REVISIONS			
REV	DESCRIPTION	DATE	APPROVED
A	Original Release (ECO 2034536)	03/11/2003	Tim Ryan
B	Remove proprietary markings, and incorporate additions, and further descriptions	05/12/2003	Tim Ryan
C	Incorporate requests for ruby characters, horizontal grouping, variable space, and rotated characters.	03/31/2005	Tim Ryan

Subtitle Specification
(XML File Format)
for
DLP Cinema[TM] Projection Technology

(Version 1.1)

Data in this specification is subject to correction or change as required.

2504382		DWN/CHK Tim Ryan	DATE	TITLE Subtitle Specification (XML File Format) for DLP Cinema[TM] Projection Technology			
2502643		ENGR Tim Ryan	DATE				
2504840	314PH		DATE				
NHA	Used On	QA	DATE	SIZE **A**	DRAWING NO 2504760		REV C
APPLICATION		APVD Bill Werner	DATE	SCALE NONE		SHEET 1 OF 36	

Subtitle Specification (XML File Format)

IMPORTANT NOTICE

DRAWING NO	REV	
2504760	C	SHEET 2

Subtitle Specification (XML File Format)

Preface

Read This First

About This Manual

This document describes the XML file format used by the DLP Cinema™ Subtitling Process. It is intended as a reference.

How to Use This Manual

Chapter 1 describes relevant parts of the XML standard, while Chapter 2 describes each supported tag in detail. Chapter 3 has an example of a subtitle file expressed as an XML document.

Notational Conventions

XML examples are shown in a `special typeface`.

Related Documentation From Texas Instruments

No other documents should be required to create a complete subtitle file for a DLP Cinema^{TM} projector.

Trademarks

The Texas Instruments logo and Texas Instruments are registered trademarks of Texas Instruments. Trademarks of Texas Instruments include: TI, DLP, DLP Cinema, CineCanvas, Digital Light Processing, DMD and Digital Micromirror Device.

Contents

DRAWING NO	REV	
2504760	C	SHEET 4

Examples

DRAWING NO	REV	
2504760	C	SHEET 5

Tables

Subtitle Specification (XML File Format)

Chapter 1

XML Standards

===

XML is a standard established by the World Wide Web Consortium (W3C), which is the central standards body of the web. The DLP Cinema™ subtitle XML format, as part of the CineCanvas^{TM} technology, follows the 1.0 version of the standard, and can be found at http://www.w3.org/TR/2000/REC-xml-20001006.

The following sections of the XML standard are particularly important with regards to a DLP Cinema™ subtitle XML file.

1.1 Well-formed XML Documents

Section 2.1 of the standard defines what is necessary for an XML document to be well-formed. All DLP Cinema™ subtitle XML files must be well-formed and valid. A document type declaration (DTD) specific to the DLP Cinema™ subtitle format is defined to validate against. The DTD is shown in Appendix A.

1.2 Character Sets

A well-formed XML file is made up of <u>Unicode</u> characters, expressed in some <u>encoding</u>. The DLP Cinema™ subtitling capability supports both UTF-8 and UTF-16 encodings.

Encoding should be indicated via use of the "encoding=" declaration in the <u>prolog</u> of the XML file.

1.3 XML Prolog

All well-formed XML files must begin with a prolog. The prolog is used for several things:

It definitely establishes the file as being in XML format.

It can specify which version of XML is used by the file.

It can specify a character set encoding (such as UTF-8 or UTF-16).

It can specify a DTD which provides a grammar for the document.

DLP Cinema™ subtitling expects the following prolog (although the encoding may differ as described above):

```
<?xml version="1.0" encoding="UTF-8"?>
```

1.4 Predefined Entities

Since XML uses the '<', '>' and '&' characters for special purposes, their use as content must be escaped.

Subtitle Specification (XML File Format)

Similarly, any Unicode character can be specified by using its decimal code-point preceded by "&#" and terminated with ";". For example, "A" represents the character 'A'.

Unicode characters can also be specified using hexadecimal notation by preceding its code-point value with "&#x". For example, "A" represents the character 'A'.

1.5 Elements

The basic building blocks of any XML document are the elements. Each element is delimited by open and close tags. An open tag begins with a '<', contains the element name, and ends with a '>'. A close tag begins with a '</', contains the same element name as the open tag, and ends with a '>'.

```
<MyElement> … </MyElement>
```

Everything between the open and close tags is the value of the element. Elements can be nested, and constitute a parent / child relationship. If an element has no value, you do not have to specify both open and close tags for that element. It can be represented by an empty tag as shown.

```
<MyEmptyElement />
```

1.6 Attributes

Attributes are used to tell something about, or add description to, an element without affecting its value. Elements may have attributes and no value at all, hence the empty tag described above. Attributes are inherited from parent to child.

```
<MyElement MyAttribute="something"> … </MyElement>

A more relevant example might be:

<Font Size="24" Color="FFFFFFFF"> … </Font>
```

1.7 Binary Data

Since XML is always text, there is currently no standard way to include binary data within the body of an XML document. Normally, an XML document is expected to point to binary data in external files using URLs.

This specification does not support embedded binary data at this time.

1.8 Comments

Comments in XML must begin with "<!--" and end with "-->". For example:

```
<!-- this is an example XML comment -->
```

DRAWING NO	REV	
2504760	C	SHEET 8

Chapter 2

DLP Cinema™ Subtitling XML Tags

There are two distinct types of DLP Cinema™ subtitling files. The first type, called the **Presentation List**, is a list of subtitling files. This acts like a table of contents for an entire presentation. The second type, called the **Presentation Data**, contains the actual subtitles themselves. Each portion, or reel, of a presentation can have an individual presentation data file.   This will allow the replacement of reels of a presentation and the accompanying subtitles easily. Therefore a presentation list consists of one or more presentation data files. If the presentation is not divided into multiple reels, then there is no need to specify a Presentation List. This chapter is a reference to all of the XML tags used in a DLP Cinema™ subtitling file, and what their individual requirements and restrictions are.

2.1 DCSubtitle

All data inside a well formed XML document must be included in the document element. There can be only one document element per XML file. The document element for a Digital Cinema Subtitle XML file is denoted with the **DCSubtitle** tag.

DCSubtitle properties

Attribute	Value
Version	Specifies the version of this subtitle XML file. The current version is 1.1. Future versions of the DLP Cinema™ subtitling system may support additional or reduced capabilities. The version is a required attribute. Projectors at a DLP Cinema™ release prior to release 9.0 will require Version="1.0". Projectors at DLP Cinema™ release 9.0 and higher will require Version="1.X". All XML constructs added in future releases of this specification will be ignored if the DLP Cinema™ release is not sufficient to recognize it.

