

Technical Specification Group Services and System Aspects **TSGS#16(02)0226**
Meeting #16, Marco Island, Florida, USA, 10-13 June 2002

Source: TSG-SA WG4

Title: CRs to TS 26.234 on Corrections Based on Interoperability Issues, Mime media type update, sample Description atom and Timed Text Format (Release 5)

Document for: Approval

Agenda Item: 7.4.3

The following CRs, agreed at the TSG-SA WG4 meeting #21, are presented to TSG SA #16 for approval.

Spec	CR	Rev	Phase	Subject	Cat	Vers	WG	Meeting	S4 doc
26.234	024	1	REL-5	Correction to Timed Text	F	5.0.0	S4	TSG-SA WG4#21	S4-020363
26.234	026	3	REL-5	Mime media type update	A	5.0.0	S4	TSG-SA WG4#21	S4-020349
26.234	027		REL-5	Corrections to the description of Sample Description atom and Timed Text Format	F	5.0.0	S4	TSG-SA WG4#21	S4-020308
26.234	029	1	REL-5	Corrections Based on Interoperability Issues	A	5.0.0	S4	TSG-SA WG4#21	S4-020351

CR-Form-v5

CHANGE REQUEST

⌘ **26.234 CR 024** ⌘ rev **1** ⌘ Current version: **5.0.0** ⌘

For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.

Proposed change affects: ⌘ (U)SIM ME/UE Radio Access Network Core Network

Title:	⌘ Corrections to Timed Text		
Source:	⌘ TSG SA WG4		
Work item code:	⌘ PSS-E	Date:	⌘ 11 June, 2002
Category:	⌘ F	Release:	⌘ REL-5
	<i>Use one of the following categories:</i> F (correction) A (corresponds to a correction in an earlier release) B (addition of feature), C (functional modification of feature) D (editorial modification) Detailed explanations of the above categories can be found in 3GPP TR 21.900.		<i>Use one of the following releases:</i> 2 (GSM Phase 2) R96 (Release 1996) R97 (Release 1997) R98 (Release 1998) R99 (Release 1999) REL-4 (Release 4) REL-5 (Release 5)

Reason for change:	⌘ <ol style="list-style-type: none"> 1) The definition of the character count for non-printing characters is unclear. 2) The definition in 'ftab' is unclear. 3) The handling of null termination of text string is unclear. 4) The definition for the case of startcharoffset=endcharoffset is unclear. 5) The definition of continuousKaraoke flag is unclear. 6) Some of the combinational use of sample modifiers is unclear.
Summary of change:	⌘ <ol style="list-style-type: none"> 1) The definition of the character count for non-printing characters is clarified. 2) The definition in 'ftab' is clarified. 3) The handling of null termination of text string is clarified 4) The definition for the case of startcharoffset=endcharoffset is added. 5) The definition of continuousKaraoke flag is clarified. 6) Combinational use of sample modifiers is clarified, and confusing combination of the modifiers is prohibited.
Consequences if not approved:	⌘ Misunderstanding of the specification may cause interoperability problem.

Clauses affected:	⌘ D.8a.15, D.8a.16, D.8a.17, D.8a.17.1.3, D.8a.18		
Other specs affected:	⌘ <input type="checkbox"/> Other core specifications <input type="checkbox"/> Test specifications <input type="checkbox"/> O&M Specifications	⌘	
Other comments:	⌘		

How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at: http://www.3gpp.org/3G_Specs/CRs.htm. Below is a brief summary:

- 1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

- 2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under <ftp://ftp.3gpp.org/specs/> For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.
- 3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

D.8a.15 Style record

Both the sample format and the sample description contain style records, and so it is defined once here for compactness.

```
aligned(8) class StyleRecord {
    unsigned int(16)    startChar;
    unsigned int(16)    endChar;
    unsigned int(16)    font-ID;
    unsigned int(8)    face-style-flags;
    unsigned int(8)    font-size;
    unsigned int(8)    text-color-rgba[4];
}
```

startChar:	character offset of the beginning of this style run (always 0 in a sample description)
endChar:	first character offset to which this style does not apply (always 0 in a sample description); shall be greater than or equal to startChar. <u>All characters, including line-break characters and any other non-printing characters, are included in the character counts.</u>
font-ID:	font identifier from the font table; in a sample description, this is the default font
face style flags:	in the absence of any bits set, the text is plain <ul style="list-style-type: none"> 1 bold 2 italic 4 underline
font-size:	font size (nominal pixel size, in essentially the same units as the width and height)
text-color-rgba:	rgb colour, 8 bits each of red, green, blue, and an alpha (transparency) value

