INTERNATIONAL STANDARD

ISO/IEC 14496-20

First edition
2006-06-15

Information technology — Coding of audio-visual objects —

Part 20: Lightweight Application Scene Representation (LASeR) and Simple Aggregation Format (SAF)

Technologies de l'information — Codage des objets audiovisuels — Partie 20: Représentation de scène d'application allégée (LASeR) et format d'agrégation simple (SAF)

Reference number
ISO/IEC 14496-20:2006(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Foreword v

Introduction vii

1 Scope 1

2 Normative References 2

3 Terms, definitions and abbreviations 3
3.1 Terms and definitions 3
3.2 Abbreviations 3

4 Document Conventions 3

5 Architecture 4

6 Scene Representation 4
6.1 Overview 4
6.2 Relationship with SVG 5
6.3 Timing Model 7
6.4 Execution Model 8
6.5 Supported Events 9
6.6 Encoder Configuration 10
6.7 LASeR Scene Commands 13
6.8 Scene Description Elements 22

7 Simple Aggregation Format (SAF) 34
7.1 Overview 34
7.2 Time and terminal model specification 35
7.3 SAF Packet 35
7.4 SAF Packet Header 37
7.5 SAF Access Unit 37
7.6 SimpleDecoderConfigDescriptor 38
7.7 SimpleDecoderSpecificInfo 39
7.8 RemoteStreamHeader 39
7.9 Cache Unit 40
7.10 EndOfStream 41
7.11 EndOfSAFSession 41

8 Profiles 41
8.1 Overview 41
8.2 LASeR mini 41
8.3 LASeR full 43

9 Compatibility of SAF Packet 44

10 Carriage of LASeR and SAF 45
10.1 Storage of LASeR in MP4 files 45
10.2 Carriage of SAF Streams over HTTP 47
10.3 Carriage of SAF Streams over RTP 47
10.4 Carriage of SAF Streams over MPEG-2 Systems 47

11 Electronic Attachments 47

12 Binary Syntax for the LASeR Encoding 48
12.1 Decoding Process 48
12.2 Binary Syntax 63

13 Usage of ISO/IEC 23001-1 132

13.1 Introduction **132**
13.2 Electronic Attachments **132**
13.3 Type Codecs **132**
13.4 Type codecs for use with ISO/IEC 23001-1 decoders **134**
13.5 DecoderInit **137**
13.6 Decoding Process **137**
Annex A (informative) **Patent statements** **141**
Bibliography **142**

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

ISO/IEC 14496-20 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

ISO/IEC 14496 consists of the following parts, under the general title *Information technology — Coding of audio-visual objects*:

- *Part 1: Systems*
- *Part 2: Visual*
- *Part 3: Audio*
- *Part 4: Conformance testing*
- *Part 5: Reference software*
- *Part 6: Delivery Multimedia Integration Framework (DMIF)*
- *Part 7: Optimized reference software for coding of audio-visual objects* [Technical Report]
- *Part 8: Carriage of ISO/IEC 14496 contents over IP networks*
- *Part 9: Reference hardware description* [Technical Report]
- *Part 10: Advanced Video Coding*
- *Part 11: Scene description and application engine*
- *Part 12: ISO base media file format*
- *Part 13: Intellectual Property Management and Protection (IPMP) extensions*
- *Part 14: MP4 file format*
- *Part 15: Advanced Video Coding (AVC) file format*
- *Part 16: Animation Framework eXtension (AFX)*

— *Part 17: Streaming text format*

— *Part 18: Font compression and streaming*

— *Part 19: Synthesized texture stream*

— *Part 20: Lightweight Application Scene Representation (LASeR) and Simple Aggregation Format (SAF)*

— *Part 21: MPEG-J GFX*

— *Part 22: Open Font Format*

Introduction

ISO/IEC 14496-20 specifies syntax and semantics for:

— The Lightweight Application Scene Representation (LASeR), specified in Clause 6, which is a binary format for encoding 2D scenes and updates of scenes. The binary format and the scene representation (based on SVG Tiny), are both designed to be suitable for lightweight embedded devices such as mobile phones.

— A Simple Aggregation Format (SAF), specified in Clause 7, to efficiently and easily transport LASeR data together with audio and/or video content over various delivery channels. This multiplexing scheme is designed to be simple to implement and to allow efficient demultiplexing on low-end devices.