Example 1. DCSubtitle usage

```
<?xml version="1.0" encoding="UTF-8"?>

<DCSubtitle Version="1.1">

  ... XML subtitle contents ...

</DCSubtitle>
```

Subtitle Specification (XML File Format)

DLP Cinema™ Software DCSubtitle Version Compatibility

DLP Cinema™ Software Release	DCSubtitle Version="1.0"	DCSubtitle Version="1.1"
Prior to 6.0	No	No
6.0 – 8.1	Yes	No
9.0+	Yes	Yes

2.2 SubtitleFile

The **SubtitleFile** tag is used to denote presentation data files inside a presentation list. At a minimum, a presentation list must contain a single **DCSubtitle** tag followed by one or more **SubtitleFile** tags. XML elements that are allowed in a presentation list include, DCSubtitle, SubtitleFile, SubtitleID, MovieTitle, and Language. Mixing **SubtitleFile** tags and other subtitle tags described in this chapter, with the exception of the tags mentioned in this paragraph, is not allowed. The value of each **SubtitleFile** tag should be an URI (URL) that points to a well formed presentation data file, or another presentation list. Care must be taken to avoid circular references. The URI can be fully qualified, or a relative name. A relative name is relative to the location specified by the server when establishing communications with the projector upon presentation start. Both are shown in the example below. The **SubtitleFile** tag has one optional property, **Offset**. It is specified as an XML attribute and described in the table below. If **Offset** is not specified, then the default offset of 0 is assumed. The time reference used in subtitle files is described in section 2.9.

SubtitleFile properties

Attribute	Value
Offset	A time reference offset that denotes the difference in time between the time reference coming from the server and the time reference in the subtitle file. Offset can be used to "push" the time reference back to allow trailers or other "pre-movie" material to be shown without affecting the time references in the XML files.

Example 2.   SubtitleFile usage

```
<?xml version="1.0" encoding="UTF-8"?>

<DCSubtitle Version="1.1">

     <SubtitleFile Offset="00:04:02:000">
     http://www.mystudio.com/mymovie/opening_credits_reel.xml
     </SubtitleFile>

     <SubtitleFile Offset="00:05:24:000"> ./subtitle_reel1.xml
     </SubtitleFile>

     <SubtitleFile Offset="00:26:02:000"> ./subtitle_reel2.xml
     </SubtitleFile>

     <SubtitleFile Offset="00:46:02:000"> ./subtitle_reel3.xml
     </SubtitleFile>

</DCSubtitle>
```

2.3 SubtitleID

The **SubtitleID** tag is used to denote a Universally Unique Identifier (UUID) for this XML
file. A UUID is a 128-bit number that is guaranteed to be unique worldwide. There are
multiple UUID generation algorithms available from varied sources. Operating systems,
such as Windows, provide functions to generate UUIDs automatically. **SubtitleID** should
be generated by the subtitling authoring system and is represented as a hexadecimal
string as shown in the example below. **SubtitleID** is required in Presentation Data files,
and must precede any Subtitle elements.

Example 3.   SubtitleID usage

```
<?xml version="1.0" encoding="UTF-8"?>

<DCSubtitle Version="1.1">

     <SubtitleID> 40950d85-63eb-4ee2-b1e8-45c126601b94 </SubtitleID>

   ... other XML subtitle contents ...

</DCSubtitle>
```

Subtitle Specification (XML File Format)

2.4 MovieTitle

The **MovieTitle** tag is used to denote the title of the movie that this subtitle file was generated for. **MovieTitle** is required in Presentation Data files, and must precede any Subtitle elements.

Example 4.   MovieTitle usage

```
<?xml version="1.0" encoding="UTF-8"?>

<DCSubtitle Version="1.1">

        <SubtitleID> 40950d85-63eb-4ee2-b1e8-45c126601b94 </SubtitleID>

        <MovieTitle> Bedtime For Bonzo </MovieTitle>

   ... other XML subtitle contents ...

</DCSubtitle>
```

2.5 ReelNumber

The **ReelNumber** tag is used to denote a particular reel number or subelement of a program that a subtitle file corresponds to. **ReelNumber** is required in Presentation Data files, and must precede any Subtitle elements.

Example 5.   ReelNumber usage

```
<?xml version="1.0" encoding="UTF-8"?>

<DCSubtitle Version="1.1">

        <SubtitleID> 40950d85-63eb-4ee2-b1e8-45c126601b94 </SubtitleID>

        <MovieTitle> Bedtime For Bonzo </MovieTitle>

        <ReelNumber> 1 </ReelNumber>

   ... other XML subtitle contents ...

</DCSubtitle>
```

Subtitle Specification (XML File Format)

2.6 Language

The **Language** tag is used to denote the language used in the subtitle file. **Language** is required in Presentation Data files, and must precede any Subtitle elements.

Example 6. Language usage

```
<?xml version="1.0" encoding="UTF-8"?>

<DCSubtitle Version="1.1">

      <SubtitleID> 40950d85-63eb-4ee2-b1e8-45c126601b94 </SubtitleID>

      <MovieTitle> Bedtime For Bonzo </MovieTitle>

      <ReelNumber> 1 </ReelNumber>

      <Language> Dutch </Language>

  ... other XML subtitle contents ...

</DCSubtitle>
```

2.7 LoadFont

The **LoadFont** tag is used to load a TrueType or OpenType font used in the subtitle file from the server. It specifies the font location as an URI (URL) and identifies the font with an ID which is used as the reference in later **Font** tags.   **LoadFont,** if specified, must precede any **Font** element and Subtitle element. If no **LoadFont** element is present in the XML file, then a default font will be used. The **LoadFont** element has only attributes, and no values, so it can be specified as an empty tag as shown below. ***Note:*** In the current version of the spec, only one (1) font can be loaded for a presentation. If more than one font is specified, only the first will be used. In addition, the size of the font must be 640KB or less. Fonts that are larger than 640KB in size may be reduced, see Appendix B for details. These limitations may be increased or eliminated in future releases.

LoadFont properties

Attribute	Value
URI	An URI (URL) that points to the font file. This can be either a fully qualified URI or a relative URI.
Id	Identifier used as a reference in subsequent **Font** elements. This must not be a NULL, or empty string value.

