

Indicates the size of the space to be inserted. Size is specified in units of em. 1 em shall be equivalent to the current font size for the characters being rendered. 2 em shall be equivalent to twice the size of the current font size, and 0.5 em shall be equivalent to half the size of the current font size. Values for size will typically be positive numbers. Negative values are allowed but should be used with care as characters may overlap. Negative spacing shall be limited to no more than -1.0em. Default Size = 0.5.

6.10 HGroup Element

The HGroup element is a container used to specify a string of characters that are to be rendered horizontally in a subtitle that has text direction set to vertical. This is most common for multi-digit numbers in an Asian text string. This is a temporary override for the text direction specified as an attribute in the Text element. The HGroup element shall be present only inside a Text element. No elements shall be present inside an HGroup element.

6.11 Rotate Element

The Rotate element is a container used to specify a string of characters that are to be rotated either right or left by 90 degrees before they are rendered on screen. This is most common for special characters in an Asian text string that is being displayed vertically. The Rotate element shall be present only inside a Text element. Font and Rotate elements shall not be present inside a Rotate element. The Rotate element has one attribute, Direction which is described below. No elements shall be present inside a Rotate element.

6.11.1 *Direction attribute [optional]*

Indicates the direction of the character rotation. Specifying none indicates that the characters are not rotated. Specifying right shall indicate that the characters are rotated to the right or clockwise. Specifying left shall indicate that the characters are rotated to the left or counter-clockwise. Default Direction = none.

7 Sample (Informative)

The following DCDM Subtitle sample XML structure is a valid instance of the `SubtitleReel` schema. XML namespace declarations have been omitted for clarity. This reel starts at `00:00:00:00`. A default font is loaded and then three subtitle instances are displayed.

```
<?xml version="1.0" encoding="UTF-8"?>
<dcst:SubtitleReel xmlns:dcst="http://www.smpte-ra.org/schemas/428-7/2010/DCST">
 <Id>urn:uuid:fbf6e056-0a6e-4dd8-8003-0a914481ed87</Id>
 <ContentTitleText>Example</ContentTitleText>
 <AnnotationText>This is a test file</AnnotationText>
 <IssueDate>2005-07-14T21:52:02.000-00:00</IssueDate>
 <ReelNumber>1</ReelNumber>
 <Language>en</Language>
 <EditRate>24 1</EditRate>
<dcst:TimeCodeRate>24</dcst:TimeCodeRate>
 <StartTime>00:00:00:00</StartTime>
 <LoadFont ID="Arial">urn:uuid:3dec6dc0-39d0-498d-97d0-928d2eb78391</LoadFont>
 <SubtitleList>
  <Font ID="Arial" Color="FFFFFFFF" Weight="normal" Size="40">
   <Subtitle SpotNumber="1" TimeIn="00:01:34:17" TimeOut="00:01:40:20">
    <Text Valign="top" Vposition="10.00">These are not the droids you're looking for.</Text>
   </Subtitle>

   <Subtitle SpotNumber="2" TimeIn="00:01:41:10" TimeOut="00:01:45:20">
    <Text Valign="top" Vposition="30.00"><Font Italic="yes">[Trooper]</Font> These are not the
                          droids we're looking for.</Text>
   </Subtitle>

   <Subtitle SpotNumber="3" TimeIn="00:01:50:01" TimeOut="00:01:56:20">
    <Image Valign="top" Vposition="10.00">urn:uuid:0392ad89-30a2-471c-b289-c210ab8b371e</Image>
   </Subtitle>
  </Font>
 </SubtitleList>
</dcst:SubtitleReel>
```

8 XML Schema (Normative)

The XML Schema document presented in this section normatively defines the structure of a DCDM Subtitle file using a machine-readable language [XML Schema Part 1][XML Schema Part 2]. While this schema is intended to faithfully represent the structure presented in the normative prose portions (Sections 5 and 6) of this document, conflicts in definition may occur. In the event of such a conflict, the normative prose shall be the authoritative expression of the standard.

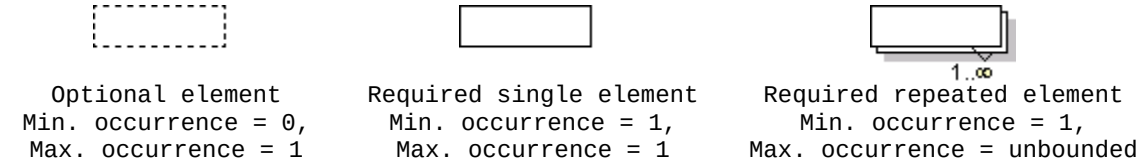
Note: The schema has been removed from this document to facilitate maintenance and processing with tools suitable to the purpose. If you are reviewing this document, you should have been provided with an up-to-date version of the schema in electronic form. The name of the file should be "DCDMSubtitle.xsd".