The International Organization for Standardization (ISO) and International Electrotechnical Commission (IEC) draw attention to the fact that it is claimed that compliance with this document may involve the use of a patent.

The ISO and IEC take no position concerning the evidence, validity and scope of this patent right.

The holder of this patent right has assured the ISO and IEC that he is willing to negotiate licences under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect, the statement of the holder of this patent right is registered with the ISO and IEC. Information may be obtained from the companies listed in Annex A.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights other than those identified in Annex A. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

Information technology — Coding of audio-visual objects —

Part 20: **Lightweight Application Scene Representation (LASeR) and Simple Aggregation Format (SAF)**

1 Scope

This International Standard defines a scene description format (LASeR) and an aggregation format (SAF) respectively suitable for representing and delivering rich-media services to resource-constrained devices such as mobile phones.

LASeR aims at fulfilling all the requirements of rich-media services at the scene description level. LASeR supports:

— an optimized set of objects inherited from SVG to describe rich-media scenes;

— a small set of key compatible extensions over SVG;

— the ability to encode and transmit a LASeR stream and then reconstruct SVG content;

— dynamic updating of the scene to achieve a reactive, smooth and continuous service;

— simple yet efficient compression to improve delivery and parsing times, as well as storage size, one of the design goals being to allow both for a direct implementation of the SDL as documented, as well as for a decoder compliant with ISO/IEC 23001-1 to decode the LASeR bitstream;

— an efficient interface with audio and visual streams with frame-accurate synchronization;

— use of any font format, including the OpenType industry standard; and

— easy conversion from other popular rich-media formats in order to leverage existing content and developer communities.

Technology selection criteria for LASeR included compression efficiency, but also code and memory footprint and performance. Other aims included: scalability, adaptability to the user context, extensibility of the format, ability to define small profiles, feasibility of a J2ME implementation, error resilience and safety of implementations.

SAF aims at fulfilling all the requirements of rich-media services at the interface between media/scene description and existing transport protocols:

— simple aggregation of any type of stream;

— signaling of MPEG and non-MPEG streams;

— optimized packet headers for bandwidth-limited networks;

— easy mapping to popular streaming formats;

— cache management capability; and

— extensibility.

SAF has been designed to complement LASeR for simple, interactive services, bringing:

— efficient and dynamic packaging to cope with high latency networks;

— media interleaving; and

— synchronization support with a very low overhead.

This International Standard defines the usage of SAF for LASeR content. However, LASeR can be used independently from SAF.

2 Normative References

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 13818-1, *Information technology — Generic coding of moving pictures and associated audio information — Part 1:Systems*

ISO/IEC 14496-1, *Information technology — Coding of audio-visual objects — Part 1: Systems*

ISO/IEC 14496-12, *Information technology — Coding of audio-visual objects — Part 12: ISO base media file format*

ISO/IEC 14496-18, *Information technology — Coding of audio-visual objects — Part 18: Font compression and streaming*

RFC 2045, *Multipurpose Internet Mail Extensions (MIME) Part one: Format of Internet message bodies*, http://www.ietf.org/rfc/rfc2045.txt

RFC 2326, *Real Time Streaming Protocol*, http://www.ietf.org/rfc/rfc2326.txt

RFC 2965, *HTTP State Management Mechanism*, http://www.ietf.org/rfc/rfc2965.txt

W3C SVG11, *Scalable Vector Graphics (SVG) 1.1 Specification* [Recommendation], http://www.w3.org/TR/2003/REC-SVG11-20030114/

W3C SMIL2, *Synchronized Multimedia Integration Language (SMIL 2.0) — [Second Edition]*, 07 January 2005. http://www.w3.org/TR/2005/REC-SMIL2-20050107/

W3C CSS, *Cascading Style Sheets, level 2* [Recommendation], http://www.w3.org/TR/1998/REC-CSS2-19980512/

W3C DOM, *Document* Object *Model Level 2 Events Specification, Version 1.0, W3C Recommendation* 13 November, 2000. http://www.w3.org/TR/2000/REC-DOM-Level-2-Events-20001113

W3C, XML, *Events, an Events Syntax for XML, W3C Recommandation* 14 October 2003. http://www.w3.org/TR/2003/REC-xml-events-20031014

W3C *xml:id Version 1.0, W3C Recommendation* 12 July 2005, http://www.w3.org/TR/2005/PR-xml-id-20050712/

W3C Xlink, *XML Linking Language, W3C Recommendation*, 27 June 2001. http://www.w3.org/TR/2001/REC-xlink-20010627/

3 Terms, definitions and abbreviations

3.1 Terms and definitions

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

3.1.1
access unit
individually accessible portion of data within a media stream

NOTE An access unit is the smallest data entity to which timing information can be attributed.

