Technical Specification Group Services and System Aspects *TSGS#11(01)0094*
Meeting #11, Palm Springs, U.S.A, 19-22 March 2001 Agenda Item: 7.4.3

Source:	**TSG_SA WG4**
Title:	**Transparent end-to-end packet switched streaming services (PSS); Protocols and codecs (Release 4)**
Document for:	**Approval**

Presentation of Specification to TSG-SA

Presentation to:	**TSG SA Meeting #11**
Document for presentation:	**TS 26.234 Version 2.0.0**
Presented for:	**Approval**

Abstract of document:

The present document specifies the protocols and codecs for the PSS within the 3GPP system. Protocols for control signalling, scene description, media transport and media encapsulations are specified. Codecs for speech, audio, video, still images, bitmap graphics, and text are specified.

The present document is applicable to IP based packet switched networks.

Changes since last presentation to TSG Meeting #10:

Restructuring of layout, formatted accordingly to 3GPP drafting rules, clarified where to find MIME types, references updated.

Added Range header in RTSP subclause, Additional information about audio codecs, clarification on SMIL, interworking with MMS, WB speech codec.

Minor changes after review by the PSM group

Minor changes after review by SA#15 plenary

Decisions on SMIL Profile at the PSM streaming Adhoc added

References to MP4 and MMS added, figure 1 redrawn, new figure 2, text on capability exchange added, MIME-type H263-2000 added, text about MP4 added in clause 9, text about RTSP added in Annex A, proposal for text in Annex B and C added

Text in clause 5.1 added, more mandatory SDP fields added in 5.3.3, 5.3.4 removed and replaced with 5.4 containing additional MIME types, references to payload format for AMR WB added, clause 8 rewritten, new clause B.4 added, Annex C about additional SMIL modules removed, new Annex about MIME types added.

Decision about audio codec added, annex D about MP4 added, corrections from PSM SWG review done

Outstanding Issues:

None.

Contentious Issues:

None.

3GPP TS 26.234 V2.0.0 (2001-03)

Technical Specification

3rd Generation Partnership Project;
Technical Specification Group Services and System Aspects;
Transparent end-to-end packet switched streaming service (PSS);
Protocols and codecs
(Release 4)

Keywords

UMTS, IP, packet-switched, streaming, protocol, codec

3GPP

Postal address

3GPP support office address

650 Route des Lucioles - Sophia Antipolis
Valbonne - FRANCE
Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Internet

http://www.3gpp.org

Copyright Notification

Contents

Foreword....................5

Introduction....................5

1 Scope....................6

2 References....................6

3 Definitions and abbreviations....................8
3.1 Definitions....................8
3.2 Abbreviations....................8

4 System description....................9

5 Protocols....................10
5.1 Session establishment....................10
5.2 Capability exchange....................11
5.3 Session set-up and control....................11
5.3.1 General....................11
5.3.2 RTSP....................11
5.3.3 SDP....................11
5.4 MIME media types....................12

6 Data transport....................12
6.1 Packet based network interface....................12
6.2 RTP over UDP/IP....................12
6.3 HTTP over TCP/IP....................13
6.4 Transport of RTSP....................13

7 Codecs....................13
7.1 General....................13
7.2 Speech....................13
7.3 Audio....................13
7.4 Video....................13
7.5 Still images....................14
7.6 Bitmap graphics....................14
7.7 Vector graphics....................14
7.8 Text....................14

8 Scene description....................14
8.1 General....................14
8.2 PSS SMIL module collection....................14

9 Interchange format for MMS....................15
9.1 General....................15
9.2 MPEG-4 file format guidelines....................15
9.2.1 Registration of non-ISO codecs....................15
9.2.2 Hint tracks....................15
9.2.3 Self-contained MP4 files....................15
9.2.4 MPEG-4 systems specific elements....................16

Annex A (informative): Protocols....................17
A.1 SDP....................17
A.2 RTSP....................18

Annex B (informative): SMIL authoring guidelines....................21
B.1 General....................21
B.2 BasicLinking....................21
B.3 BasicLayout....................21
B.4 EventTiming....................21
B.5 MetaInformation....................22

B.6 XML entities 22

Annex C (normative): MIME media types 23

C.1 MIME media type H263-2000 23

C.2 MIME media type xhtml+xml 23

Annex D (normative): Support for non-ISO code streams in MP4 files 24

D.1 General 24

D.2 Sample Description atom 24

D.3 VisualSampleEntry atom 25

D.4 AudioSampleEntry atom 26

D.5 AMRSampleEntry atom 27

D.6 H263SampleEntry atom 28

D.7 DecoderSpecificInfo field for AMRSampleEntry atom 29

D.8 DecoderSpecificInfo field for H263SampleEntry atom 30

Annex E (informative): Change history 32

Foreword

The 3rd Generation Partnership Project (3GPP) Technical Specification Group (TSG) Services and Systems Aspects has produced this Technical Specification (TS).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

x the first digit:

1 presented to TSG for information;

2 presented to TSG for approval;

3 or greater indicates TSG approved document under change control.

y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.

z the third digit is incremented when editorial only changes have been incorporated in the specification;

The 3GPP transparent end-to-end packet-switched streaming service (PSS) specification consists of two 3G TSs; 3GPP TS 26.233 "Transparent end-to end packet switched streaming service (PSS); General description" [2] and the present document. The first TS provides an overview of the 3GPP PSS and the present document the details of protocol and codecs used by the service.

Introduction

Streaming refers to the ability of an application to play synchronised media streams like audio and video streams in a continuous way while those streams are being transmitted to the client over a data network.

Applications, which can be built on top of streaming services, can be classified into on-demand and live information delivery applications. Examples of the first category are music and news-on-demand applications. Live delivery of radio and television programs are examples of the second category.

The 3GPP PSS provides a framework for Internet Protocol (IP) based streaming applications in 3G networks.

1 Scope

The present document specifies the protocols and codecs for the PSS within the 3GPP system. Protocols for control signalling, scene description, media transport and media encapsulations are specified. Codecs for speech, audio, video, still images, bitmap graphics, and text are specified.

The present document is applicable to IP based packet switched networks.

2 References

The following documents contain provisions, which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies.

This specification may contain references to pre-Release-4 GSM specifications. These references shall be taken to refer to the Release 4 version where that version exists. Conversion from the pre-Release-4 number to the Release 4 (onwards) number is given in subclause 6.1 of 3GPP TR 41.001[1].

[1] 3GPP TR 41.001: "GSM Release specifications".

[2] 3GPP TS 26.233: "Transparent end-to-end packet switched streaming service (PSS); General description".

[3] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

[4] IETF RFC 1738: "Uniform Resource Locators (URL)", Berners-Lee, Masinter & McCahill, December 1994.

[5] IETF RFC 2326: "Real Time Streaming Protocol (RTSP)", Schulzrinne H., Rao A. and Lanphier R., April 1998.