Subtitle Specification (XML File Format)

Example 7. LoadFont usage

```
<?xml version="1.0" encoding="UTF-8"?>

<DCSubtitle Version="1.1">

    <SubtitleID> 40950d85-63eb-4ee2-b1e8-45c126601b94 </SubtitleID>

    <MovieTitle> Bedtime For Bonzo </MovieTitle>

    <ReelNumber> 1 </ReelNumber>

    <Language> Dutch </Language>

    <LoadFont Id="Freds_Font" URI="Fred.ttf" />

  ... other XML subtitle contents ...

</DCSubtitle>
```

2.8 Font

The **Font** tag specifies which font to use for the subtitle, and what properties to use with that font. The different font properties are specified as attributes, and are listed in the table below. The font name is specified using the Id="…" attribute.   The **Font** element can be used anywhere within the Presentation Data file following the **SubtitleID, MovieTitle, ReelNumber,** and **Language** elements. **Font** elements can be nested within each other to temporarily override a particular attribute. Only the overridden attribute needs to be specified, all other attributes will be inherited from the previous **Font** element.

Font properties

Attribute	Value
Id	An Id as used in a previous **LoadFont** element.
Color	Foreground color to render the font. Color is specified as a 32-bit hexadecimal value in an AARRGGBB format. AA = alpha blend, 0xFF is opaque, and 0x00 is transparent. RR = red, GG = green, BB = blue. Default Color = 0xFFFFFFFF (white)
Effect	Special effect used to increase contrast of subtitles on screen. Options: **none, border, shadow** Border draws a colored border around each character. shadow draws a colored drop shadow behind each character. The color is specified in the EffectColor attribute. Default Effect = shadow

Subtitle Specification (XML File Format)

Attribute	Value
EffectColor	Color to use for the effect chosen. Color should be chosen with respect to the foreground color to increase contrast of subtitles. Color is specified as a 32-bit hexadecimal value in an AARRGGBB format. AA = alpha blend, 0xFF is opaque, and 0x00 is transparent. RR = red, GG = green, BB = blue. **<u>Note:</u>** <u>the alpha value of the Color attribute is used for both the foreground color and the effect color. The alpha value of the EffectColor, as described here, may be used in future DLP Cinema™ releases.</u> Default EffectColor = 0xFF000000 (black)
Italic	Used to indicate that the text should be rendered in italics. This is an oblique transform of the original character. For horizontal text it will shear the top edge to the right. For vertical text it will shear the right edge towards the top. Options: **yes, no** Default Italic = no
Script	Used to indicate that the text should be rendered either above or below the baseline as a superscript, or subscript. Options: **normal, super, sub** Default Script = normal
Size	Used to specify the size of the font. Size is given in points. Fonts are rendered as if the screen height is 11 inches, so a 72pt font would be 1/11 screen height. Default Size = 42
AspectAdjust	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. 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

Subtitle Specification (XML File Format)

Attribute	Value
Underlined	Used to indicate that the text should be underlined. **Note:** Underlining is only performed for horizontal text direction. Options: **yes, no** Default Underlined = no
Weight	Used to specify the thickness of the font. Options: **bold, normal** Default Weight = normal
Spacing	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. 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 1em. Negative values are allowed but should be used with care as characters could overlap. Negative spacing shall be limited to no more than -1.0**em**. Default Spacing = **0em**

Example 8.   Font usage

```
<Font Id="Freds_Font" Color="ffffffff"

     Effect="shadow" EffectColor="FF808080" Size="42">

<!-- opaque white text in Freds font, size 42pt, with a medium grey
drop shadow -->

<Subtitle SpotNumber="54" TimeIn="00:12:43:040"
     TimeOut="00:12:50:010" FadeUpTime="20" FadeDownTime="40">

     <Text HAlign="left" HPosition="10.2" VAlign="bottom"
     VPosition="20.0"> This text is normal </Text >

     <Font Italic="yes">

     <Text HAlign="left" HPosition="10.2" VAlign="bottom"
     VPosition="15.0"> This text is italic </Text >

     </Font>

     <Text HAlign="left" HPosition="10.2" VAlign="bottom"
     VPosition="10.0"> This <Font Script="super">word </Font>is
     superscript </Text >

</Subtitle>

</Font>
```

DRAWING NO 2504760	REV C	SHEET 16

Subtitle Specification (XML File Format)

2.9 Subtitle

The **Subtitle** tag denotes the beginning of a particular subtitle. The Subtitle element is a parent element. It includes five distinct properties and one or more child elements, but no other data. Children of the Subtitle element are **Text**, and **Image**. The different subtitle properties are specified as attributes, and are listed in the table below. They include **SpotNumber**, **TimeIn**, **TimeOut**, **FadeUpTime**, and **FadeDownTime**.

Subtitle properties

Attribute	Value
SpotNumber	The spot number as described on the spotting list of the original translation file. This attribute is only present to help identify or locate specific subtitle elements in the finished Presentation Data file. It is not interpreted or used by the projector, its only purpose is to help a human reader.
TimeIn	The time that a particular subtitle should first appear. This is the time that the subtitle first appears on screen, and will fade up over the time specified in the **FadeUpTime** attribute. The time is specified in the format, HH:MM:SS:TTT where HH = hours, MM = minutes, SS = seconds, and TTT = ticks. A "tick" is defined as 4 msec and has a range of 0 to 249. Alternatively, time may be specified in the format HH:MM:SS.sss where HH = hours, MM = minutes, SS = seconds, and sss = decimal fractions of a second. In this format, 01:12:42.5 would indicate 1 hour, 12 minutes, 42 and ½ seconds. This definition of time was chosen because it will allow frame accurate timing at multiple frame rates, without specifying the display frame rate in the subtitle file.
TimeOut	The time that a particular subtitle should disappear. The subtitle will begin to fade down at time (**TimeOut – FadeDownTime**) and will be completely removed by the time specified in the **TimeOut** attribute. The time is specified in the format, HH:MM:SS:TTT where HH = hours, MM = minutes, SS = seconds, and TTT = ticks. A "tick" is defined as 4 msec and has a range of 0 to 249. Alternatively, time may be specified in the format HH:MM:SS.sss where HH = hours, MM = minutes, SS = seconds, and sss = decimal fractions of a second. In this format, 01:12:45.5 would indicate 1 hour, 12 minutes, 45 and ½ seconds. This definition of time was chosen because it will allow frame accurate timing at multiple frame rates, without specifying the display frame rate in the subtitle file.