9 XML Diagram Legend (Informative)

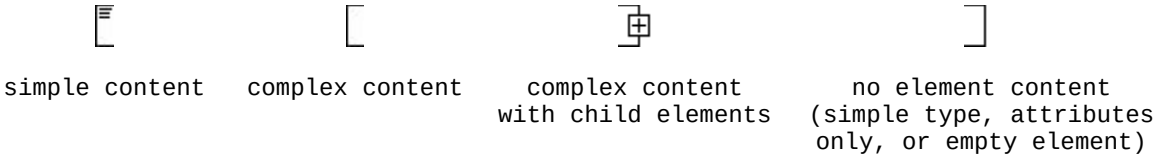
The following provides a legend for notation used in diagrams depicting XML structures.

9.1 Element symbols

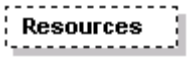
In the schema design diagrams, presented above in this document, only the elements are drawn. Attributes are not visible. The cardinality of the element (0..1, 1 exactly, 0..n, 1..n) is indicated by the border of the elements. Optional elements are drawn with a dashed line, required elements with a solid line. A maximum occurrence greater one is indicated by a double border.



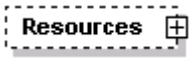
The content model of elements is symbolized on the left and right side of the element boxes. The left side indicates whether the element contains a simple type (text, numbers, dates, etc.) or a complex type (further elements). The right side of the element symbol indicates whether it contains child elements or not:



9.1.1 *Examples*



Optional single element without child elements. Minimum Occurrence = 0, Maximum Occurrence = 1, content = complex.



As above, but with child elements. The "plus" at the right side indicates the presence of one or more undisplayed child elements.

Mandatory single element. Minimum Occurrence = 1, Maximum Occurrence = 1, content = complex, no child elements (i.e. this denotes an *empty element*). The gray or green text below the element displays the xml-schema annotation associated with the element.

Mandatory multiple element containing child elements (content = complex). This element must occur at least once (Minimum Occurrence = 1) and may occur as often as desired (Maximum Occurrence = unbounded).

Mandatory single element with containing simple content (e.g. text) or mixed complex content (e.g. text with xhtml markup). Minimum Occurrence = 1, Maximum Occurrence = 1, type = xsd:string (for example), content = simple. The three lines in the upper left corner are used for both text and numeric content.

9.2 Model symbols ("compositors")

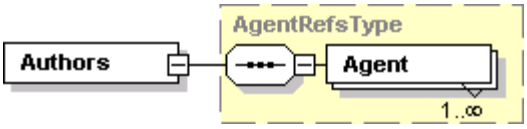
A sequence of elements. The elements must appear exactly in the sequence in which they appear in the schema diagram.

A choice of elements. Only a single element from those in the choice may appear at this position.

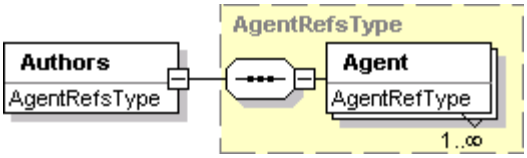
The "all" model, in which the sequence of elements is not fixed.

9.3 Types

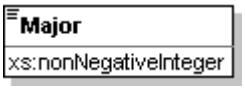
If an element refers to a complex global type, the type is shown with a border and a gray or shadowed background.



The type name may be shown in the line below the element name:



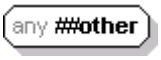
In that case, the type names of simple types are shown as well:



9.4 Model groups and references

An *element group* is a named container with one or several elements. The group of elements can be reused at multiple places in the schema. Model groups are invisible in the instance document. Model groups have been used sparingly since they do not map to a feature in object-oriented programming languages (unless they support multiple inheritance).

Import note on reading the diagrams for model groups: If the model group symbol is drawn with simple lines (i.e. not dashed), this does not imply that the elements in the model group are required. The optionality of the group depends on the optionality of elements contained in the model group. (Model groups can be made optional, e.g., to make a model group with required elements optional in some cases, but this has not been used.)



The *"any"* group is a special kind of model group. It is a placeholder for elements not defined in the schema. The "any" element defines points where the schema can be extended. After the "Any" keyword the namespace from which the elements may come is defined, for example, "##other" specifies that the extension elements may come from any namespace, except from the current schema namespace.

Element references are indicated through a link arrow in the lower left corner. They are similar to references to model groups within a schema, but instead of refining the model group, they directly refer to a single global element. The global element can then be reused in multiple places.

Annex A (Informative)
Bibliography

Internet Engineering Task Force (IETF) (1994, December). RFC 1738 – *Uniform Resource Locators (URL)*. http://www.ietf.org/rfc/rfc3986.txt

RDDL - Resource Directory Description Language J. Border and T. Bray 2002. http://www.rddl.org/

World Wide Web Consortium (W3C) - *Namespaces in XML,* http://www.w3.org/TR/REC-xml-names/

World Wide Web Consortium (W3C) - *QA Framework: Specification Guidelines, Formal Languages*, http://www.w3.org/TR/2004/WD-qaframe-spec-20041122/

World Wide Web Consortium (W3C) – *XML Schema Primer,* http://www.w3.org/

World Wide Web Consortium (W3C) – *Ruby Annotation* , http://www.w3.org/TR/ruby/