[6] IETF RFC 2327: "SDP: Session Description Protocol", Handley M. and Jacobson V., April 1998

[7] IETF STD 0006: "User Datagram Protocol", Postel J., August 1980

[8] IETF STD 0007: "Transmission Control Protocol", September 1981

[9] IETF RFC 1889: "RTP: A Transport Protocol for Real-Time Applications", Schulzrinne H. et al., January 1996.

[10] IETF RFC 1890: "RTP Profile for Audio and Video Conferences with Minimal Control", Schulzrinne H. et al., January 1996.

[11] 3GPP TS 26.235: "Packet Switched Conversational Multimedia Applications; Default Codecs; Annex D: RTP payload format for AMR".

[12] 3GPP TS 26.235: "Packet Switched Conversational Multimedia Applications; Default Codecs; Annex B: AMR-WB RTP payload and MIME type registration".

[13] IETF RFC 3016: "RTP payload format for MPEG-4 audio/visual streams", Kikuchi Y. et al., November 2000.

[14] IETF RFC 2429: "RTP Payload Format for the 1998 Version of ITU-T Rec. H.263 Video (H.263+)", Bormann C. et al., October 1998.

[15] IETF RFC 2046: "Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types", N. Freed, N. Borenstein, November 1996.

[16] IETF RFC 3023: "XML Media Types", Murata, M., St.Laurent, S., Kohn, D., January 2001.

[17] IETF RFC 2616: "Hypertext Transfer Protocol – HTTP/1.1", Fielding R. et al., June 1999.

[18] 3GPP TS 26.071: "Mandatory Speech Codec speech processing functions, AMR Speech Codec; General Description"

[19] 3GPP TS 26.101: "AMR Speech Codec Frame Structure".

[20] 3GPP TS 26.171: "AMR Wideband Speech Codec; General description".

[21] International Standard ISO/IEC 14496-3: "Information technology - Generic coding of audio-visual object – Part 3: Audio, 1999

[22] ITU-T Recommendation H.263: "Video coding for low bitrate communication"

[23] ITU-T Recommendation H.263: "Annex X, Profiles and Levels Definition"

[24] International Standard ISO/IEC 14496-2: "Information technology - Generic coding of audio-visual object – Part 2: Visual", 1999.

[25] ISO/IEC 14496-2:1999/FDAM4, ISO/IEC JTC1/SC 29/WG11 N3904, Pisa, January, 2001

[26] ITU-T Recommendation T.81 | ISO/IEC 10918-1:1992: "Digital compression and coding of continuous-tone still images – requirements and guidelines", (9/91).

[27] "JPEG File Interchange Format", Version 1.02, September 1, 1992.

[28] W3C Recommendation: "XHTML Basic", http://www.w3.org/TR/2000/REC-xhtml-basic-20001219, December 2000

[29] ISO/IEC 10646-1: "2000 Information technology – Universal Multiple-Octet Coded Character Set (UCS) – Part1: Architecture and Basic Multilingual Plane"

[30] The Unicode Consortium: "The Unicode Standard", Version 3.0 Reading, MA, Addison-Wesley Developers Press, 2000, ISBN 0-201-61633-5.

[31] W3C Working Draft Recommendation: "Synchronised Multimedia Integration Language (SMIL 2.0) Specification ", http://www.w3.org/TR/2001/WD-smil20-20010301/

[32] CompuServe Incorporated: "GIF Graphics Interchange Format: A Standard defining a mechanism for the storage and transmission of raster-based graphics information", Columbus, OH, USA, 1987.

[33] CompuServe Incorporated: "Graphics Interchange Format: Version 89a", Columbus, OH, USA, 1990.

[34] International Standard ISO/IEC 14496-1: "Information technology - Coding of audio-visual objects -- Part 1: Systems", 2000.

[35] 3GPP TS 23.140: "Multimedia Messaging Service (MMS); Functional description; Stage 2".

3 Definitions and abbreviations

3.1 Definitions

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

continuous media: media with an inherent notion of time, in this document speech, audio and video

discrete media: media that itself does not contain an element of time, in this document all media not defined as continuous media

presentation description: contains information about one or more media streams within a presentation, such as the set of encodings, network addresses and information about the content

PSS client: a client for the 3GPP packet based streaming service based on the IETF RTSP/SDP and/or HTTP standards, with possible additional 3GPP requirements according to the present document

PSS server: a server for the 3GPP packet based streaming service based on the IETF RTSP/SDP and/or HTTP standards, with possible additional 3GPP requirements according to the present document

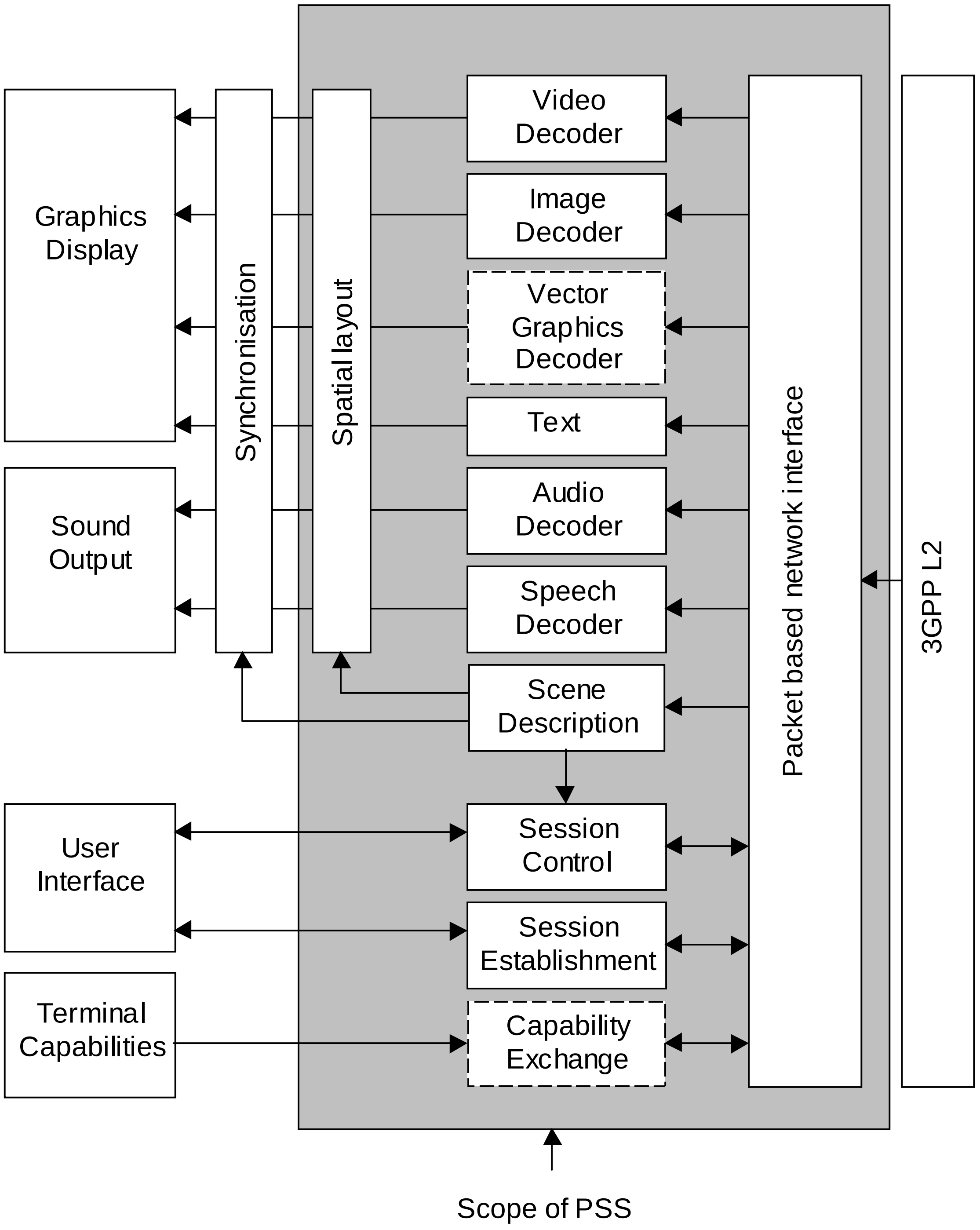
scene description: a description of the spatial layout and temporal behaviour of a presentation, it can also contain hyperlinks