DRAWING NO	REV	
2504760	C	SHEET 17

Subtitle Specification (XML File Format)

Attribute	Value
FadeUpTime	The time that it takes to bring a subtitle fully on screen. A subtitle begins to appear at the time specified in the **TimeIn** attribute, and fades in over the time specified in this attribute. **FadeUpTime** is in the format TTT, where TTT = ticks. A "tick" is defined as 4 msec and has a range of 0 to 249. For fade times greater than one second, the full time format of HH:MM:SS:TTT or HH:MM:SS.sss must be used. See the attribute description for **TimeIn** or **TimeOut** for the definition of these formats. **FadeUpTime** will be limited to a maximum of 8 seconds. Any time greater than this will be replaced with the max time of 8 seconds. This definition of time was chosen because it will allow frame accurate timing at multiple frame rates, without specifying the display frame rate in the subtitle file. Default FadeUpTime = 20 (this is 80 msec)
FadeDownTime	The time that it takes to remove a subtitle fully from the screen. A subtitle begins to disappear at time (**TimeOut- FadeDownTime**), and fades out over the time specified in this attribute. **FadeDownTime** is in the format TTT, where TTT = ticks. A "tick" is defined as 4 msec and has a range of 0 to 249. For fade times greater than one second, the full time format of HH:MM:SS:TTT or HH:MM:SS.sss must be used. See the attribute description for **TimeIn** or **TimeOut** for the definition of these formats. **FadeDownTime** will be limited to a maximum of 8 seconds. Any time greater than this will be replaced with the max time of 8 seconds. This definition of time was chosen because it will allow frame accurate timing at multiple frame rates, without specifying the display frame rate in the subtitle file. Default FadeDownTime = 20 (this is 80 msec)

Example 9. Subtitle usage

```
<Subtitle SpotNumber="54" TimeIn="00:12:43:040"
     TimeOut="00:12:50:010" FadeUpTime="20" FadeDownTime="40">

<!-- subtitle will appear at 12 minutes, 43 seconds, 40 ticks -->

<!-- subtitle will disappear at 12 minutes, 50 seconds, 10 ticks -->

<!-- fade up over 80 msec time = 2 frames at 24fps -->

<!-- fade down over 160 msec time = 4 frames at 24fps -->


    <Text HAlign="center" VAlign="bottom" VPosition="20.0"> This is
        the time for all good men ... </Text >

</Subtitle>
```

2.10   Text

The **Text** tag denotes the actual subtitle text that is to be rendered on screen. The text string should be short enough to fit on screen using the font specified at the position specified. Multiple lines of text should be handled by multiple **Text** elements. The **Text** element has five attributes, **Direction**, **HAlign**, **HPosition**, **VAlign**, and **VPosition**. These are described in the table below.

Text Position properties

Attribute	Value
Direction	Direction of the rendered text. Valid values are **horizontal**, and **vertical**. Specifying **horizontal** means that the text is rendered horizontally on screen. Specifying **vertical** means that the text is rendered vertically on screen (used for some Asian text). The characters are **not** rotated when **vertical** direction is specified. Default Direction = **horizontal**
HAlign	Horizontal alignment property. Valid values are **left**, **right**, and **center**. Specifying **left** means **HPosition** is relative to the left edge of the picture. Specifying **right** means **HPosition** is relative to the right edge of the picture, and the text will end at the location indicated. Specifying **center** means the text is centered on the picture and **HPosition** is used to shift the center position either to the right with a positive **HPosition**, or left with a negative **HPosition**. Default HAlign = **center**
HPosition	The horizontal position to start, end, or center the text string at. **HPosition** is specified as a percentage of the picture width from the edge or center specified in the **HAlign** attribute. To start the text at a location that is 5% of the picture width from the left edge of the picture, **HAlign**="left" **HPosition**= "5.0". To have the text end at a location that is 8.5% of the picture width from the right edge of the picture, **HAlign**= "right", **HPosition**= "8.5". To have the text centered horizontally on the picture, **HAlign**= "center" **HPosition**= "0" Default HPosition = 0

Subtitle Specification (XML File Format)

Attribute	Value
VAlign	Vertical alignment property. Valid values are **top**, **bottom**, and **center**. Specifying **top** means **VPosition** is relative to the top edge of the picture. Specifying **bottom** means **VPosition** is relative to the bottom edge of the picture. Specifying **center** means the text is centered on the picture and **VPosition** is used to shift the center position either down with a positive **VPosition**, or up with a negative **VPosition**. Default VPosition = **center**
VPosition	The vertical position to start the text string at. **VPosition** is specified as a percentage of the picture height from the edge or center specified in the **VAlign** attribute. To start the text at a location that is 20% of the picture height from the bottom edge of the picture, **VAlign**="bottom" **VPosition** = "20.0". The vertical position specified is the position of the <u>baseline</u> for the characters drawn. Specifying **VAlign**="top" **VPosition**="0" will result in characters being drawn off the top of the screen. Similarly, specifying **VAlign**="bottom" **VPosition**=-"0" will result in the descents of certain characters, like 'g', being drawn off the bottom of the screen. Default VPosition = 0

Example 10. Text usage

```
<Text Direction="horizontal" HAlign="left" HPosition="10.2"
VAlign="bottom" VPosition="20.0"> This is the time for all good men
... </Text >
```

Subtitle Specification (XML File Format)

Example 11. Text horizontal positioning -- graphical representation

Image

Center of Subtitle
Aligned with
Center of Image
(HAlign = center
HPosition = 0)

SUBTITLE

Image

Center of Subtitle
Offset from
Center of Image
(HAlign = center
HPosition = 10)

SUBTITLE

Image

Left Edge of Subtitle
Offset from Left Edge
of Image
(HAlign = left
HPosition = 10)

