INTERNATIONAL STANDARD

ISO/IEC 14496-14

First edition
2003-11-15

Information technology — Coding of audio-visual objects —

Part 14: MP4 file format

Technologies de l'information — Codage des objets audiovisuels — Partie 14: Format de fichier MP4

Reference number
ISO/IEC 14496-14:2003(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

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 14496-14 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

Part 8: Carriage of ISO/IEC 14496 contents over IP networks

Part 9: Reference hardware description

Part 10: Advanced Video Coding (AVC)

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)

Introduction

0.1 Derivation

This specification defines MP4 as an instance of the ISO Media File format [ISO/IEC 14496-12 and ISO/IEC 15444-12].

The general nature of the ISO Media File format is fully exercised by MP4. MPEG-4 presentations can be highly dynamic, and there is an infrastructure — the Object Descriptor Framework —, which serves to manage the objects and streams in a presentation. An Initial Object Descriptor serves as the starting point for this framework. In the usage modes documented in the ISO Media File, an Initial Object Descriptor would normally be present, as shown in the following diagrams.

0.2 Interchange

The following diagram gives an example of a simple interchange file, containing two streams.

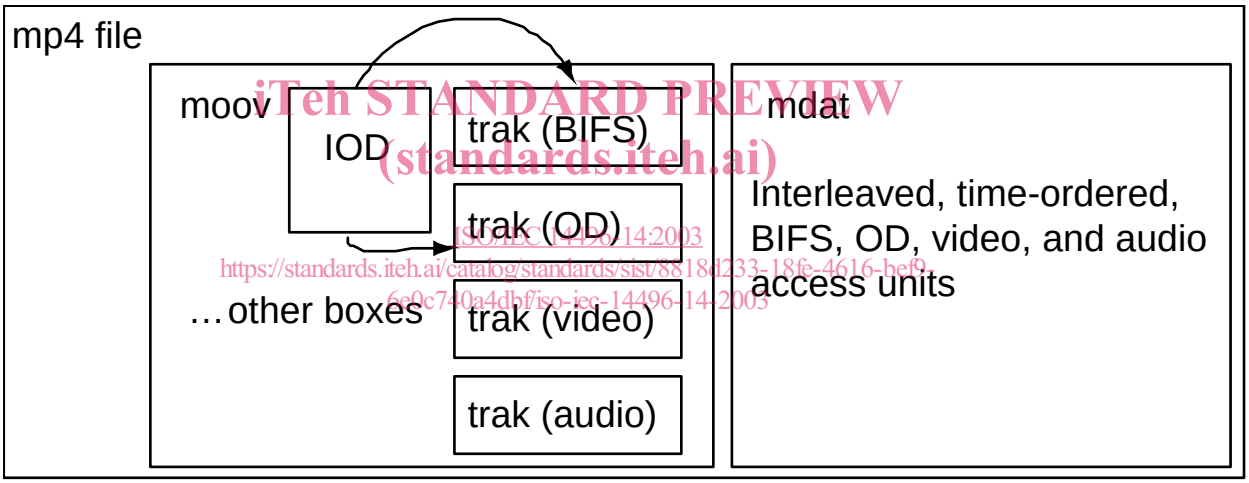

Figure 1 — Simple interchange file

0.3 Content Creation

In the following diagram, a set of files being used in the process of content creation is shown.