3.2 Abbreviations

For the purposes of the present document, the abbreviations given in TR 3G 21.905 [3] and the following apply.

AAC	Advanced Audio Coding
BIFS	Binary Format for Scene description
DCT	Discrete Cosine Transform
GIF	Graphics Interchange Format
HTML	Hyper Text Markup Language
ITU-T	International Telecommunications Union – Telecommunications
JFIF	JPEG File Interchange Format
MIME	Multipurpose Internet Mail Extensions
MMS	Multimedia Messaging Service
MP4	MPEG-4 file format
QCIF	Quarter Common Intermediate Format
PSS	Packet-switched Streaming Service
RTCP	RTP Control Protocol
RTP	Real-time Transport Protocol
RTSP	Real-Time Streaming Protocol
SDP	Session Description Protocol
SMIL	Synchronised Multimedia Integration Language
UCS-2	Universal Character Set (the two octet form)
UTF-8	Unicode Transformation Format (the 8-bit form)
W3C	WWW Consortium
WML	Wireless Markup Language
XHTML	eXtensible Hyper Text Markup Language
XML	eXtensible Markup Language

4 System description



NOTE: Dashed components are not specified for the simple PSS.

Figure 1: Functional components of a PSS client

Figure 1 shows the functional components of a PSS client. Figure 2 gives an overview of the protocol stack used in a PSS client and also shows a more detailed view of the packet based network interface. The functional components can be divided into control, scene description, media codecs and the transport of media and control data. TS 26.233

"Transparent end-to end packet switched streaming service (PSS); General description" [2] defines the simple and extended PSS. Dashed functional components in figure 1 are not specified for the simple PSS.

The control related elements are session establishment, capability exchange and session control (see clause 5).

- Session establishment refers to methods to invoke a PSS session from a browser or directly by entering an URL in the terminal's user interface.

- Capability exchange enables choice or adaptation of media streams depending on different terminal capabilities.

- Session control deals with the set-up of the individual media streams between a PSS client and one or several PSS servers. It also enables control of the individual media streams by the user. It may involve VCR-like presentation control functions like start, pause, fast forward and stop of a media presentation.

The scene description consists of spatial layout and a description of the temporal relation between different media that is included in the media presentation. The first gives the layout of different media components on the screen and the latter controls the synchronisation of the different media (see clause 8).

The PSS includes media codecs for video, still images, bitmap graphics, text, audio, and speech (see clause 7).

Transport of media and control data consists of the encapsulation of the coded media and control data in a transport protocol (see clause 6). This is shown in figure 1 as the "packet based network interface" and displayed in more detail in the protocol stack of figure 2.

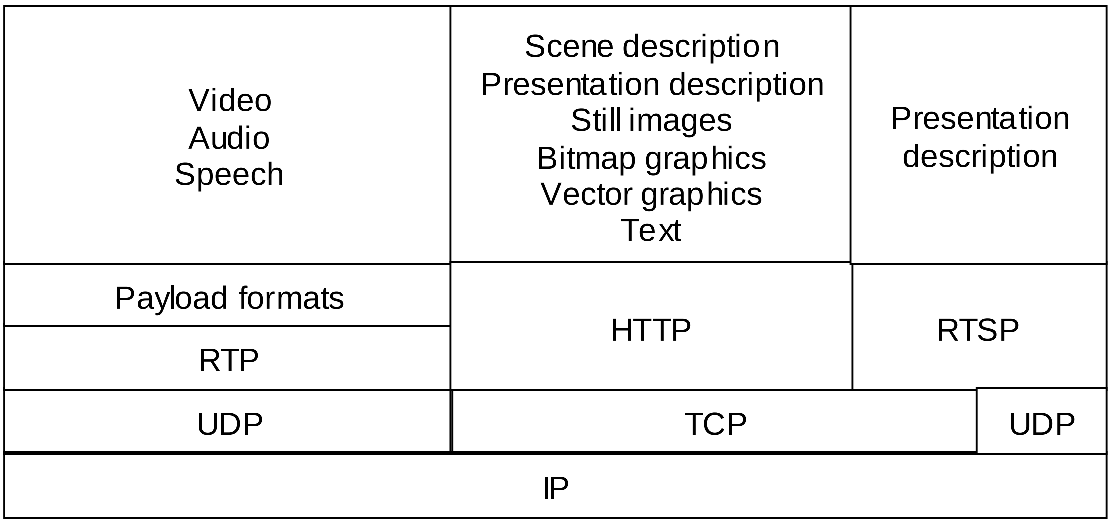


Figure 2: Overview of the protocol stack

5 Protocols

5.1 Session establishment

Session establishment refers to the method by which a PSS client obtains the initial session description. The initial session description can e.g. be a presentation description, a scene description or just an URL to the content.

A PSS client shall support initial session descriptions specified in one of the following formats: SMIL, SDP, or plain RTSP URL.

In addition to rtsp:// the PSS client shall support URLs [4] to valid initial session descriptions starting with file:// (for locally stored files) and http:// (for presentation descriptions or scene descriptions delivered via HTTP).

Examples for valid inputs to a PSS client are: file://temp/morning_news.smil, http://mediaportal/morning_news.sdp, and rtsp://mediaportal/morning_news.

URLs can be made available to a PSS client in many different ways. It is out of the scope of this recommendation to mandate any specific mechanism. However, an application using the 3GPP PSS shall at least support URLs of the above type, specified or selected by the user.