Terminals shall support plain text, and underlined horizontal text, and may support bold, italic and bold-italic depending on their capabilities and the font selected. If a style is not supported, the text shall still be rendered in the closest style available.

...

D.8a.16 Sample Description Format

The sample table box ('stbl') contains sample descriptions for the text track. Each entry is a sample entry box of type 'tx3g'. This name defines the format both of the sample description and the samples associated with that sample description. Terminals shall not attempt to decode or display sample descriptions with unrecognised names, nor the samples attached to those sample descriptions.

It starts with the standard fields (the reserved bytes and the data reference index), and then some text-specific fields. Some fields can be overridden or supplemented by additional boxes within the text sample itself. These are discussed below.

There can be multiple text sample descriptions in the sample table. If the overall text characteristics do not change from one sample to the next, the same sample description is used. Otherwise, a new sample description is added to the table. Not all changes to text characteristics require a new sample description, however. Some characteristics, such as font size, can be overridden on a character-by-character basis. Some, such as dynamic highlighting, are not part of the text sample description and can be changed dynamically.

The TextDescription extends the regular sample entry with the following fields.

```
class FontRecord {
    unsigned int(16)    font-ID;
    unsigned int(8)    font-name-length;
    unsigned int(8)    font[font-name-length];
}

class FontTableBox() extends Box('ftab') {
    unsigned int(16)    entry-count;
    FontRecord    font-entry[entry-count];
}
```

```

}

class BoxRecord {
    signed int(16) top;
    signed int(16) left;
    signed int(16) bottom;
    signed int(16) right;
}

class TextSampleEntry() extends SampleEntry ('tx3g') {
    unsigned int(32) displayFlags;
    signed int(8) horizontal-justification;
    signed int(8) vertical-justification;
    unsigned int(8) background-color-rgba[4];
    BoxRecord default-text-box;
    StyleRecord default-style;
    FontTableBox font-table;
}

```

displayFlags:

scroll In	0x00000020	
scroll Out	0x00000040	
scroll direction	0x00000180	/ see above for values
continuous karaoke	0x00000800	
write text vertically	0x00020000	

horizontal and vertical justification: / two eight-bit values from the following list:

left, top	0
centered	1

bottom, right -1

background-color-rgba:

rgb color, 8 bits each of red, green, blue, and an alpha (transparency) value

Default text box: the default text box is set by four values, relative to the text region; it may be over-ridden in samples;

style record of default style: startChar and endChar shall be zero in a sample description

The text box is inset within the region defined by the track translation offset, width, and height. The values in the box are relative to the track region, and are uniformly coded with respect to the pixel grid. So, for example, the default text box for a track at the top left of the track region and 50 pixels high and 100 pixels high is {0, 0, 50, 100}.

A font table shall follow these fields, to define the complete set of fonts used. The font table is an atom of type 'ftab'. Every font used in the samples is defined here by name. Each entry consists of a 16-bit local font identifier, and a font name, expressed as a string, preceded by an 8-bit field giving the length of the string in bytes. The name is expressed in UTF-8 characters, unless preceded by a UTF-16 byte-order-mark, whereupon the rest of the string is in 16-bit Unicode characters. The string should be a comma separated list of font names to be used as alternative font, in preference order. The special names "Serif", "Sans-serif" and "Monospace" may be used. The terminal should use the first font in the list which it can support; if it cannot support any for a given character, but it has a font which can, it should use that font. Note that this substitution is technically character by character, but terminals are encouraged to keep runs of characters in a consistent font where possible.

...

D.8a.17 Sample Format

Each sample in the media data consists of a string of text, optionally followed by sample modifier boxes.

