
**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
iTeh STANDARD PREVIEW
(standards.iteh.ai)

ISO/IEC 14496-14:2020

<https://standards.iteh.ai/catalog/standards/sist/c1e39785-0594-4198-ae57-fbe5d3f4229b/iso-iec-14496-14-2020>



iTeh STANDARD PREVIEW (standards.iteh.ai)

[ISO/IEC 14496-14:2020](https://standards.iteh.ai/catalog/standards/sist/c1e39785-0594-4198-ae57-fbe5d3f4229b/iso-iec-14496-14-2020)

<https://standards.iteh.ai/catalog/standards/sist/c1e39785-0594-4198-ae57-fbe5d3f4229b/iso-iec-14496-14-2020>



COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2020

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Fax: +41 22 749 09 47
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Page

Foreword	iv
Introduction	v
1 Scope	1
2 Normative references	1
3 Terms and definitions	1
4 Storage of MPEG-4	1
4.1 Elementary stream tracks.....	1
4.1.1 Elementary stream data.....	1
4.1.2 Elementary stream descriptors.....	2
4.1.3 Object descriptors.....	2
4.2 Track identifiers.....	3
4.3 Synchronization of streams.....	4
4.4 Composition.....	5
4.5 Handling of M4Mux.....	5
5 File identification	6
6 Additions to the Base Media Format	6
6.1 General.....	6
6.2 Object Descriptor Box.....	7
6.2.1 Description.....	7
6.2.2 Syntax.....	7
6.2.3 Semantics.....	7
6.3 Track reference types.....	7
6.4 Track header box.....	8
6.5 Handler reference types.....	8
6.6 MPEG-4 media header boxes.....	8
6.6.1 General.....	8
6.6.2 Syntax.....	8
6.6.3 Semantics.....	8
6.7 Sample description boxes.....	9
6.7.1 Description.....	9
6.7.2 Syntax.....	10
6.7.3 Semantics.....	10
6.8 Degradation priority values.....	10
7 Template fields used	11
Annex A (informative) Handling of audio timestamps and profile/level indication	12
Bibliography	13

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.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents) or the IEC list of patent declarations received (see <http://patents.iec.ch>).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see www.iso.org/iso/foreword.html.

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

This third edition cancels and replaces the second edition (ISO/IEC 14496-14:2018), of which it constitutes a minor revision. The changes compared to the previous edition are contained in Annex A and the Bibliography.

A list of all parts in the ISO/IEC 14496 series can be found on the ISO website.

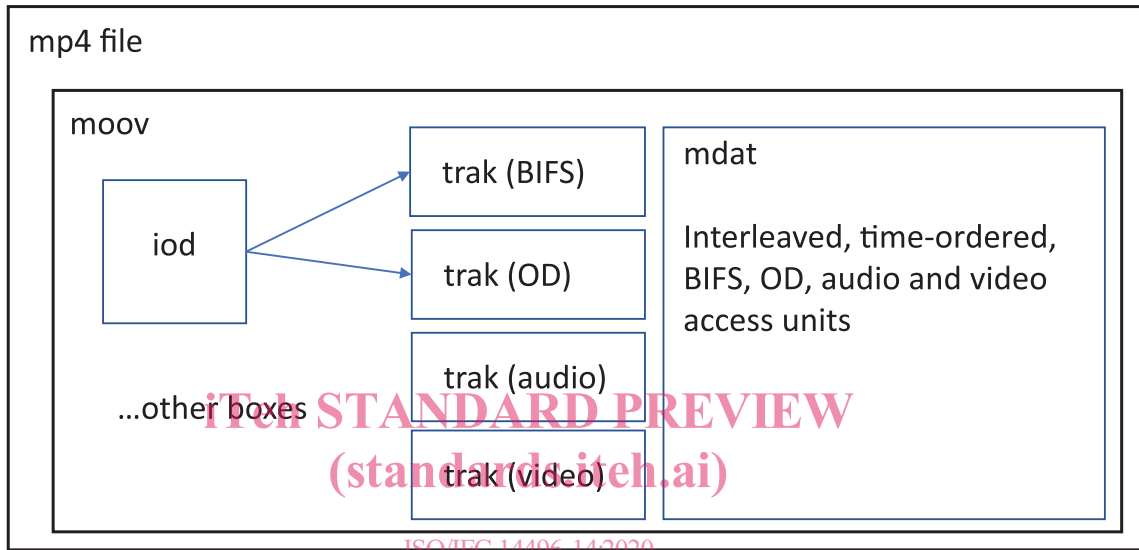
Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

Introduction

This document defines MP4 as an instance of the ISO Media File format (ISO/IEC 14496-12).

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 [Figures 1 to 3](#).

[Figure 1](#) gives an example of a simple interchange file, containing two streams.