The preferred way would be to embed URLs to initial session descriptions within HTML or WML pages. Browser applications that support the HTTP protocol could then download the initial session description and pass the content to the PSS client for further processing. How exactly this is done is an implementation specific issue and out of the scope of this recommendation.

5.2 Capability exchange

No explicit capability exchange protocol is specified for the simple PSS.. Instead it is assumed that the user is aware of that the content he/she is about to stream fits the capabilities, e.g. screen size, of the particular device used. Protocols for capability exchange can be specified for the extended PSS..

5.3 Session set-up and control

5.3.1 General

Continuous media is media that have an intrinsic time line. Discrete media on the other does not it self contain an element of time. In this specification speech, audio and video belongs to first category and still images and text to the latter one. Bitmap graphics can fall into both groups, but is in this specification defined to be discrete media.

Streaming of continuous media using RTP/UDP/IP (see clauses 6.2) requires a session control protocol to set-up and control of the individual media streams. For the transport of discrete media this specification adopts the use of HTTP/TCP/IP (see clause 6.3). In this case there is no need for a separate session set-up and control protocol since this is built into HTTP. This clause describes session set-up and control of continuous media.

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

5.3.3 SDP

RTSP requires a presentation description. SDP shall be used as the format of the presentation description for both PSS clients and servers. PSS servers shall provide and clients interpret the SDP syntax according to the SDP specification [6] and Appendix C of [5]. The SDP delivered to the PSS client shall declare the media types to be used in the session using a codec specific MIME media type for each media. MIME media types to be used in the SDP file are described in subclause 5.4 of the present document.

The SDP [6] specification requires certain fields to always be included in an SDP file. Apart from this a PSS server shall always include the following fields in the SDP:

- "a=control:" according to clause C.1.1, C.2 and C.3 in [5];
- "a=range:" according to clause C.1.5 in [5];
- "a=rtpmap:" according to clause 6 in [6];
- "a=fmtp:" according to clause 6 in [6].

The bandwidth field in SDP can be used to indicate to the PSS client the amount of bandwidth that is required for the session and the individual media in the presentation. Therefore, a PSS server should include the "b=AS:" field in the SDP (both on the session and media level) and a PSS client shall be able to interpret this field. The bandwidth value shall indicate maximum net rates of media streams without lower level packetisation overhead

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 subclause 7.2) MIME media type as defined in [11];
- AMR wide band speech codec (see subclause 7.2) MIME media type as defined in [12];
- MPEG-4 AAC audio codec (see subclause 7.3) MIME media type as defined in RFC 3016 [13].
- MPEG-4 video codec (see subclause 7.4) MIME media type as defined in RFC 3016 [13];
- H.263 video codec (see subclause 7.4) MIME media type as defined in Annex C, clause C.1 of the present document.

MIME media types for JPEG, GIF 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 subclause 7.6) MIME media type as defined in [15];
- GIF (see subclause 7.7) MIME media type as defined in [15];
- XHTML (see subclause 7.8) MIME media type as defined in Annex C clause C.2 of the present document.

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

6 Data transport

6.1 Packet based network interface

PSS clients and servers shall support an IP-based network interface for the transport of session control and media data. Control and media data are sent using TCP/IP [8] and UDP/IP [7]. An overview of the protocol stack can be found in figure 2 of the present document.

6.2 RTP over UDP/IP

The IETF RTP [9] [10] provides a means for sending real-time or streaming data over UDP (see [7]). The encoded media is encapsulated in the RTP packets with media specific RTP payload formats. RTP payload formats are defined by IETF. RTP also provides a protocol called RTCP (see clause 6 in [9]) for feedback about the transmission quality.

RTP/UDP/IP transport of continuous media (speech ,audio and video) shall be supported.

For RTP/UDP/IP transport of continuous media the following RTP payload formats shall be used:

- AMR narrow band speech codec (see subclause 7.2) RTP payload format according to [11];,
- AMR wide band speech codec (see subclause 7.2) RTP payload format according to [12];
- MPEG-4 AAC audio codec (see subclause 7.3) RTP payload format according to RFC 3016 [13];
- MPEG-4 video codec (see subclause 7.4) RTP payload format according to RFC 3016 [13];

- H.263 video codec (see subclause 7.4) RTP payload format according to RFC 2429 [14];

6.3 HTTP over TCP/IP

The IETF TCP provides reliable transport of data over IP networks, but with no delay guarantees. It is the preferred way for sending the scene description, text, bitmap graphics and still images. There is also need for an application protocol to control the transfer. The IETF HTTP [17] provides this functionality.

HTTP/TCP/IP transport shall be supported for

- still images (see subclause 7.5);
- bitmap graphics (see subclause 7.6);
- text (see subclause 7.8);
- scene description (see clause 8);
- presentation description (see clause 5.3.3).

6.4 Transport of RTSP

Transport of RTSP shall be supported according to RFC 2326 [5].

7 Codecs

7.1 General

For PSS offering a particular media type, media codecs are specified in the following clauses.

7.2 Speech

The AMR codec shall be supported for narrow-band speech [18]. The AMR wideband speech codec [20] shall be supported when wideband speech working at 16 kHz sampling frequency is supported.

7.3 Audio

MPEG-4 AAC Low Complexity object type 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 object type may be supported.

7.4 Video

ITU-T H.263 baseline [22] shall be supported. This is the mandatory video codec for the PSS. In addition, PSS should support

- H.263 Profile 3 Level 10, [23];
- MPEG-4 Visual Simple Profile Level 0, [24] and [25].

These two video codecs are optional to implement.

NOTE: ITU-T H.263 baseline has been mandated to ensure that video-enabled PSS support a minimum baseline video capability and interoperability can be guaranteed (an H.263 baseline bitstream can be decoded by both H.263 and MPEG-4 decoders). It also provides a simple upgrade path for mandating more advanced codecs in the future (from both the ITU-T and ISO MPEG).

7.5 Still images

ISO/IEC JPEG [26] together with JFIF [27] shall be supported. The support for ISO/IEC JPEG only apply to the following two modes:

- baseline DCT, non-differential, Huffman coding, as defined in table B.1, symbol 'SOF0' in [26];
- progressive DCT, non-differential, Huffman coding, as defined in table B.1, symbol 'SOF2' [26].

7.6 Bitmap graphics

The following bitmap graphics codecs should be supported:

- GIF87a, [32];
- GIF89a, [33].

7.7 Vector graphics

No vector graphics codec is specified for the simple PSS. For the extended PSS mandatory and/or optional vector graphics codecs can be specified.

7.8 Text

Text shall be formatted according to XHTML Basic [28] [29] [30].

The following character encoding shall be supported:

- UTF-8, [29];
- UCS-2, [30].