For example, if one word in the sample has a different size than the others, a 'styl' box is appended to that sample, specifying a new text style for those characters, and for the remaining characters in the sample. This overrides the style in the sample description. These boxes are present only if they are needed. If all text conforms to the sample description, and no characteristics are applied that the sample description does not cover, no boxes are inserted into the sample data.

```

class TextSampleModifierBox(type) extends Box(type) {
}

class TextSample {
    unsigned int(16)    text-length;
    unsigned int(8)    text[text-length];
    TextSampleModifierBox text-modifier[]; // to end of the sample
}

```

The initial string is preceded by a 16-bit count of the number of bytes in the string. There is no need for null termination of the text string. The sample size table provides the complete byte-count of each sample, including the trailing modifier boxes; by comparing the string length and the sample size, you can determine how much space, if any, is left for modifier boxes.

Authors should limit the string in each text sample to not more than 2048 bytes, for maximum terminal interoperability.

Any unrecognised box found in the text sample should be skipped and ignored, and processing continue as if it were not there.

...

D.8a.17.1.3 Dynamic Highlight

'krok' – Karaoke, closed caption, or dynamic highlighting. The number of highlight events is specified, and each event is specified by a starting and ending character offset and an end time for the event. The start time is either the sample start time or the end time of the previous event. The specified characters are highlighted from the previous end-time (initially the beginning of this sample's time), to the end time. The times are all specified relative to the sample's time; that is, a time of 0 represents the beginning of the sample time. The times are measured in the timescale of the track.

The atom starts with the start-time offset of the first highlight event, a 16-bit count of the event count, and then that number of 8-byte records. Each record contains the end-time offset as a 32-bit number, and the text start and end values, each as a 16-bit number. These values are specified as in the highlight record – the offset of the first character to highlight, and the offset of the first character not highlighted. The special case, where the startcharoffset equals to the endcharoffset, can be used to pause during or at the beginning of dynamic highlighting. The records shall be ordered and not overlap, as in the highlight record. The time in each record is the end time of this highlight event; the first highlight event starts at the indicated start-time offset from the start time of the sample. The time values are in the units expressed by the timescale of the track. The time values shall not exceed the duration of the sample.

The continuouskaraoke flag controls whether to highlight only those characters (continuouskaraoke = 0) selected by a karaoke entry, or the entire string from the beginning up to the characters highlighted (continuouskaraoke = 1) at any given time. In other words, the flag specifies whether karaoke should ignore the starting offset and highlight all text from the beginning of the sample to the ending offset.

Karaoke highlighting is usually achieved by using the highlight colour as the text colour, without changing the background.

At most one dynamic highlight ('krok') atom may occur in a sample.

```

class TextKaraokeBox() extends TextSampleModifierBox ('krok') {
    unsigned int(32)    highlight-start-time;
    unsigned int(16)    entry-count;
    for (i=1; i<=entry-count; i++) {
        unsigned int(32)    highlight-end-time;
        unsigned int(16)    startcharoffset;
        unsigned int(16)    endcharoffset;
    }
}

```

...

D.8a.18 Combinations of features

Two modifier boxes of the same type shall not be applied to the same character (e.g. it is not permitted to have two href links from the same text). As the 'hclr', 'dlay' and 'tbox' are globally applied to the whole text in a sample, two modifier boxes of the same type shall not be present within a sample.

Table D.8 details the effects of multiple options:

Table D.8: Combinations of features

		Sample description style record	First sample modifier atom				
			styl	hlit	krok	href	blnk
Second sample modifier atom	styl	1	3				
	hlit			3			
	krok			4	3		
	href	2	2		5	3	
	blnk		6	6	6	6	6

1. The sample description provides the default style; the style records over-ride this for the selected characters.
2. The terminal over-rides the chosen style for HREF links.
3. Two records of the same type cannot be applied to the same character.
4. ~~The characters specified by the highlight record are highlighted all the time; the Karaoke highlighting is applied at the selected times. This may be visually confusing and is not recommended.~~ Dynamic and static highlighting must not be applied to the same text.
5. Dynamic highlighting and linking must not be applied to the same text.
6. Blinking text is optional, particularly when requested in combination with other features.