3.1.2
media time line
axis on which times are expressed within the transport or system carrying a LASeR or other stream

3.1.3
normal play time
indicates the stream absolute position relative to the beginning of the presentation

[RFC 2326]

3.1.4
packet
smallest data entity managed by SAF consisting of a header and a payload

3.1.5
scene segment
a set of access units of a LASeR stream, where only the first access unit contains a LASeRHeader

3.1.6
scene time line

axis on which times are expressed within the SVG/LASeR scene, e.g. begin and end

3.2 Abbreviations

CSS Cascading Style Sheets, a W3C standard

SMIL Synchronized Multimedia Integration Language, a W3C standard

SVG Scalable Vector Graphics, a W3C standard

4 Document Conventions

This document uses the following styling conventions for various types of information.

Any name of element, attribute, descriptor or command defined in this specification is styled in bold italic, such as ***Add***. Any name of element, attribute, descriptor or command defined in another specification is prefixed with the name of that specification, such as ***SVG animate*** or ***SMIL video***.

XML examples use the following style:

```
<?xml version="1.0" encoding="UTF-8"?>
<svg width="480" height="360" viewBox="0 0 480 360"
     version="1.1" baseProfile="tiny">
        <defs>    …
```

SDL descriptions of binary syntax use the following style:

```
Insert extends LASeRUpdate {
  const bit(UpdateBits) InsertCode;
  uint(idBits) ref;
```

The following is the style used for ECMA Script:

```
function Insert(parentId, field, value) {…
```

5 Architecture

LaSER is defined in terms of abstract access units, which may be adapted for transmission over a variety of protocols. LaSER streams may be packaged with some or all of their related media into files of the ISO base media file format family (e.g. MP4) and delivered over reliable protocols. There is also a simple aggregation format (SAF), which aggregates a LaSER stream with some or all of its associated media into stream order. SAF may be delivered over reliable or unreliable protocols. Finally, LASeR streams could be adapted to other delivery protocols such as RTP [RFC 2326] or MPEG-2 transport [ISO/IEC 13818-1]; however, the definitions of these mappings is outside the scope of this specification.

Figure 1 presents the LASeR and SAF architecture.

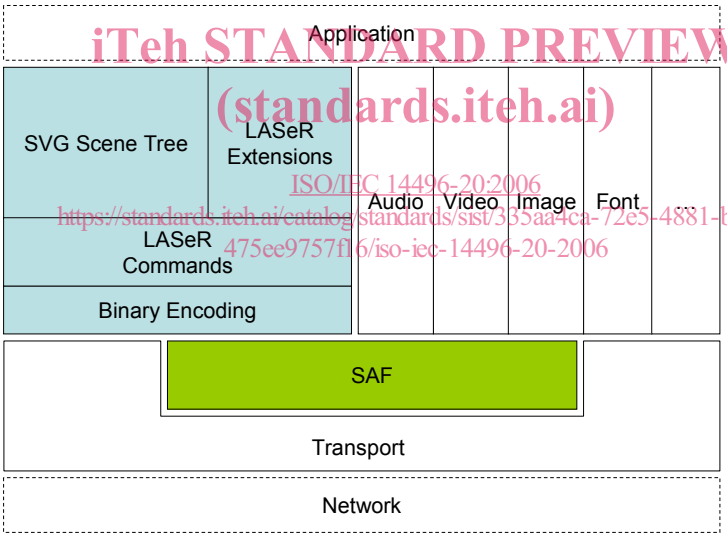

Figure 1 — Architecture of LASeR and SAF

6 Scene Representation

6.1 Overview

In this document, a multimedia presentation is a collection of a scene description and media (zero, one or more). A media is an individual audiovisual content of the following type: image (still picture), video (moving pictures), audio and by extension, font data. A scene description is constituted of text, graphics, animation, interactivity and spatial, audio and temporal layout.