8 Scene description

8.1 General

The 3GPP PSS use a subset of SMIL 2.0 [31] as format of the scene description. This subset, or profile, is defined in this clause through the specification of the SMIL 2.0 modules that a minimal 3GPP PSS client shall support. This profile is a subset of the SMIL 2.0 Language Profile, but a superset of the SMIL 2.0 Basic Language Profile. The present document also includes an informative Annex B that provides guidelines for SMIL content authors.

NOTE: The interpretation of this is not that all streaming sessions are required to use SMIL. For some types of sessions, e.g. consisting of one single continuous media or two media synchronised by using RTP timestamps, SMIL may not be needed.

8.2 PSS SMIL module collection

PSS clients and servers offering scene descriptions shall support the SMIL 2.0 Basic Language Profile plus the following SMIL 2.0 modules:

- EventTiming;
- MediaClipping;
- MetaInformation.

The modules in the SMIL 2.0 Basic Language Profile plus the three additional modules mentioned above constitute the PSS SMIL module collection. SMIL requires that a module collection have a unique namespace URI identifier. The namespace URI identifier for the PSS SMIL module collection shall be http://www.3gpp.org/SMIL20/PSS4/.

In addition to the modules specified above, a PSS client should support the PrefetchControl module. This module is optional.

NOTE: The SMIL 2.0 Basic Language Profile is equal to the SMIL 2.0 Host Language Conformance subset of SMIL 2.0 and consists of the modules Structure, BasicContentControl, BasicInlineTiming, BasicLayout, BasicLinking, BasicMedia, BasicTimeContainers, MinMaxTiming, RepeatTiming and SkipContentControl.

9 Interchange format for MMS

9.1 General

The MPEG-4 file format [34] is mandated in [35] to be used for continuous media along the entire delivery chain envisaged by the MMS, independent on whether the final delivery is done by streaming or download, thus enhancing interoperability.

In particular, the following stages are considered:

- upload from the originating terminal to the MMS proxy
- file exchange between MMS servers
- transfer of the media content to the receiving terminal, either by file download or by streaming. In the first case the self-contained file is transferred, whereas in the second case the content is extracted from the file and streamed according to open payload formats. In this case, no trace of the file format remains in the content that goes on the wire/in the air.

Additionally, the MPEG-4 file format can be used for the storage in the servers and the "hint track" mechanism can be used for the preparation for streaming.

The clause 9.2 of the present document gives the necessary requirements to follow for the MPEG-4 file format used in MMS. These requirements will guarantee PSS to interwork with MMS as well as the MPEG-4 file format to be used internally within the MMS system. For PSS servers not interworking with MMS there is no requirement to follow these guidelines.

9.2 MPEG-4 file format guidelines

9.2.1 Registration of non-ISO codecs

How to include the non-ISO code streams AMR narrow-band speech and H.263 encoded video in MP4 files is described in Annex D of the present document

9.2.2 Hint tracks

The hint tracks are a mechanism that the server implementation may choose to use in preparation for the streaming of media content contained in MP4 files. However, it should be observed that the usage of the hint tracks is an internal implementation matter for the server, and it falls outside the scope of the present document.

9.2.3 Self-contained MP4 files

All media in the MP4 file shall be self-contained, i.e. there shall not be referencing to external media data from inside the MP4 file.

9.2.4 MPEG-4 systems specific elements

Tracks relative to MPEG-4 system architectural elements (e.g. BIFS scene description tracks or OD Object descriptors) are optional and shall be ignored. The adoption of the MPEG-4 file format does not imply the usage of MPEG-4 systems architecture. The receiving terminal is not required to implement any of the specific MPEG-4 system architectural elements.

Annex A (informative): Protocols

A.1 SDP

This clause gives some background information on SDP.

Table A.1 provides an overview of the different SDP fields that can be identified in a SDP file.

Table A.1: Overview of fields in SDP

Type	**Description**		**Requirement according to [6].**	**Requirement according to the present document.**
Session Description				
V	Protocol version		R	R
O	Owner/creator and session identifier		R	R
S	Session Name		R	R
I	Session information		O	O
U	URI of description		O	O
E	Email address		O	O
P	Phone number		O	O
C	Connection Information		O	O
B	Bandwidth information	AS	O	R
Z	Time zone adjustments		O	O
K	Encryption key		O	O
A	Session attributes	control	O	R
		range	O	R
Time Description				
T	Time the session is active		R	R
R	Repeat times		O	O
Media Description				
M	Media name and transport address		R	R
I	Media title		O	O
C	Connection information		O	O
B	Bandwidth information	AS	O	R
K	Encryption Key		O	O
A	Attribute Lines	control	O	R
		range	O	R
		fmtp	O	R
		rtpmap	O	R
Note: R = Required, O = Optional				

The example below shows an SDP file that could be sent to a PSS client to initiate unicast streaming of a H.263 video sequence.

EXAMPLE:

```
v=0
o=ghost 2890844526 2890842807 IN IP4 192.168.10.10
s=3GPP Unicast SDP Example
i=Example of Unicast SDP file
u=http://www.infoserver.com/ae600
e=ghost@mailserver.com
c=IN IP4 192.168.30.29
a=range:npt=0-45.678
b=AS:128
t=0 0
m=video 1024 RTP/AVP 96
a=rtpmap:96 H263-2000/90000
a=fmtp:96 profile=3;level=10
a=control:rtsp;//mediaserver.com/movie
a=recvonly
b=AS:128
```

A.2 RTSP

The example below is intended to give some more understanding of how RTSP and SDP are used within the 3GPP PSS. The example assumes that the streaming client has the RTSP URL to a presentation consisting of an H.263 video sequence and AMR speech. RTSP messages sent from the client to the server are in **bold** and messages from the server to the client in *italic*. In the example the server provides aggregate control of the two streams.

EXAMPLE:

DESCRIBE rtsp://mediaserver.com/movie.test RTSP/1.0
CSeq: 1

RTSP/1.0 200 OK
CSeq: 1
Content-Type: application/sdp
Content-Length: 203

v=0
o=- 950814089 950814089 IN IP4 144.132.134.67
s=Example of aggregate control of AMR speech and H.263 video
a=range:npt=0-59.3478
a=control:*

b=AS:77
t=0 0
m=audio 0 RTP/AVP 97
a=rtpmap:97 AMR/8000
a=fmtp:97 mode-set=0,2,5,7; maxframes=1a=control:streamID=0
b=AS:13
m=video 0 RTP/AVP 98
a=rtpmap:98 H263-2000/90000
a=fmtp:98 profile=3;level=10
a=control: streamID=1
b=AS:64

SETUP rtsp://mediaserver.com/movie.test/streamID=0 RTSP/1.0
CSeq: 2
Transport: RTP/AVP/UDP;unicast;client_port=3456-3457

RTSP/1.0 200 OK
CSeq: 2
Transport: RTP/AVP/UDP;unicast;client_port=3456-3457; server_port=5678-5679
Session: dfhyrio90llk