CR-Form-v5

CHANGE REQUEST

⌘ **26.234 CR 26** ⌘ rev **3** ⌘ Current version: **5.0.0** ⌘

For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.

Proposed change affects: ⌘ (U)SIM ME/UE Radio Access Network Core Network

Title:	⌘ Mime media type update		
Source:	⌘ TSG SA WG4		
Work item code:	⌘ PSTREAM	Date:	⌘
Category:	⌘ A	Release:	⌘ REL-5
	Use <u>one</u> of the following categories:		Use <u>one</u> of the following releases:
	F (correction)		2 (GSM Phase 2)
	A (corresponds to a correction in an earlier release)		R96 (Release 1996)
	B (addition of feature),		R97 (Release 1997)
	C (functional modification of feature)		R98 (Release 1998)
	D (editorial modification)		R99 (Release 1999)
	Detailed explanations of the above categories can be found in 3GPP TR 21.900 .		REL-4 (Release 4)
			REL-5 (Release 5)

Reason for change:	<p>⌘ It is possible to send audio configuration for MPEG-4 audio inband (in the RTP payload) or outband (in the SDP) but not both. If the configuration information is send inband there is a risk that the information get lost and thus making it impossible to decode the stream.</p> <p>MPEG4 video configuration information is transported inband and outband. However the RFC 3016 does not clearly express this. The information from RFC 3016 thus needs to be repeated in 26.234, and it is explicitly mandated to transport MPEG4 configuration information outband (in the SDP) and inband (in the RTP payload).</p> <p>It is not necessary to make this change for AMR, AMR-WB and H.263, even if the configuration information is carried inband, since those codecs regularly (on frame basis) repeats the configuration information.</p>
Summary of change:	<p>⌘ Configuration information is mandated to be send in the SDP (which is transported over TCP and thus safely delivered to the terminal) for MPEG4 audio and MPEG4 video.</p> <p>MPEG4 video configuration information that is resent in the RTP stream (as part of the payload) shall be the same as in the SDP (with exceptions of two field that may vary if present).</p>
Consequences if not approved:	<p>⌘ Configuration information transported on an unreliable channel (RTP) may get lost. If lost the decoder in the terminal will not be able to decode the content correctly making the service unusable.</p>

Clauses affected:	⌘ 5.4
Other specs	⌘ <input type="checkbox"/> Other core specifications ⌘

affected:	<input type="checkbox"/>	Test specifications	
	<input type="checkbox"/>	O&M Specifications	
Other comments:	⌘		

How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at: http://www.3gpp.org/3G_Specs/CRs.htm. Below is a brief summary:

- 1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.
- 2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under <ftp://ftp.3gpp.org/specs/> For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.
- 3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

5.4 MIME media types

For continuous media (speech, audio and video) the following MIME media types shall be used:

- AMR narrow-band speech codec (see clause 7.2) MIME media type as defined in [11];
- AMR wideband speech codec (see clause 7.2) MIME media type as defined in [11];
- - MPEG-4 AAC audio codec (see clause 7.3) MIME media type as defined in RFC 3016 [13]. When used in SDP the attribute "cpresent" SHALL be set to "0" indicating that the configuration information is only carried out of band in the SDP "config" parameter.
- - MPEG-4 video codec (see clause 7.4) MIME media type as defined in RFC 3016 [13]. When used in SDP the configuration information shall be carried outband in the "config" SDP parameter and inband (as stated in RFC 3016). As described in RFC 3016, the configuration information sent inband and the config information in the SDP shall be the same except that first half vbv occupancy and latter half vbv occupancy which, if exist, may vary in the configuration information sent inband.
- - H.263 [22] video codec (see clause 7.4) MIME media type as defined in annex C, clause C.1 of the present document.

MIME media types for JPEG, GIF, PNG, SP-MIDI, SVG, timed text and XHTML can be used both in the "Content-type" field in HTTP and in the "type" attribute in SMIL 2.0. The following MIME media types shall be used for these media:

- JPEG (see clause 7.5) MIME media type as defined in [15];
- GIF (see clause 7.6) MIME media type as defined in [15];
- PNG (see sub clause 7.6) MIME media type as defined in [38];
- SP-MIDI (see sub clause 7.3A) MIME media type as defined in clause C.2 in Annex C of the present document;
- SVG (see sub clause 7.7) MIME media type as defined in [42];
- XHTML (see clause 7.8) MIME media type as defined in [16].
- Timed text (see subclause 7.9) MIME media type as defined in clause D.9 in Annex D of the present document.