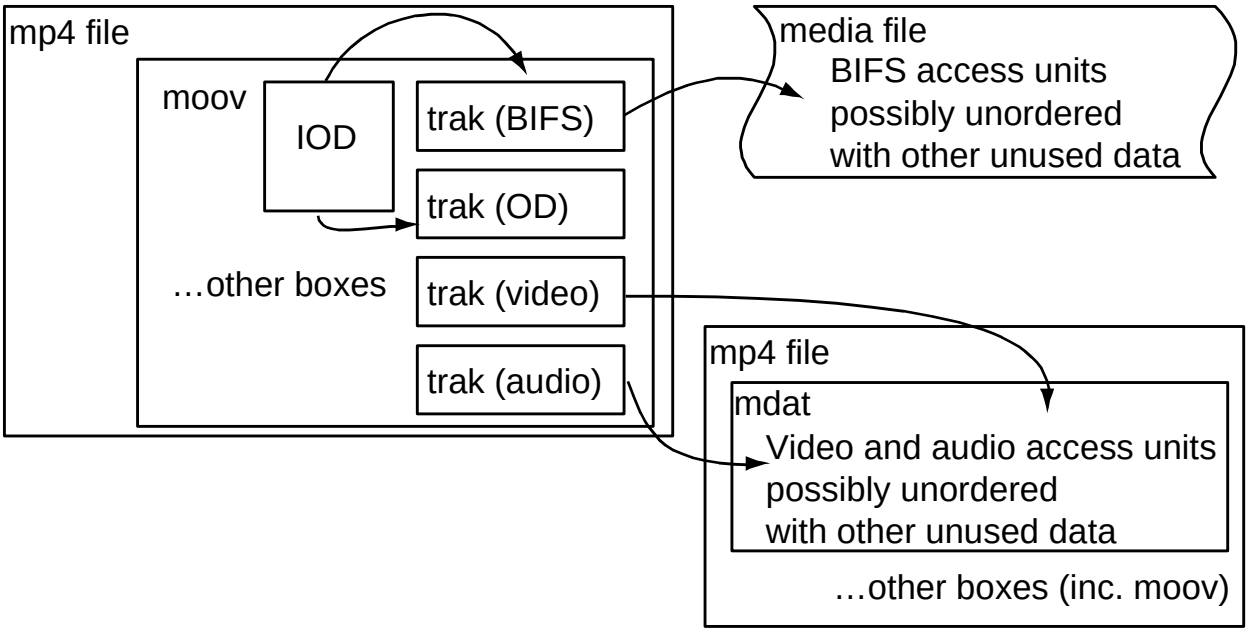

Figure 2 — Content Creation File

0.4 Streamed presentation

The following diagram shows a presentation prepared for streaming over a multiplexing protocol, only one hint track is required.