ISO/IEC 14496-14:2020
<https://standards.iteh.ai/catalog/standards/sist/c1e39785-0594-4198-ae57-2e29743e2974>
Figure 1 — Simple interchange file

In [Figure 2](#), a set of files being used in the process of content creation is shown.

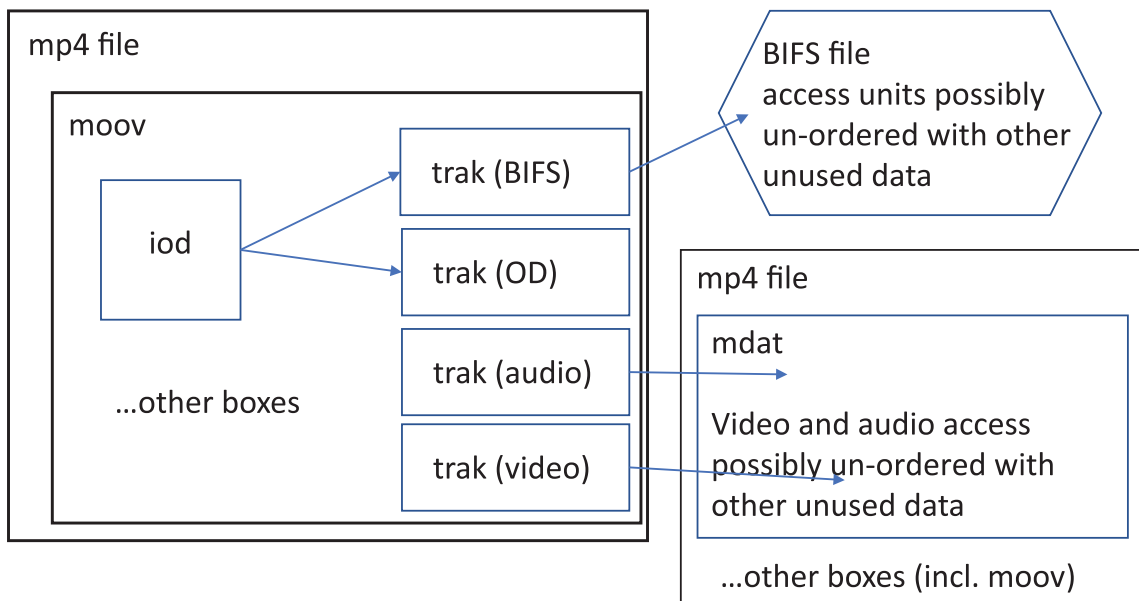
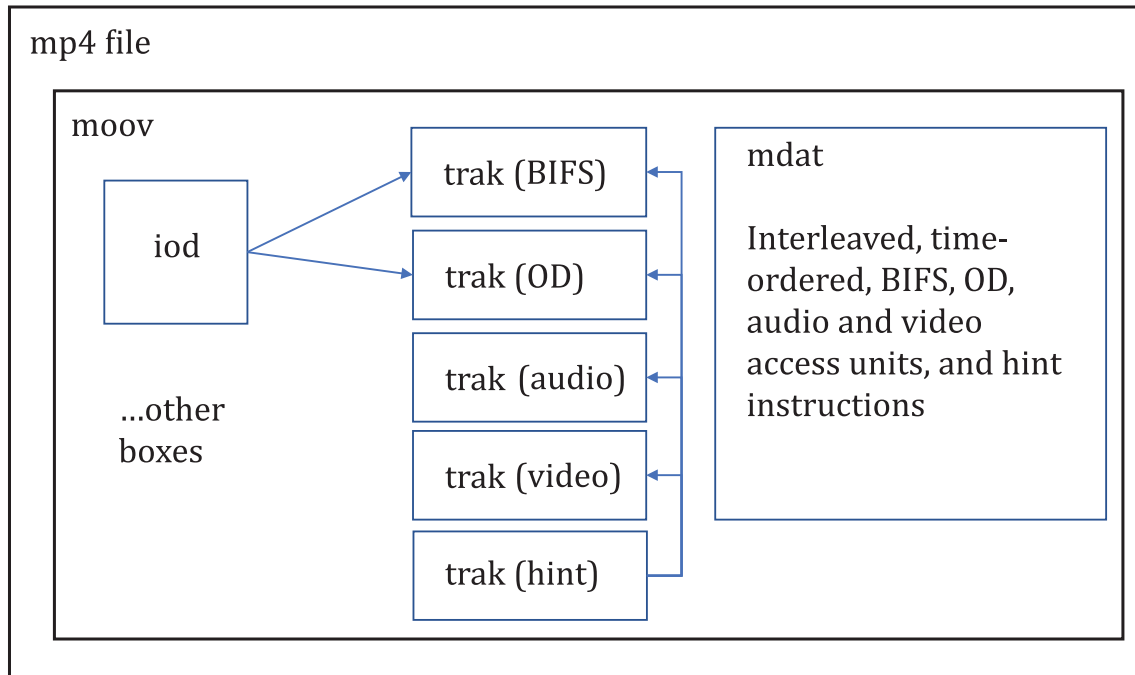


Figure 2 — Content creation file

Figure 3 shows a presentation prepared for streaming over a multiplexing protocol, only one hint track is required.



iTeh STANDARD PREVIEW
Figure 3 — Hinted presentation for streaming
(standards.iteh.ai)

Handling of audio timestamps and profile/level indication is covered in [Annex A](#).