MIME media type used for SMIL files shall be according to [31] and for SDP files according to [6].

CHANGE REQUEST

⌘ **TS 26.234 CR 027** ⌘ rev **-** ⌘ Current version: **5.0.0** ⌘

For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.

Proposed change affects: ⌘ (U)SIM ME/UE Radio Access Network Core Network

Title: ⌘ Corrections to the description of Sample Description atom and Timed Text Format.

Source: ⌘ TSG SA WG4

Work item code: ⌘ PSTREAM

Date: ⌘ 11 June 2002

Category: ⌘ **F**

Release: ⌘ REL-5

Use one of the following categories:

Use one of the following releases:

F (correction)

2 (GSM Phase 2)

A (corresponds to a correction in an earlier release)

R96 (Release 1996)

B (addition of feature),

R97 (Release 1997)

C (functional modification of feature)

R98 (Release 1998)

D (editorial modification)

R99 (Release 1999)

Detailed explanations of the above categories can be found in 3GPP [TR 21.900](#).

REL-4 (Release 4)

REL-5 (Release 5)

Reason for change: ⌘

- 1) The description of Sample Description atom is not correct.
- 2) The syntax of TrackHeaderBox is not for text track.
- 3) The scroll direction is described by hexadecimal notation.
- 4) The description of Sample Description Format is not correct.
- 5) The syntax of 'krok' is not complete.

Summary of change: ⌘

- 1) The CR corrects the description of Sample Description atom.
- 2) The CR corrects the syntax of TrackHeaderBox for text track.
- 3) The CR corrects hexadecimal notation to binary notation.
- 4) The CR corrects the description of Sample Description Format.
- 5) The CR corrects the syntax of 'krok'.

Consequences if not approved: ⌘ There could be a critical interoperability problem in 3GPP based MP4 files.

Clauses affected: ⌘ D.2, D.8a.7, D.8a.8, D.8a.16 and D.8a.17.1.3

Other specs Affected: ⌘ Other core specifications ⌘
 Test specifications
 O&M Specifications

Other comments: ⌘

How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at: http://www.3gpp.org/3G_Specs/CRs.htm. Below is a brief summary:

- 1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.
- 2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under <ftp://ftp.3gpp.org/specs/>. For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.
- 3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

D.2 Sample Description atom

...

The format of SampleDescriptionEntry and its fields are explained as follows:

```
SampleDescriptionEntry ::= VisualSampleEntry |
                          AudioSampleEntry |
                          HintSampleEntry |
                          MpegSampleEntry
                          H263SampleEntry |
                          AMRSampleEntry
```

Table D.1: SampleDescriptionEntry fields

Field	Type	Details	Value
VisualSampleEntry		Entry type for visual samples defined in the MPEG-4 specification.	
AudioSampleEntry		Entry type for audio samples defined in the MPEG-4 specification.	
HintSampleEntry		Entry type for hint track samples defined in the MPEG-4 specification.	
MpegSampleEntry		Entry type for MPEG related stream samples defined in the MPEG-4 specification.	
H263SampleEntry		Entry type for H.263 visual samples defined in clause D.6 of the present document.	
AMRSampleEntry		Entry type for AMR and AMR-WB speech samples defined in clause D.5 of the present document.	

From the above 56 atoms, only the VisualSampleEntry, AudioSampleEntry, H263SampleEntry and AMRSampleEntry atoms are taken into consideration, since MPEG specific streams and hint tracks are out of the scope of the present document.

D.8a.7 Text rendering position and composition

Text is rendered within a region (a concept derived from SMIL). There is a text box set within that region. This permits the terminal to position the text within the overall presentation, and also to render the text appropriately given the writing direction. For text written left to right, for example, the first character would be rendered at, or near, the left edge of the box, and with its baseline down from the top of the box by one baseline height (a value derived from the font and font size chosen). Similar considerations apply to the other writing directions.