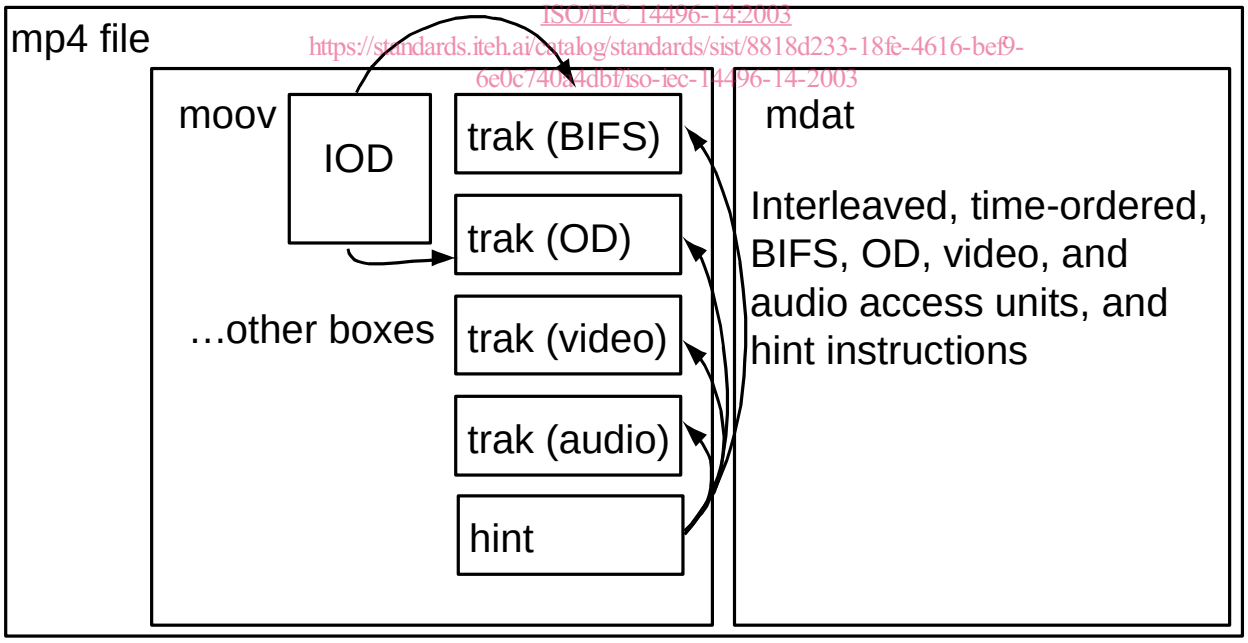

Figure 3 — Hinted Presentation for Streaming

Information technology — Coding of audio-visual objects —

Part 14:
MP4 file format

1 Scope

This International Standard defines the MP4 file format, as derived from the ISO Base Media File format.

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 14496-1:2001, *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* (technically identical to ISO/IEC 15444-12)

3 Storage of MPEG-4

3.1 Elementary Stream Tracks

3.1.1 Elementary Stream Data

To maintain the goals of streaming protocol independence, the media data is stored in its most ‘natural’ format, and not fragmented. This enables easy local manipulation of the media data. Therefore media-data is stored as access units, a range of contiguous bytes for each access unit (a single access unit is the definition of a ‘sample’ for an MPEG-4 media stream). This greatly facilitates the fragmentation process used in hint tracks. The file format can describe and use media data stored in other files, however this restriction still applies. Therefore if a file is to be used which contains ‘pre-fragmented’ media data (e.g. a FlexMux stream on disc), the media data will need to be copied to re-form the access units, in order to import the data into this file format.

This is true for all stream types in this specification, including such ‘meta-information’ streams as Object Descriptor and the Clock Reference. The consequences of this are, on the positive side, that the file format treats all streams equally; on the negative side, this means that there are ‘internal’ cross-links between the streams. This means that adding and removing streams from a presentation will involve more than adding or deleting the track and its associated media-data. Not only must the stream be placed in, or removed from, the scene, but also the object descriptor stream may need updating.

For each track, the entire ES-descriptor is stored as the sample description or descriptions. The SLConfigDescriptor for the media track shall be stored in the file using a default value (predefined = 2), except when the Elementary Stream Descriptor refers to a stream through a URL, i.e. the referred stream is outside the scope of the MP4 file. In that case the SLConfigDescriptor is not constrained to this predefined value.

In a transmitted bit-stream, the access units in the SL Packets are transmitted on byte boundaries. This means that hint tracks will construct SL Packet headers using the information in the media tracks, and the hint tracks will reference the access units from the media track. The placement of the header during hinting is possible without bit shifting, as each SL Packet and corresponding contained access unit will both start on byte boundaries.

3.1.2 Elementary Stream Descriptors

The ESDescriptor for a stream within the scope of the MP4 file as described in this document is stored in the sample description and the fields and included structures are restricted as follows.

- `ES_ID` — set to 0 as stored; when built into a stream, the lower 16 bits of the TrackID are used.
- `streamDependenceFlag` — set to 0 as stored; if a dependency exists, it is indicated using a track reference of type ‘dpnd’.
- `URLflag` — kept untouched, i.e. set to false, as the stream is in the file, not remote.
- `SLConfigDescriptor` — is predefined type 2.
- `OCRStreamFlag` — set to false in the file.

The ESDescriptor for a stream referenced through an ES URL is stored in the sample description and the fields and included structures are restricted as follows.

- `ES_ID` — set to 0 as stored; when built into a stream, the lower 16 bits of the TrackID are used.
- `streamDependenceFlag` — set to 0 as stored; if a dependency exists, it is indicated using a track reference of type ‘dpnd’.

- `URLflag` — kept untouched, i.e. set to true, as the stream is not in the file.
- `SLConfigDescriptor` — kept untouched.
- `OCRStreamFlag` — set to false in the file.

Note that the QoSDescriptor also may need re-writing for transmission as it contains information about PDU sizes etc.