SUBTITLE

Image

Right Edge of Subtitle
Offset from Right Edge
of Image
(HAlign = right
HPosition = 10)
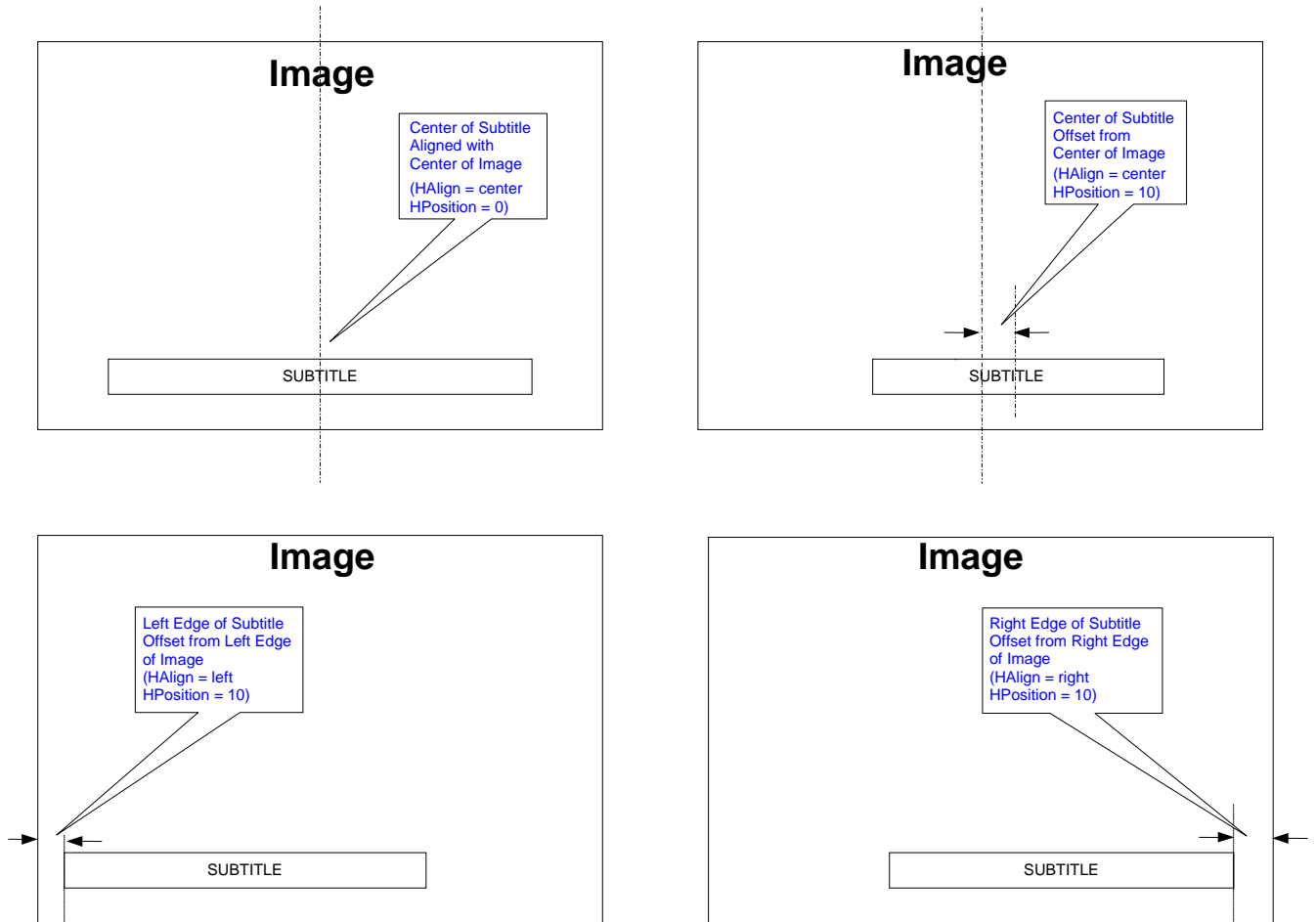
SUBTITLE

Subtitle Specification (XML File Format)
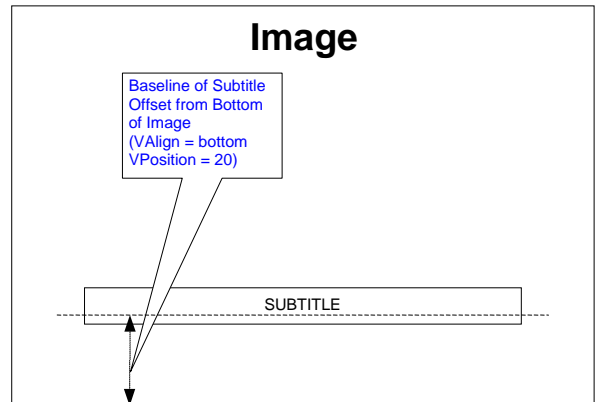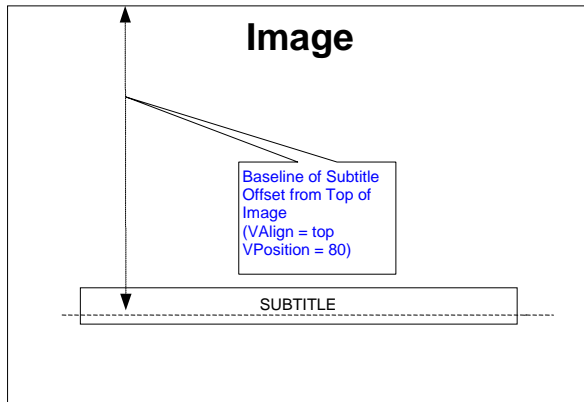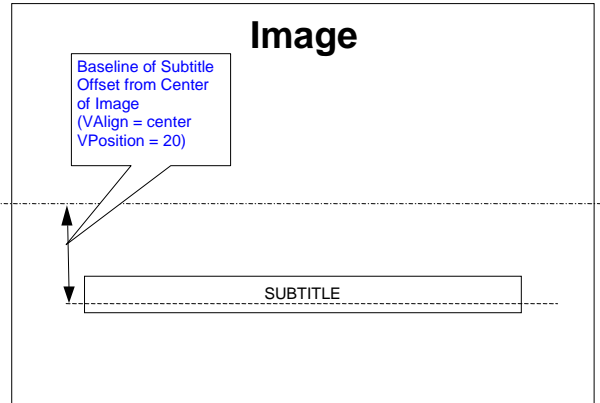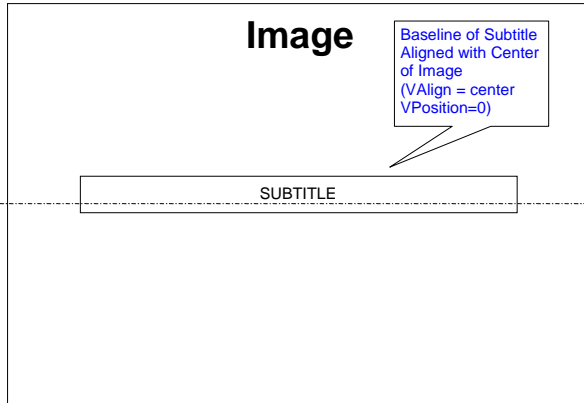
Example 12. Text vertical positioning -- graphical representation

Subtitle Specification (XML File Format)

2.11 Ruby

The **Ruby** tag is new in version 1.1. It has no definition in a version 1.0 subtitle file and any text inside it will be ignored. It is a container tag used to specify ruby characters that are to be associated with a set of base characters in Asian text, specifically Japanese. The **Ruby** element must contain exactly 1 **Rb** element, and 1 **Rt** element. Any other text inside the **Ruby** element will be ignored. The **Ruby** element is only valid inside a **Text** element. **Font** elements are not allowed inside a **Ruby** element. **HGroup** elements are not allowed inside a **Ruby** element

2.12 Rb

The **Rb** or RubyBase tag is new in version 1.1. It has no definition in a version 1.0 subtitle file and any text inside it will be ignored. It is a container tag used to specify the set of base characters that ruby text will be associated with in Asian text, specifically Japanese. The **Rb** tag is only valid inside a **Ruby** element. In this version of the specification, there must be a single **Rb** element inside a **Ruby** element.