A scene description specifies four aspects of a presentation:

— how the scene elements (media or graphics) are organised spatially, e.g. the spatial layout of the visual elements;

— how the scene elements (media or graphics) are organised temporally, i.e. if and how they are synchronised, when they start or end;

— how to interact with the elements in the scene (media or graphics), e.g. when a user clicks on an image;

— and if the scene is changing, how the scene changes happen.

A scene description may change by means of animations. The different states of the scene during the whole animation may be deterministic (i.e. known when the animation starts) or not. The former case is illustrated by parametric animations. The latter case is illustrated by, for instance, a server sending modification to the scene on the fly. The sequence of a scene description and its timed modifications is called a scene description stream.

The scene description format specified herein is called **LASeR**. A scene description stream is called a **LASeR Stream**. Modifications to the scenes are called **LASeR Commands**. A command is used to act on elements or attributes of the scene at a given instant in time. LASeR Commands that need to be executed at the same time are grouped into one **LASeR Access Unit (AU)**.

This specification defines an XML language to describe scenes which can be encoded with the LASeR format defined throughout subclauses 6.5 to 6.8.37. The exact XML syntax for these elements and attributes is described in the schemas provided as electronic attachments to this specification.

This specification also defines a binary format to efficiently represent 2D scene descriptions.

6.2 Relationship with SVG

6.2.1 Scene tree

The scene constructs on which the binary format defined in this specification is based are the elements defined by the W3C in the SVG specification [W3C SVG11] [2]. Subclause 6.8 explicitly refers to the SVG or SMIL elements and attributes which can be encoded using the binary format defined in this specification. A LASeR scene is an SVG scene possibly with LASeR extensions. These extensions are also defined in this subclause. This specification defines in subclause 6.6.2.3 a set of commands, called LASeR Commands, which can be applied to a LASeR scene.

6.2.2 Fonts

LASeR supports the encoding of fonts. Fonts shall be encoded separately from the scene, e.g. using ISO/IEC 14496-18, and sent as a media stream together with the scene stream. SVG elements related to font description are not supported by LASeR.

NOTE 1 to encode SVG scenes with SVG fonts in LASeR, font information shall be extracted from the SVG scene, encoded separately and sent as a media stream. ISO/IEC 14496-18 is one option to encode and transmit the font, and more options may be specified in the future.

NOTE 2 when using LASeR to encode an SVG scene which includes SVG Fonts derived from OpenType fonts, a better quality can be achieved by transmitting the original OpenType fonts.

NOTE 3 care should be taken when extracting font information from an SVG scene that the effective target of references into the SVG scene, e.g. from scripts, is not changed. One possible way is to replace the extracted font element with a suitable supported (possibly empty) element.

```
<?xml version="1.0" encoding="UTF-8"?>
<svg width="480" height="360" viewBox="0 0 480 360" version="1.1"
     baseProfile="tiny">
        <defs>
            <font horiz-adv-x="959">
                <font-face font-family="TestComic" .../>
                <missing-glyph horiz-adv-x="1024" d="M128 0V1638H896V0H1…"/>
                <glyph unicode="@" horiz-adv-x="1907"
                       d="M1306 412Q1200 412 1123 443T999 ..."/>
                <glyph unicode="A" horiz-adv-x="1498"
                       d="M1250 -30Q1158 -30 1090 206Q1064 ..."/>
                <glyph unicode="y" horiz-adv-x="1066"
                       d="M1011 892L665 144Q537 -129 469 ..."/>
                <glyph unicode="Ö" horiz-adv-x="1635"
                       d="M802 -61Q520 -61 324 108Q116 ..."/>
                <glyph unicode="ç" horiz-adv-x="1052"
                       d="M770 -196Q770 -320 710 -382T528 ..."/>
            </font>
        </defs>
        <g transform="translate(165, 220)" font-family="TestComic"
                font-size="60" fill="black" stroke="none">
            <line x1="0" y1="0" x2="210" y2="0" stroke-width="1"
                  stroke="#888888"/>
            <text>AyÖ@ç</text>
        </g>
</svg>
```