3.1.3 Object Descriptors

The initial object descriptor and object descriptor streams are handled specially within the file format. Object descriptors contain ES descriptors, which in turn contain stream specific information. In addition, to facilitate editing, the information about a track is stored as an ESDescriptor in the sample description within that track. It must be taken from there, re-written as appropriate, and transmitted as part of the OD stream when the presentation is streamed.

As a consequence, ES descriptors are not stored within the OD track or initial object descriptor. Instead, the initial object descriptor has a descriptor used only in the file, containing solely the track ID of the elementary stream. When used, an appropriately re-written ESDescriptor from the referenced track replaces this descriptor. Likewise, OD tracks are linked to ES tracks by track references. Where an ES descriptor would be used within the OD track, another descriptor is used, which again occurs only in the file. It contains the index into the set of mpod track references that this OD track owns. A suitably re-written ESDescriptor replaces it by the hinting of this track.

The ES_ID_Inc is used in the Object Descriptor Box:

```
class ES_ID_Inc extends BaseDescriptor : bit(8) tag=ES_IDIncTag {
   unsigned int(32)    Track_ID;    // ID of the track to use
}

ES_ID_IncTag = 0x0E is reserved for file format usage.
```

The ES_ID_Ref is used in the OD stream:

```
class ES_ID_Ref extends BaseDescriptor : bit(8) tag=ES_IDRefTag {
   bit(16)    ref_index;  // track ref. index of the track to use
}

ES_ID_RefTag = 0x0F is reserved for file format usage.
MP4_IOD_Tag = 0x10 is reserved for file format usage.
MP4_OD_Tag = 0x11 is reserved for file format usage.
IPI_DescrPointerRefTag = 0x12 is reserved for file format usage.
ES_DescrRemoveRefTag = 0x07 is reserved for file format usage (command tag).
```

NOTE The above tag values are defined in 8.2.2.2 Table 1 and 8.2.3.2 Table 2 of the MPEG-4 Systems Specification, and the actual values should be referenced from those tables.

A hinter may need to send more OD events than actually occur in the OD track: for example, if the ES_description changes at a time when there is no event in the OD track. In general, any OD events explicitly authored into the OD track should be sent along with those necessary to indicate other changes. The ES descriptor sent in the OD track is taken from the description of the temporally next sample in the ES track (in decoding time).

3.2 Track Identifiers

The track identifiers used in an MP4 file are unique within that file; no two tracks may use the same identifier.

Each elementary stream in the file is stored as a media track. In the case of an elementary stream, the lower two bytes of the four-byte `track_ID` shall be set to the elementary stream identifier (ES_ID).; the upper two bytes of the track_ID are zero in this case. Hint tracks may use track identifier values in the same range, if this number space is adequate (which it generally is). However, hint track identifiers may also use larger values of track identifier, as their identifiers are not mapped to elementary stream identifiers. Thus very large presentations may use the entire 16-bit number space for elementary stream identifiers.

The next track identifier value, found in next_track_ID in the MovieHeaderBox, as defined in the ISO Base Media Format, generally contains a value one greater than the largest track identifier value found in the file. This enables easy generation of a track identifier under most circumstances. However, if this value is equal to or larger than 65535, and a new media track is to be added, then a search must be made in the file for a free track identifier. If the value is all 1s (32-bit maxint) then this search is needed for all additions.

If it is desired to add a track with a known track identifier (elementary stream identifier) then the file must be searched to ensure that there is no conflict. Note that hint tracks can be re-numbered fairly easily while more care should be taken with media tracks, as there may be references to their ES_ID (track ID) in other tracks.

If hint tracks have track IDs outside the allowed range for elementary stream tracks, then next track ID documents the next available hint track ID. Since this is larger than 65535, a search will then always be needed to find a valid elementary stream track ID.

If two presentations are merged, then there may be conflict between their track IDs. In that case, one or more tracks will have to be re-numbered. There are two actions to be taken here:

- Changing the ID of the track itself, which is easy (track ID in the track header).
- Changing pointers to it.