2.13 Rt

The **Rt** or RubyText tag is new in version 1.1. It has no definition in a version 1.0 subtitle file and any text inside it will be ignored. It is a container tag 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 is only valid inside a **Ruby** element. In this version of the specification, there must be a single **Rt** element inside a **Ruby** element. The **Rt** element has four attributes, **Size**, **Position**, **Offset**, and **Spacing**. These are described in the table below. The ruby characters will be centered with respect to the base characters in all cases.

Ruby text properties

Attribute	Value
Size	Size of the rendered ruby characters. Character sizes are specified in units of **em**. 1 **em** is equivalent to the current font size for the base characters. 2 **em** is equivalent to twice the size of the current font size, and 0.5 **em** is equivalent to half the size of the current font size. Values for size should be positive numbers (greater than 0). Default Size = **0.5em**

Subtitle Specification (XML File Format)

Attribute	Value
Position	Position of the rendered ruby characters with respect to the base characters. Valid values are **before**, and **after**. Specifying **before** means that the ruby characters are rendered above the base characters if the text direction is horizontal on screen. Specifying **before** means that the ruby characters are rendered to the right of the base characters if the text direction is vertical on screen. Specifying **after** means that the ruby characters are rendered below the base characters if the text direction is horizontal on screen. Specifying **after** means that the ruby characters are rendered to the left of the base characters if the text direction is vertical on screen. Default Position = **before**
Offset	Offset or amount of space between the rendered ruby characters and the base characters. The offset is specified in units of **em**. 1 **em** is equivalent to the current font size for the base characters. 2 **em** is equivalent to twice the size of the current font size, and 0.5 **em** is equivalent to half the size of the current font size. Negative offset shall not exceed -1**em**. Default Offset = **0em**
Spacing	Additional spacing between the rendered ruby characters. The spacing is specified in units of **em**. 1 **em** is equivalent to the current font size for the base characters, and 0.5 **em** is 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 -1.0**em**. **Note:** spacing can be specified here as well as in the **Font** element. The effect of both is cumulative. Default Spacing = **0em**
AspectAdjust	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

Example 13. Ruby usage

Example 1:

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

<Ruby>

      <Rb>雄</Rb>

      <Rt Size="0.5em" Position="before" Offset="0.5em"
      Spacing="0em">おす</Rt>

</Ruby>

</Text>
```

Should produce something like this.

おす

雄

Example 2:

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

カール　君は

<Ruby>

      <Rb>"隷属"</Rb>

      <Rt Size="0.5em" Position ="After" Offset="0.5em">れいぞく</Rt>

</Ruby>

って

</Text>
```

Should produce something like this.

カール　君は "隷属" って

れいぞく

DRAWING NO	REV	
2504760	C	SHEET 25

2.14   Space

The **Space** tag is new in version 1.1. It has no definition in a version 1.0 subtitle file and any text inside it will be ignored. It is 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 defined by using the attribute **Size** described in the table below. The **Space** tag is only valid inside a **Text** element. **Font** elements are not allowed inside a **Space** element. The **Space** element has only attributes, and no values, so it can be specified as an empty tag as shown below.

Space property

Attribute	Value
Size	Size of the space to be inserted. Size is specified in units of **em**. 1 **em** is equivalent to the current font size for the characters being rendered. 2 **em** is equivalent to twice the size of the current font size, and 0.5 **em** is 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 could overlap. Negative spacing shall be limited to no more than -1.0**em**. Default Size = **0.5em**

Example 14. Space usage

```
<Text HAlign="center" HPosition="0.0" VAlign="bottom"
VPosition="10.2">This is some text<Space Size="2.5em"/>with
a 2 and a half character space in the middle.</Text>
```

2.15   HGroup

The **HGroup** tag is new in version 1.1. It has no definition in a version 1.0 subtitle file and any text inside it will be ignored. It is a container tag 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** tag is only valid inside a **Text** element.   The **HGroup** element is not allowed inside a **Ruby** element. The **Ruby** element is not allowed inside an **HGroup** element. **Font** elements are not allowed inside an **HGroup** element.

Example 15. HGroup usage

```
<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.

1963

年は良い年だった。

Subtitle Specification (XML File Format)

2.16   Rotate

The **Rotate** tag is new in version 1.1. It has no definition in a version 1.0 subtitle file and any text inside it will be ignored. It is a container tag 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** tag is only valid inside a **Text** element. **Font** elements are not allowed inside a **Rotate** element. **Rotate** elements are not allowed to be nested inside another **Rotate** element. The **Rotate** element has one attribute, **Direction** which is described in the table below.

Rotate property

Attribute	Value
Direction	Direction of the character rotation. Specifying **none** means that the characters are not rotated. Specifying **right** means that the characters are rotated 90° to the right or clockwise. Specifying **left** means that the characters are rotated 90° to the left or counter-clockwise. Default Direction = **none**

Example 16. Rotate usage