Within the region, text is rendered within a text box. There is a default text box set, which can be over-ridden by a sample.

The text box is filled with the background colour; after that the text is painted in the text colour. If highlighting is requested one or both of these colours may vary.

Terminals may choose to anti-alias their text, or not.

The text region and layering are defined using structures from the ISO base media file format.

This track header box is used for text track:

```
aligned(8) class TrackHeaderBox
  extends FullBox('tkhd', version, flags){
```

```

if (version==1) {
    unsigned int(64)    creation_time;
    unsigned int(64)    modification_time;
    unsigned int(32)    track_ID;
    const unsigned int(32) reserved = 0;
    unsigned int(64)    duration;
} else { // version==0
    unsigned int(32)    creation_time;
    unsigned int(32)    modification_time;
    unsigned int(32)    track_ID;
    const unsigned int(32) reserved = 0;
    unsigned int(32)    duration;
}
const unsigned int(32)[2] reserved = 0;
template int(16) layer=0;
template int(16) alternate_group = 0;
template int(16) volume = {if track_is_audio 0x0100 else 0};
const unsigned int(16) reserved = 0;
template int(32)[9] matrix=
{ 0x00010000,0,0,0,0x00010000,0,0,tx,ty,0x40000000 };
// unity matrix
template unsigned int(32) width=;
{if track_is_visual 0x01400000 else 0};
template unsigned int(32) height=;
{if track_is_visual 0x00F00000 else 0};
}

```

Visually composed tracks including video and text are layered using the ‘layer’ value. This compares, for example, to z-index in SMIL. More negative layer values are towards the viewer. (This definition is compatible with that in ISO/MJ2).

The region is defined by the track width and height, and translation offset. This corresponds to the SMIL region. The width and height are stored in the track header fields above. The sample description sets a text box within the region, which can be over-riden by the samples.

The translation values are stored in the track header matrix in the following positions:

```
{ 0x00010000,0,0, 0,0x00010000,0, tx, ty, 0x40000000 }
```

These values are fixed-point 16.16 values, here restricted to be integers (the lower 16 bits of each value shall be zero). The X axis increases from left to right; the Y axis from top to bottom. (This use of the matrix is conformant with ISO/MJ2.)

So, for example, a centered region of size 200x20, positioned below a video of size 320x240, would have track_width set to 200 (width= 0x00c80000), track_height set to 20 (height= 0x00140000), and tx = (320-200)/2 = 60, and ty=240.

Since matrices are not used on the video tracks, all video tracks are set at the coordinate origin. Figure D.2 provides an overview:

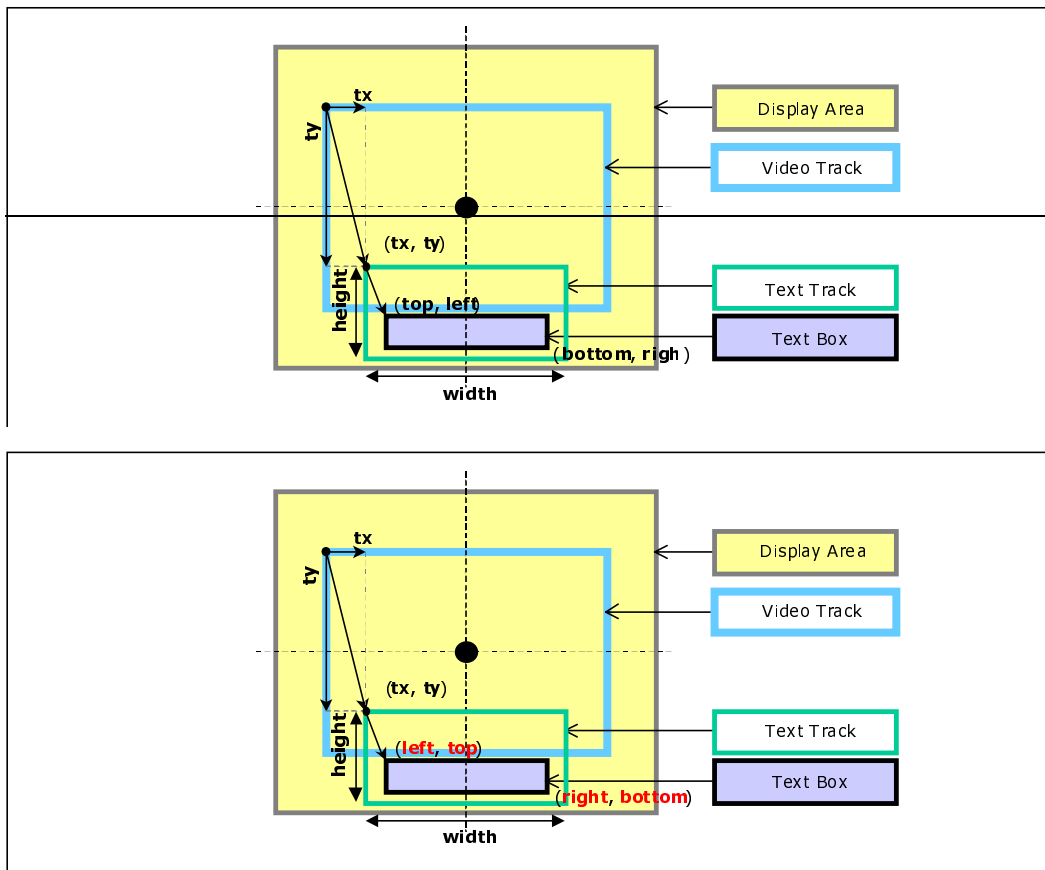


Figure D.2: Illustration of text rendering position and composition

The top and left positions of the text track is determined by the tx and ty, which are the translation values from the coordinate origin (since the video track is at the origin, this is also the offset from the video track). The default text box set in the sample description sets the rendering area unless over-riden by a 'tbox' in the text sample. The box values are defined as the relative values from the top and left positions of the text track.

It should be noted that this only specifies the relationship of the tracks within a single 3GP (MP4) file. If a SMIL presentation lays up multiple files, their relative position is set by the SMIL regions. Each file is assigned to a region, and then within those regions the spatial relationship of the tracks is defined.

D.8a.8 Marquee Scrolling

...

The scrolling direction is set by a two-bit field, with the following possible values:

- 0x00b – text is vertically scrolled up ('credits style'), entering from the bottom of the bottom and leaving towards the top.
- 0x01b – text is horizontally scrolled ('marquee style'), entering from the right and leaving towards the left.
- 0x10b – text is vertically scrolled down, entering from the top and leaving towards the bottom.
- 0x11b – text is horizontally scrolled, entering from the left and leaving towards the right.

...

D.8a.16 Sample Description Format

...

The text box is inset within the region defined by the track translation offset, width, and height. The values in the box are relative to the track region, and are uniformly coded with respect to the pixel grid. So, for example, the default text box for a track at the top left of the track region and 50 pixels high and 100 pixels ~~high~~ wide is {0, 0, 50, 100}.

A font table shall follow these fields, to define the complete set of fonts used. The font table is an atom of type 'ftab'. Every font used in the samples is defined here by name. Each entry consists of a 16-bit local font identifier, and a font name, expressed as a string, preceded by an 8-bit field giving the length of the string in bytes. The name is expressed in UTF-8 characters, unless preceded by a UTF-16 byte-order-mark, whereupon the rest of the string is in 16-bit Unicode characters. The string should be a series of font names, in preference order. The special names "Serif", "Sans-serif" and "Monospace" may be used. The terminal should use the first font in the list which it can support; if it cannot support any for a given character, but it has a font which can, it should use that font. Note that this substitution is technically character by character, but terminals are encouraged to keep runs of characters in a consistent font where possible.

...

D.8a.17.1.3 Dynamic Highlight

...

At most one dynamic highlight ('krok') atom may occur in a sample.

```
class TextKaraokeBox() extends TextSampleModifierBox ('krok') {
    unsigned int(32)    highlight-start-time;
    unsigned int(16)    entry-count;
    for (i=1; i<=entry-count; i++) {
        unsigned int(32)    highlight-end-time;
        unsigned int(16)    startcharoffset;
        unsigned int(16)    endcharoffset;
    }
}
```


3GPP TSG-SA WG4 Meeting #21
May 13-17, 2002, Rennes, France

Tdoc S4-020351

CR-Form-v5
CHANGE REQUEST
⌘ 26.234 CR 029 ⌘ rev 1 ⌘ Current version: 5.0.0 ⌘

For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.