SETUP rtsp://mediaserver.com/movie.test/streamID=1 RTSP/1.0
CSeq: 3
Transport: RTP/AVP/UDP;unicast;client_port=3458-3459
Session: dfhyrio90llk

RTSP/1.0 200 OK
CSeq: 3
Transport: RTP/AVP/UDP;unicast;client_port=3458-3459; server_port=5680-5681
Session: dfhyrio90llk

PLAY rtsp://mediaserver.com/movie.test RTSP/1.0
CSeq: 4
Session: dfhyrio90llk

RTSP/1.0 200 OK
CSeq: 4
Session: dfhyrio90llk
Range: npt=0-
RTP-Info: url= rtsp://mediaserver.com/movie.test/streamID=0; seq=9900093;rtptime=4470048, url= rtsp://mediaserver.com/movie.test/streamID=1; seq=1004096;rtptime=1070549

The user watches the movie for 20 seconds and then decides to fast forward to 10 seconds before the end…

PAUSE rtsp://mediaserver.com/movie.test RTSP/1.0
CSeq: 5
Session: dfhyrio90llk

PLAY rtsp://mediaserver.com/movie.test RTSP/1.0
CSeq: 6
Range: npt=50-59.3478
Session: dfhyrio90llk

RTSP/1.0 200 OK
CSeq: 5
Session: dfhyrio90llk

RTSP/1.0 200 OK
CSeq: 6
Session: dfhyrio90llk
Range: npt=50-59.3478
RTP-Info: url= rtsp://mediaserver.com/movie.test/streamID=0; seq=39900043;rtptime=44470648, url= rtsp://mediaserver.com/movie.test/streamID=1; seq=31004046;rtptime=41090349

After the movie is over the client issues a TEARDOWN to end the session…

TEARDOWN rtsp://mediaserver.com/movie.test RTSP/1.0
CSeq: 7
Session: dfhyrio90llk

RTSP/1.0 200 OK
Cseq: 7
Session: dfhyrio90llk
Connection: close

Annex B (informative): SMIL authoring guidelines

B.1 General

This is an informative Annex for SMIL presentation authors. Authors can expect that PSS clients can handle the SMIL module collection defined in clause 8.2, with the restrictions defined in this Annex. When creating SMIL documents the author is recommended to consider that terminals may have small displays and simple input devices. The media types and their encoding included in the presentation should be restricted to what is described in clause 7 of the present document. Considering that many mobile devices may have limited software and hardware capabilities, the number of media to be played simultaneous should be limited. For example, many devices will not be able to handle more than one video sequence at the time.

B.2 BasicLinking

The Linking Modules define elements and attributes for navigational hyperlinking, either through user interaction or through temporal events. The BasicLinking module defines the a and area elements for basic linking:

a Similar to the "a" element in HTML it provides a link from a media object through the href attribute (which contains the URI of the link’s destination). The "a" element includes a number of attributes for defining the behaviour of the presentation when the link is followed.

area Whereas the a element only allows a link to be associated with a complete media object, the area element allows links to be associated with spatial and/or temporal portions of a media object.

The area element may be useful for enabling services that rely on interactivity where the display size is not big enough to allow the display of links alongside a media (e.g., QCIF video) window. Instead, the user could, for example, click on a watermark logo displayed in the video window to visit the company website.

Even if the area element may be useful some mobile terminals will not be able to handle area elements that include multiple selectable regions within an area element. One reason for this could be that the terminals do not have the appropriate user interface. Such area elements should therefore be avoided. Instead it is recommended that the "a" element be used. If the "area" element is used, the SMIL presentation should also include alternative links to navigate through the presentation; i.e. the author should not create presentations that rely on that the player can handle "area elements.

B.3 BasicLayout

The "fit" attribute defines how different media should be fitted into their respective display regions.

The rendering and layout of some objects on a small display might be difficult and all mobile devices may not support features such as scroll bars; in addition, the root-layout window may represent the full screen of the display. Therefore "fit=scroll" should not be used.

Due to hardware restrictions in mobile devices, operations such that scaling of a video sequence, or even images, may be very difficult to achieve. According to the SMIL 2.0 specification SMIL players may in these situations clip the content instead. To be sure of that the presentation is displayed as the author intended, content should be encoded in a size suitable for the terminals intended and it is recommended to use "fit=hidden".

B.4 EventTiming

The two attributes "endEvent" and "repeatEvent" in the EventTiming module may cause problems for a mobile SMIL player. The end of a media element triggers the "endEvent". In the same way the "repeatEvent" occurs when the second and subsequent iterations of a repeated element begin playback. Both these events rely on that the SMIL player receives information about that the media element has ended. One example could be when the end of a video sequence initiates the event. If the player has not received explicit information about the duration of the video sequence, e.g. by the "dur" attribute in SMIL or by some external source as the "a=range" field in SDP. The player will have to rely on the RTCP BYE message to decide when the video sequence ends. If the RTCP BYE message is lost, the player will have problems initiate the event. For these reasons is recommended that the "endEvent" and "repeatEvent" attributes are used with care, and if used the player should be provided with some additional information about the duration of the media element that triggers the event. This additional information could e.g. be the "dur" attribute in SMIL or the "a=range" field in SDP.

The "inBoundsEvent" and "outOfBoundsEvent" attributes assume that the terminal has a pointer device for moving the focus to within a window (i.e., clicking within a window). Not all terminals will support this functionality since they do not have the appropriate user interface. Hence care should be taken in using these particular event triggers.

B.5 MetaInformation

Authors are encouraged to make use of meta data whenever providing such information to the mobile terminal appears to be useful. However, they should keep in mind that some mobile terminals will parse but not process the meta data.

Furthermore, authors should keep in mind that excessive use of meta data will substantially increase the file size of the SMIL presentation that needs to be transferred to the mobile terminal. This may result in longer set-up times.

B.6 XML entities

Entities are a mechanism to insert XML fragments inside an XML document. Entities can be internal, essentially a macro expansion, or external. Use of XML entities in SMIL presentations is not recommended, as many current XML parsers do not fully support them.

Annex C (normative): MIME media types

C.1 MIME media type H263-2000

MIME media type name: video
MIME subtype name: H263-2000

Required parameters: None

Optional parameters:
profile: H.263 profile number, in the range 0 through 8, specifying the supported H.263 annexes/subparts.
level: Level of bitstream operation, in the range 0 through 99, specifying the level of computational complexity of the decoding process. When no profile and level parameters are specified, Baseline Profile (Profile 0) level 10 are the default values.

The profile and level specifications can be found in [23]. Note that the RTP payload format for H263-2000 is the same as for H263-1998 and is defined in [14], but additional annexes/subparts are specified along with the profiles and levels.

NOTE: The above text will be replaced with a reference to the RFC describing the H263-2000 MIME media type as soon as this becomes available.