```
<Text Direction="vertical" HAlign="right" HPosition="10.0"
VAlign="top" VPosition="8.25"> 私はドイツ国籍で<Rotate
Direction="right">─ </Rotate></Text>
```

Should produce something like this.

私
は
ド
イ
ツ
国
籍
で
│

DRAWING NO	REV	
2504760	C	SHEET 28

Subtitle Specification (XML File Format)

2.17 Image

The **Image** tag references an image to be displayed on screen instead of a text string. The value specifies the location of the image as an URI (URL). This can be either a fully qualified URI or a relative path. This specification supports the image format Portable Network Graphics (PNG). The image position on screen is defined by using the position specified by the position attributes **HAlign**, **HPosition**, **VAlign**, and **VPosition**. Images will be stored on the server, and retrieved when referenced.

Image Position properties

Attribute	Value
HAlign	Horizontal alignment property. Valid values are **left**, **right**, and **center**. Specifying **left** means **HPosition** is relative to the left edge of the picture. Specifying **right** means **HPosition** is relative to the right edge of the picture. Specifying **center** means the image is centered on the picture and **HPosition** is used to shift the center position either to the right with a positive **HPosition**, or left with a negative **HPosition**. Default HAlign = **center**
HPosition	The horizontal position to start, end, or center the image at. HPosition is specified as a percentage of the picture width from the edge specified in the **HAlign** attribute. To start the image at a location that is 5% of the picture width from the left edge of the picture, **HAlign**="left" **HPosition** = "5.0". To have the image end at a location that is 8.5% of the picture width from the right edge of the picture, **HAlign**= "right", **HPosition**= "8.5". To have the image centered horizontally on the picture, **HAlign**= "center" **HPosition**= "0" Default HPosition = 0
VAlign	Vertical alignment property. Valid values are **top**, **bottom**, and **center**. Specifying **top** means **VPosition** is relative to the top edge of the picture. Specifying **bottom** means **VPosition** is relative to the bottom edge of the picture. Specifying **center** means the image is centered on the picture and **VPosition** is used to shift the center position either down with a positive **VPosition**, or up with a negative **VPosition**. Default VPosition = **center**

DRAWING NO	REV	
2504760	C	SHEET 29

Subtitle Specification (XML File Format)

Attribute	Value
VPosition	The vertical position to place the image at. **VPosition** is specified as a percentage of the picture height from the edge or center specified in the **VAlign** attribute. To place the image at a location that is 5% of the picture height from the top edge of the picture, **VAlign**="top" **VPosition**="5.0". The top edge of the image is used as the reference for **VPosition** if **VAlign**="top". To place the image at a location that is 20% of the picture height from the bottom edge of the picture, **VAlign**="bottom" **VPosition** = "20.0". The bottom edge of the image is used as the reference for **VPosition** if **VAlign**="bottom". If **VAlign**="center", then the center of the image is used as the reference for **VPosition**. Note that this is different than the **VPosition** scheme used for Text elements. Default VPosition = 0

Example 17. Image usage

```
<Image HAlign="right" HPosition="2.5" VAlign="bottom"
VPosition="10.2"> BonzoImage1.png </Image>
```

DRAWING NO	REV	
2504760	C	SHEET 30

Example DLP Cinema™ Subtitle XML File

Note: Indentation is used to enhance readability, and is not required.

```xml
<?xml version="1.0" encoding="UTF-8"?>

<!-- *** XML Subtitle File *** -->

<!-- *** Created By Texas Instruments *** -->

<!-- *** Wednesday, November 13, 2002 *** -->


<DCSubtitle Version="1.0">

        <SubtitleID> 5517935f-7cb2-4f47-a243-7b587b68e32e </SubtitleID>

        <MovieTitle> Julius Ceasar </MovieTitle>

        <ReelNumber> 1 </ReelNumber>

        <Language> English </Language>

        <!-- Set the default font characteristics -->

        <LoadFont Id="theFont" URI="/Font/Helvetica.ttf">  </LoadFont>

        <Font Id="theFont" Color="FFFFFFFF" Effect="shadow" EffectColor="FF000000"
        Size="42">

        <!-- Turn on italics for the following line -->

        <Font Italic="yes">

        <Subtitle SpotNumber="1" TimeIn="00:00:25:219" TimeOut="00:00:30:198"
        FadeUpTime="20" FadeDownTime="20">

                <Text HAlign="center" VAlign="bottom" VPosition="20.00"> Julius
                Ceasar </Text>

        </Subtitle>

        <!-- Turn off italics but leave all other characteristics -->

        </Font>

        <Subtitle SpotNumber="2" TimeIn="00:00:35:219" TimeOut="00:00:37:198"
        FadeUpTime="20" FadeDownTime="20">

                <Text HAlign="center" VAlign="bottom" VPosition="10.00"> Hence!
                Home, you idle creatures get you home. </Text>

        </Subtitle>

        <Subtitle SpotNumber="3" TimeIn="00:00:38:011" TimeOut="00:00:40:011"
        FadeUpTime="20" FadeDownTime="20">

                <Text HAlign="center" VAlign="bottom" VPosition="10.00"> Is this a
                holiday? </Text>
```

DRAWING NO	REV	
2504760	C	SHEET 31

Subtitle Specification (XML File Format)

```
        </Subtitle>

        <Subtitle SpotNumber="4" TimeIn="00:00:41:052" TimeOut="00:00:45:219"
        FadeUpTime="20" FadeDownTime="20">

                <Text HAlign="center" VAlign="bottom" VPosition="20.00"> What! Know
                you not, </Text>

                <Text HAlign="center" VAlign="bottom" VPosition="10.00"> being
                mechanical, you ought not walk </Text>

        </Subtitle>

        <Subtitle SpotNumber="5" TimeIn="00:00:46:031" TimeOut="00:00:48:198"
        FadeUpTime="20" FadeDownTime="20">

                <Text HAlign="center" VAlign="bottom" VPosition="10.00"> upon a
                labouring day without the sign of your profession? </Text>

        </Subtitle>

        <Subtitle SpotNumber="6" TimeIn="00:00:50:011" TimeOut="00:00:52:011"
        FadeUpTime="20" FadeDownTime="20">

                <Text HAlign="center" VAlign="bottom" VPosition="10.00"> Speak,
                what trade art thou?</Text>

        </Subtitle>

        <Subtitle SpotNumber="7" TimeIn="00:00:53:052" TimeOut="00:00:54:219"
        FadeUpTime="20" FadeDownTime="20">

                <Text HAlign="center" VAlign="bottom" VPosition="10.00"> Why, sir,
                a carpenter. </Text>

        </Subtitle>

        <Subtitle SpotNumber="9" TimeIn="00:00:56:094" TimeOut="00:00:58:156"
        FadeUpTime="20" FadeDownTime="20">

                <Text HAlign="center" VAlign="bottom" VPosition="10.00"> Where is
                thy leather apron and thy rule? </Text>

        </Subtitle>



                ... other subtitles removed for brevity ...