Proposed change affects: ⌘ (U)SIM ME/UE Radio Access Network Core Network

Title:	⌘ Corrections based on interoperability issues
Source:	⌘ TSG SA WG4
Work item code:	⌘ PSTREAM Date: ⌘ 11 June 2002
Category:	⌘ A Release: ⌘ REL-5 <i>Use one of the following categories:</i> F (correction) A (corresponds to a correction in an earlier release) B (addition of feature), C (functional modification of feature) D (editorial modification) Detailed explanations of the above categories can be found in 3GPP TR 21.900 .
	<i>Use one of the following releases:</i> 2 (GSM Phase 2) R96 (Release 1996) R97 (Release 1997) R98 (Release 1998) R99 (Release 1999) REL-4 (Release 4) REL-5 (Release 5)

Reason for change:	⌘ Three issues were found to cause interoperability problems in IMTC PSS-AG testing.
Summary of change:	⌘ (a) A PSS Server shall include the Range header field in all PLAY responses. (b) A PSS Server offering AAC streams shall use the MIME parameters defined in rfc3016 to signal 'profile-level-id' and 'object' type in SDP. (c) Each AMR or AMR-WB track in a 3GPP/MP4 file shall be limited to referencing a single AMRSampleEntry.
Consequences if not approved:	⌘ (a) A PSS client which does not receive the Range header might experience synchronization problems or display clip position incorrectly after a seek. (b) Different interpretations of how these MIME parameters should be used will lead to inconsistent signalling by PSS servers; PSS clients will find it difficult to determine whether they can support an AAC stream, given the information in SDP. (c) Without clarification, some implementers will assume multiple AMRSampleEntries are allowed, and some will not. This will lead to interoperability problems between file authors and file readers.

Clauses affected:	⌘ 5.3.2, 7.3, D.5
Other specs affected:	⌘ <input type="checkbox"/> Other core specifications ⌘ <input type="checkbox"/> <input type="checkbox"/> Test specifications ⌘ <input type="checkbox"/> <input type="checkbox"/> O&M Specifications ⌘ <input type="checkbox"/>
Other comments:	⌘ <input type="text"/>

How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at: http://www.3gpp.org/3G_Specs/CRs.htm. Below is a brief summary:

- 1) Fill out the above form. The symbols above marked ☞ contain pop-up help information about the field that they are closest to.
- 2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under <ftp://ftp.3gpp.org/specs/> For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.
- 3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

...

5.3.2 RTSP

RTSP [5] shall be used for session set-up and session control. PSS clients and servers shall follow the rules for minimal on-demand playback RTSP implementations in appendix D of [5]. In addition to this:

- PSS servers and clients shall implement the DESCRIBE method (see clause 10.2 in [5]);
- PSS servers and clients shall implement the Range header field (see clause 12.29 in [5]).

- PSS servers shall include the Range header field in all PLAY responses.

5.3.3 SDP

...

7.3 Audio

MPEG-4 AAC Low Complexity (**AAC-LC**) object type decoder [21] should be supported. The maximum sampling rate to be supported by the decoder is 48 kHz. The channel configurations to be supported are mono (1/0) and stereo (2/0). In addition, the MPEG-4 AAC Long Term Prediction (**AAC-LTP**) object type decoder may be supported.

When a server offers an AAC-LC or AAC-LTP stream with the specified restrictions, it shall include the "profile-level-id" and "object" MIME parameters in the SDP "a=fmtp" line. The following values shall be used:

Object Type	profile-level-id	object
AAC-LC	15	2
AAC-LTP	15	4

...

D.5 AMRSampleEntry atom

For narrow-band AMR, the atom type of the AMRSampleEntry Atom shall be 'samr'. For AMR wideband (AMR-WB), the atom type of the AMRSampleEntry Atom shall be 'sawb'. Each AMR or AMR-WB track shall be associated with a single AMRSampleEntry.

The AMRSampleEntry Atom is defined as follows:

...