C.2 MIME media type xhtml+xml

MIME media type name: application
MIME subtype name: xhtml+xml

Required parameters: none

Optional parameters:
charset: This parameter has identical semantics to the charset parameter of the "application/xml" media type as specified in [16].

NOTE: The above text will be replaced with a reference to the RFC describing the xhtml+xml MIME media type as soon as this becomes available.

Annex D (normative): Support for non-ISO code streams in MP4 files

D.1 General

The purpose of this Annex is to define the necessary structure for integration of the H.263 and AMR media specific information in an MP4 file. Clauses D.2-D.4 give some background information about the Sample Description atom, VisualSampleEntry atom and the AudioSampleEntry atom in the MPEG-4 file format. Then, the definitions of the SampleEntry atoms for AMR and H.263 are given in clauses D.5-D.8.

AMR data is stored in the stream according to clause 8 of [11].

D.2 Sample Description atom

In an MP4 file, Sample Description Atom gives detailed information about the coding type used, and any initialisation information needed for that coding. The Sample Description Atom can be found in the MP4 Atom Structure Hierarchy shown in figure D.1:

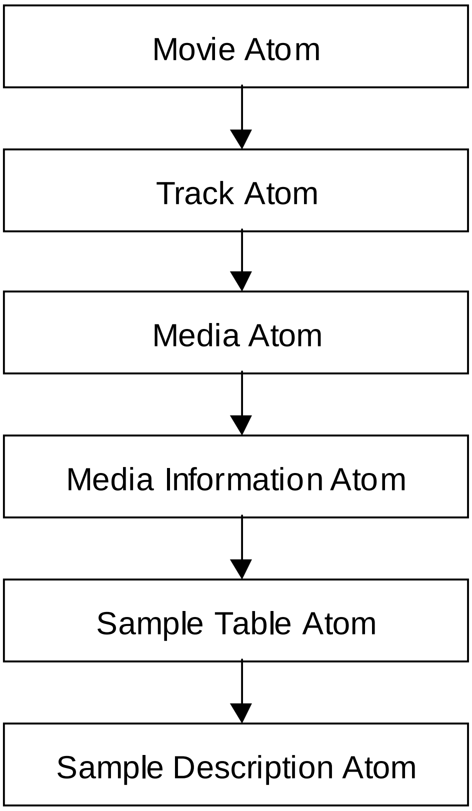


Figure D.1: MP4 Atom Structure Hierarchy

The Sample Description Atom can have one or more SampleDescriptionEntry fields. Valid Sample Description Entry atoms already defined for MP4 are AudioSampleEntry, VideoSampleEntry, HintSampleEntry and MPEGSampleEntry Atoms. The Sample DescriptionEntry Atoms for AMR and H.263 shall be AMRSampleEntry and H263SampleEntry, respectively.

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 speech samples defined in clause D.5 of the present document.	

From the above 5 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 this document.

D.3 VisualSampleEntry atom

The VisualSampleEntry Atom is defined as follows:

VisualSampleEntry ::= AtomHeader

Reserved_6

Data-reference-index

Reserved_16

Reserved_4

Reserved_4

Reserved_4

Reserved_4

Reserved_2

Reserved_32

Reserved_2

Reserved_2

ESDAtom

Table D.2: VisualSampleEntry fields

Field	Type	Details	Value
AtomHeader.Size	Unsigned int(32)		
AtomHeader.Type	Unsigned int(32)		'mp4v'
Reserved_6	Unsigned int(8)		0
Data-reference-index	Unsigned int(16)	Index to a data reference that to use to retrieve the sample data. Data references are stored in data reference Atoms.	
Reserver_16	Const unsigned int(32)		0
Reserved_4	Const unsigned int(32)		0x014000f0
Reserved_4	Const unsigned int(32)		0x00480000
Reserved_4	Const unsigned int(32)		0x00480000
Reserved_4	Const unsigned int(32)		0
Reserved_2	Const unsigned int(16)		1
Reserved_32	Const unsigned int(8)		0
Reserved_2	Const unsigned int(16)		24
Reserved_2	Const int(16)		-1
ESDAtom		Elementary stream descriptor for this stream.	

The stream type specific information is in the ESDAtom structure, which will be explained later.

D.4 AudioSampleEntry atom

AudioSampleEntryAtom is defined as follows:

AudioSampleEntry ::= AtomHeader

Reserved_6

Data-reference-index

Reserved_8

Reserved_2

Reserved_2

Reserved_4

TimeScale

Reserved_2

ESDAtom

Table D.3: AudioSampleEntry fields

Field	Type	Details	Value
AtomHeader.Size	Unsigned int(32)		
AtomHeader.Type	Unsigned int(32)		'mp4a'
Reserved_6	Unsigned int(8)		0
Data-reference-index	Unsigned int(16)	Index to a data reference that to use to retrieve the sample data. Data references are stored in data reference Atoms.	
Reserved_8	Const unsigned int(32)		0
Reserved_2	Const unsigned int(16)		2
Reserved_2	Const unsigned int(16)		16
Reserved_4	Const unsigned int(32)		0
TimeScale	Unsigned int(16)	Copied from track	
Reserved_2	Const unsigned int(16)		0
ESDAtom		Elementary stream descriptor for this stream.	

The stream type specific information is in the ESDAtom structure, which will be explained later.

D.5 AMRSampleEntry atom

The atom type of the AMRSampleEntry Atom shall be 'samr'.

The AMRSampleEntry Atom is defined as follows:

AMRSampleEntry ::= AtomHeader

Reserved_6

Data-reference-index

Reserved_8

Reserved_2

Reserved_2

Reserved_4

TimeScale

Reserved_2

DecoderSpecificInfo

Table D.4: AMRSampleEntry fields

Field	Type	Details	Value
AtomHeader.Size	Unsigned int(32)		
AtomHeader.Type	Unsigned int(32)		'samr'
Reserved_6	Unsigned int(8)		0
Data-reference-index	Unsigned int(16)	Index to a data reference that to use to retrieve the sample data. Data references are stored in data reference Atoms.	
Reserved_8	Const unsigned int(32)		0
Reserved_2	Const unsigned int(16)		2
Reserved_2	Const unsigned int(16)		16
Reserved_4	Const unsigned int(32)		0
TimeScale	Unsigned int(16)	Copied from media header atom of this media	
Reserved_2	Const unsigned int(16)		0
DecoderSpecificInfo		Information specific to the decoder.	

If one compares the AudioSampleEntry Atom - AMRSampleEntry Atom the main difference is in the replacement of the ESDAtom, which is specific to MPEG-4 systems, with an atom suitable for AMR. The DecoderSpecificInfo field structure for AMR is described in clause D.7.

D.6 H263SampleEntry atom

The atom type of the H263SampleEntry Atom shall be 's263'.