        <Subtitle SpotNumber="280" TimeIn="00:20:37:156" TimeOut="00:20:39:219"
        FadeUpTime="20" FadeDownTime="20">

                <Text HAlign="center" VAlign="bottom" VPosition="20.00"> For it is
                after midnight, and ere day </Text>

                <Text HAlign="center" VAlign="bottom" VPosition="10.00"> we will
                awake him and be sure of him.</Text>



        </Subtitle>

        </Font>

    </DCSubtitle>

<!-- *** Created By Texas Instruments *** -->

<!-- *** XML Subtitle File *** -->
```

DRAWING NO	REV	
2504760	C	SHEET 32

Appendix A

DLP Cinema™ Subtitle DTD

The DLP Cinema ^{TM} subtitle XML file should be validated against a document type declaration (DTD). The DTD for this version of the specification is defined below. This DTD is provided by, and maintained by Texas Instruments.

A.1 DTD Specification

```
<!ELEMENT DCSubtitle (SubtitleID, MovieTitle, ReelNumber, Language, LoadFont*,
Font*, Subtitle*)>

  <!ATTLIST DCSubtitle Version CDATA #REQUIRED>

<!ELEMENT SubtitleID (#CDATA)>

<!ELEMENT MovieTitle (#CDATA)>

<!ELEMENT ReelNumber (#CDATA)>

<!ELEMENT Language (#CDATA)>

<!ELEMENT LoadFont EMPTY>

  <!ATTLIST LoadFont Id CDATA #REQUIRED>

  <!ATTLIST LoadFont URI CDATA #REQUIRED>

<!ELEMENT Font (#PCDATA | Font | Subtitle | Text | Image)*>

  <!ATTLIST Font Id CDATA #IMPLIED>

  <!ATTLIST Font Color CDATA  #IMPLIED>

  <!ATTLIST Font Effect CDATA #IMPLIED>

  <!ATTLIST Font EffectColor CDATA #IMPLIED>

  <!ATTLIST Font Italic CDATA #IMPLIED>

  <!ATTLIST Font Script CDATA #IMPLIED>

  <!ATTLIST Font Size CDATA #IMPLIED>

<!ELEMENT Subtitle (Font | Text | Image)*>

  <!ATTLIST Subtitle SpotNumber CDATA #REQUIRED>

  <!ATTLIST Subtitle TimeIn CDATA #REQUIRED>

  <!ATTLIST Subtitle TimeOut CDATA #REQUIRED>

  <!ATTLIST Subtitle FadeUpTime CDATA #IMPLIED>

  <!ATTLIST Subtitle FadeDownTime CDATA #IMPLIED>

<!ELEMENT Text (#PCDATA | Font | Ruby* | Space* | HGroup* | Rotate*)*>

  <!ATTLIST Text Direction CDATA #IMPLIED>

  <!ATTLIST Text HAlign CDATA #IMPLIED>
```

DRAWING NO	REV	
2504760	C	SHEET 33

Subtitle Specification (XML File Format)

```
<!ATTLIST Text HPosition CDATA #IMPLIED>

<!ATTLIST Text VAlign CDATA #IMPLIED>

<!ATTLIST Text VPosition CDATA #IMPLIED>

<!ELEMENT Ruby (Rb, Rt)>

<!ELEMENT Rb (#CDATA)>

<!ELEMENT Rt (#PCDATA)>

<!ATTLIST Rt Size CDATA #IMPLIED>

<!ATTLIST Rt Position CDATA #IMPLIED>

<!ATTLIST Rt Offset CDATA #IMPLIED>

<!ATTLIST Rt Spacing CDATA #IMPLIED>

<!ELEMENT Space EMPTY>

<!ATTLIST Space Size CDATA #IMPLIED>

<!ELEMENT HGroup (#CDATA)>

<!ELEMENT Rotate (#CDATA)>

<!ELEMENT Image (#PCDATA)>

<!ATTLIST Image HAlign CDATA #IMPLIED>

<!ATTLIST Image HPosition CDATA #IMPLIED>

<!ATTLIST Image VAlign CDATA #IMPLIED>

<!ATTLIST Image VPosition CDATA #IMPLIED>
```

DRAWING NO	REV	
2504760	C	SHEET 34

Other Resources

Other tools and resources have been developed to aid in the preparation of XML subtitle files. Some of them are listed here.

B.1 Font Compressor

A font compression tool was needed to reduce the size of TrueType fonts that were larger than the max size indicated in this specification. This compressor works under the following assumptions.

1. The font is too large because it has unnecessary tables such as the embedded bitmap tables, or PCL printer tables, that are not used by the rendering engine.

2. The font is too large because it has a large glyph set, such as Chinese, Japanese, or Korean (CJK) fonts, which typically have 20,000 or more glyphs.

The font compression tool addresses both of these assumptions. If embedded bitmap tables, or printer tables are present in the font, they are removed. The reduction of the number of glyphs is accomplished by first analyzing the subtitle XML file itself. The font compression tool will read the entire subtitle XML file, and record each character that is actually used in the presentation. It will then construct a new font with only the glyphs necessary to render the text for that presentation. This reduced font will only be usable for this presentation, and if the text in the subtitle file were to change, then the font compression tool must be run on the original, large, font again to produce a new reduced font. Using this tool, it is possible to reduce very large fonts to sizes compatible with this specification. As an example, a Chinese font with an original size of 8.5MB was reduced to 274KB for a 90 minute presentation.

This type of font compression tool can be written by any interested party. Information on the tool already written can be obtained from Texas Instruments.

DRAWING NO	REV	
2504760	C	SHEET 35

Subtitle Specification (XML File Format)

B.2 XML Formatting Tool

It is assumed that companies that produce subtitle files will modify their existing tools, or develop new tools that will output their work in the new XML format in the future. In the meantime it was necessary to write a tool that would convert existing subtitle files to the XML format described in this document.

A tool was written that takes two UTF-8 encoded text documents, the first being a spotting list with spot numbers and times, in feet-frames. The second document is a translation file with spot numbers and translated text. The tool matches the spot numbers from the translation file with the same spot number in the spotting list to find the correct TimeIn and TimeOut parameters, and produces an XML file in the format described in this document.

Information on the tool already written can be obtained from Texas Instruments.

B.3 Server Simulator

It is assumed that companies that produce Digital Cinema Servers will modify their existing server software to support digital subtitling. This includes running an HTTP server, and communicating with the projector through a set of defined server commands. The server commands necessary to support digital subtitling are described in the DLP Cinema™ server interface specification. In the meantime it was necessary to write a tool that would act as a server simulator. This tool can be run on any PC that can be connected to the projector via Ethernet. Effective demonstrations have been run using a laptop running Windows2000. The server simulator provides an interface, and the means to communicate with the projector subtitling system.

Information on this tool can be obtained from Texas Instruments.