Example 1 — SVG scene with embedded font information


```
<?xml version="1.0" encoding="UTF-8"?>
<saf:SAFSession xmlns:saf="urn:mpeg:mpeg4:SAF:2005" …>
    <saf:sceneHeader>
        <LASeRHeader …/>
    </saf:sceneHeader>
    <saf:mediaHeader streamType="12" objectTypeIndication="6" streamID="font"/>
    <saf:mediaUnit streamIDref="font" .../>
      <!--this media unit contains the OpenType font -->
    <saf:sceneUnit>
        <lsru:NewScene>
            <svg width="480" height="360" viewBox="0 0 480 360" version="1.1"
                 baseProfile="tiny">
                <defs>
                    <desc>this was a font</desc>
                </defs>
                <g transform="translate(165, 220)" font-family="TestComic"
                   font-size="60" fill="black" stroke="none">
                    <line x1="0" y1="0" x2="210" y2="0" stroke-width="1"
                          stroke="#888888"/>
                    <text>AyÖ@ç</text>
                </g>
            </svg>
        </lsru:NewScene>
    </saf:sceneUnit>
    <saf:endOfSAFSession/>
</saf:SAFSession>
```


Example 2 — LASeR/SAF equivalent of Example 1

(the remainder of this subclause is informative)

Differences between example 1 and 2 are:

— the SVG scene has been wrapped in a NewScene update, then in a SAF layer.

— the font description is removed from the SVG scene, encoded with ISO/IEC 14496-18 and placed in a SAF mediaUnit. The attributes streamType=”12” and objectTypeIndication=”6” in the SAF mediaHeader with streamID “font” identify the content of the SAF stream.

— the SAF mediaHeader and SAF mediaUnit are connected through the streamID “font”, which is encoded as a number, and is strictly local to SAF.

— connection between font-family=”TestComic” and the font encoded in the SAF mediaUnit happens through the font name which is part of the OpenType encoding.

6.3 Timing Model

There are Scene Times, Wallclock Times, SMPTE timecodes, Media Times, and Encoded Scene Times. Wallclock times and SMPTE timecodes are not affected by the following discussion.

Logically, a LASeR scene at any instant could be represented by an XML document, which appears like an SVG document:

```
<svg>
...
   <animate begin="X" ... \>
...
</svg>
```

Times within this logical XML document are uniformly expressed in scene times. Scene times have a zero origin and the timescale is defined in SVG.

Logically XML fragments are sent in access units which have Media Time timestamps (MT). These may not have a known origin, and are expressed on a timescale declared at the transport layer. Note that the equations below do not show the correction for timescale units, for simplicity.

The XML fragment containing the "svg" element in this example is sent in an access unit which is a NewScene. The media timestamp MT(ns) of that access unit is arbitrary, but the defined SceneTime of it is zero; ST(ns) = 0.

The XML fragment which supplies the construct "r" is sent in a later access unit with media timestamp MT(r). The defined scene time of that access unit is ST(r) = MT(r) - MT(ns).

Scene times within that access unit ("begin" in this example) are encoded for transmission relative to the scenetime of the access unit. In this example, the "begin" time X is transmitted as the Encoded Scene Time X-ST(r). For LASeR commands inside a script element, ST(r) is the scene time when the script is activated.

A RefreshScene command has an arbitrary media time, as usual, but contains within the access unit the defined SceneTime for that media time. This enables terminals which "tune in" after the NewScene was sent, or for any other reason did not receive the NewScene, to nonetheless establish Scene Times. The encoder could calculate the value of that scenetime by comparing the media timestamp of the RefreshScene MT(rs) with the media timestamp of the preceding NewScene MT(ns), and sending MT(rs) - MT(ns) .

When a scene segment starts with a NewScene, the scene time is reset to 0. In such a scene segment, the scene time of a LASeR access unit is defined as the difference between the media time of that access unit and the media time of the closest previous NewScene.

When a scene segment does not start with a NewScene, the scene time is not reset to 0 and let T_{s0} be the scene time within the initial scene segment upon reception of the first access unit of that new scene segment. In such a scene segment, the scene time of a LASeR access unit is defined as the difference between the media time of that access unit and the media time of the first access unit of that scene segment incremented by T_{s0}. Note: the determination of T_{s0} will vary if there is any variation in delivery times between terminals.