The AMRSampleEntry Atom is defined as follows:

H263SampleEntry ::= AtomHeader

Reserved_6

Data-reference-index

Reserved_16

Reserved_4

Reserved_4

Reserved_4

Reserved_4

Reserved_2

Reserved_32

Reserved_2

Reserved_2

DecoderSpecificInfo

Table D.5: H263SampleEntry fields

Field	Type	Details	Value
AtomHeader.Size	Unsigned int(32)		
AtomHeader.Type	Unsigned int(32)		's263'
Reserved_6	Unsigned int(8)		0
Data-reference-index	Unsigned int(16)	Index to a data reference that to use to retrieve the sample data. Data references are stored in data reference Atoms.	
Reserver_16	Const unsigned int(32)		0
Reserved_4	Const unsigned int(32)		0x014000f0
Reserved_4	Const unsigned int(32)		0x00480000
Reserved_4	Const unsigned int(32)		0x00480000
Reserved_4	Const unsigned int(32)		0
Reserved_2	Const unsigned int(16)		1
Reserved_32	Const unsigned int(8)		0
Reserved_2	Const unsigned int(16)		24
Reserved_2	Const int(16)		-1
DecoderSpecificInfo		Information specific to the decoder.	

If one compares the VisualSampleEntry – H263SampleEntry Atom the main difference is in the replacement of the ESDAtom, which is specific to MPEG-4 systems, with an atom suitable for H.263. The DecoderSpecificInfo field structure for H.263 is described in clause D.8.

D.7 DecoderSpecificInfo field for AMRSampleEntry atom

The DecoderSpecificInfo fields for AMR shall be as defined in table D.6. The DecoderSpecificInfo for the AMRSampleEntry Atom shall always be included if the MP4 file contains AMR media.

Table D.6: The DecoderSpecificInfo fields for AMRSampleEntry

Field	Type	Details	Value
DecSpecificInfoTag	Bit(8)		0x05
SizeOfDecSpecificInfo	Unsigned int(32)		
DecSpecificInfo	AMRDecSpecStruc	Structure which holds the AMR Specific information	

DecSpecificInfoTag: identifies that this is a DecoderSpecificInfo Field. It must be set to 0x05.

SizeOfDecSpecificInfo: defines the size (in Bytes) of the DecSpecificInfo structure following.

DecSpecificInfo: the structure where the AMR stream specific information resides.

The AMRDecSpecStruc is defined as follows:

```
struct AMRDecSpecStruc{
                    Unsigned int (32)   vendor
                    Unsigned int (8)    decoder_version
                    Unsigned int (16)   mode_set
                    Unsigned int (8)    mode_change_period
                    Unsigned int (8)    frames_per_sample
}
```

The definitions of AMRDecSpecStruc members are as follows:

vendor: four character code of the manufacturer of the codec, e.g. ‘VXYZ’

decoder_version : version of the decoder which created the AMR stream being stored, the value is set to 0 if version has no importance

mode_set : The active codec modes. A value of 0x1F means all modes are possibly present in the AMR stream. Each bit of the mode_set parameter corresponds to one mode. The bit index of the mode is calculated according to the 4 bit FT field of the AMR frame structure. The mapping of existing AMR modes to FT is given in Table 1.a in [19]. The mode_set bit structure is as follows: (B15xxxxxxB8B7xxxxxxB0) where B0 (Least Significant Bit) corresponds to Mode 0, and B8 corresponds to Mode 8. As an example, if mode_set = 0000000110010101b, only AMR Modes 0, 2, 4, 7 and 8 are present in the AMR stream.

mode_change_period: Defines a number N, which restricts the mode changes only at a multiple of N frames. If no restriction is applied, this value should be set to 0. If mode_change_period is not 0, the following restrictions apply to it according to the frames_per_sample field:

```
if (mode_change_period < frames_per_sample)
    frames_per_sample  = k x (mode_change_period)
else if (mode_change_period > frames_per_sample)
    mode_change_period = k x (frames_per_sample)
where k : integer [2, …]
```

If mode_change_period is equal to frames_per_sample, then AMR mode is the same for all frames inside one sample.

frames_per_sample: Defines the number of frames to be considered as 'one sample' inside the MP4 file. This number should be greater than 0. A value of 1 means each frame is treated as one sample. A value of 10 means that 10 AMR frames (of duration 20 msec each) are put together and treated as one sample. It must be noted that, in this case, one sample duration is 20 (msec / frame) x 10 (frame) = 200 msec. For the last sample of the AMR stream, the number of frames can be smaller than frames_per_sample, if the number of remaining frames is smaller than frames_per_sample.

NOTE: The "hinter", for the creation of the hint tracks, can use the information given by the AMRDecSpecStruc members.

D.8 DecoderSpecificInfo field for H263SampleEntry atom

The DecoderSpecificInfo fields for H. 263 shall be as defined in table D.7. The DecoderSpecificInfo for the H263SampleEntry Atom shall always be included if the MP4 file contains H.263 media.

The DecoderSpecificInfo for H263 is composed of the following fields:

Table D.7: The DecoderSpecificInfo fields H263SampleEntry

Field	Type	Details	Value
DecSpecificInfoTag	Bit(8)		0x05
SizeOfDecSpecificInfo	Unsigned int(32)		
DecSpecificInfo	H263DecSpecStruc	Structure which holds the H.263 Specific information	

DecSpecificInfoTag: It identifies that this is a DecoderSpecificInfo field. It shall be set to 0x05.

SizeOfDecSpecificInfo: It defines the size (in Bytes) of the DecSpecificInfo structure following.

DecSpecificInfo: This is the structure where the H263 stream specific information resides.

H263DecSpecStruc is defined as follows:

struct **H263DecSpecStruc**{

Unsigned int (32)	**vendor**
Unsigned int (8)	**decoder_version**
Unsigned int (8)	**H263_Level**
Unsigned int (8)	**H263_Profile**
Unsigned int (16)	**max_width**
Unsigned int (16)	**max_height**

}

The definitions of H263DecSpecStruc members are as follows:

vendor : Four character code of the manufacturer of the codec, e.g. ‘VXYZ’

decoder_version: Version of the decoder which created the H263 stream being stored. This value is set to 0 if version has no importance.

H263_Level and H263_Profile: These two parameters define which H263 profile and level is used. These parameters are based on the MIME media type video/H263-2000. The profile and level specifications can be found in [23].

EXAMPLE 1: H.263 Baseline = {H263_Level = 10, H263_Profile = 0}

EXAMPLE 2: H.263 Profile 3 @ Level 10 = {H263_Level = 10 , H263_Profile = 3}

max_width: The maximum width of encoded image.

max_height: The maximum height of encoded image.

NOTE 1: max_width and max_height parameters together may be used to allocate the necessary memory in the playback device without need to analyse the H.263 stream

NOTE 2: The "hinter", for the creation of the hint tracks, can use the information given by the H263DecSpecStruc members.

Annex E (informative): Change history

Change history							
Date	TSG #	TSG Doc.	CR	Rev	Subject/Comment	Old	New