ISO/IEC 14496-14:2020

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.

ISO and IEC take no position concerning the evidence, validity and scope of this patent right. The holder of this patent right has assured ISO and IEC that he/she 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 ISO and IEC. Information may be obtained from the patent database available at www.iso.org/patents.

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

Information technology — Coding of audio-visual objects —

Part 14: MP4 file format

1 Scope

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

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements 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 14496-1:2010, *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*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 14496-12 and the following apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

- IEC Electropedia: available at <http://www.electropedia.org/>
- ISO Online browsing platform: available at <http://www.iso.org/obp>

4 Storage of MPEG-4

4.1 Elementary stream tracks

4.1.1 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. an M4Mux 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 is it necessary that the stream be placed in, or removed from, the scene; the object descriptor stream might also 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 bitstream, 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.

4.1.2 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. [ISO/IEC 14496-14:2020](https://standards.iteh.ai/catalog/standards/sist/c1e39785-0594-4198-ae57-)
<https://standards.iteh.ai/catalog/standards/sist/c1e39785-0594-4198-ae57->

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` may also need re-writing for transmission as it contains information about PDU sizes etc.

4.1.3 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 shall 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

ESdescriptor 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 ISO/IEC 14496-1:2010, 7.2.2.1 and 7.2.2.3.2 (Tables 1 and 2), and the actual values are 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).

ISO/IEC 14496-14:2020

[https://standards.iteh.ai/catalog/standards/sist/c1e39785-0594-4198-ae57-](https://standards.iteh.ai/catalog/standards/sist/c1e39785-0594-4198-ae57-fbe5d3f4229b/iso-iec-14496-14-2020)

4.2 Track identifiers

[fbc5d3f4229b/iso-iec-14496-14-2020](https://standards.iteh.ai/catalog/standards/sist/c1e39785-0594-4198-ae57-fbe5d3f4229b/iso-iec-14496-14-2020)

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 in the file is needed 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 it is necessary to search 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.

The pointers may only occur in the file format structure itself. The file format uses track IDs only through track references, which are easily found and modified. Track IDs become ES_IDs in the MPEG-4 data, and ES_IDs occur within the OD Stream. Since all pointers to ES_IDs in the OD stream are replaced by means of track references there is no need to inspect the OD stream for cross-references within MPEG-4 streams.

In the file format, `ES_DescriptorRemove` command and `IPI_DescrPointer` descriptor are converted to `ES_DescrRemoveRef` and `IPI_DescrPointerRef` by:

- changing the tag value to `ES_DescrRemoveRefTag` or `IPI_DescrPointerRefTag` respectively,
- changing any `ES_ID` to the appropriate track reference index (using references of type `mpod` and `ipir` respectively – see subclause 4.3).

When hinting or serving, the tag value and track reference index changes shall be reversed.

4.3 Synchronization of streams

In the absence of explicit declarations to the contrary, tracks (streams) coming from the same file shall be presented synchronized. This means that hinters and/or servers have to either pick one of the streams to serve as the OCR source for the others or add an OCR stream to associate all the streams with it. Track references of type 'sync' may be used in the file to defeat the default behaviour. In MPEG-4 the `OCRStreamFlag` and `OCR_ES_ID` fields in the `ESDescriptor` govern the synchronization relationships. The mapping of MP4 structures into those fields shall obey the following rules:

- The MPEG-4 `ESDescriptor`, as stored in the file, usually contains `OCRStreamFlag` set to `FALSE`, and no `OCR_ES_ID`. If an `OCR_ES_ID` is set, it is ignored.
- If a track (stream) contains a track reference of type 'sync' whose value is 0, then the hinter or server shall set the `OCRStreamFlag` field in the MPEG-4 `ESDescriptor` to `FALSE` and shall not insert any `OCR_ES_ID` field. This means that this stream is not synchronized to another, but other streams may be synchronized to it.
- If a track (stream) contains a track reference of type 'sync' whose value is not 0, then the hinter or server shall set the `OCRStreamFlag` field in the MPEG-4 `ESDescriptor` to `TRUE` and shall insert an `OCR_ES_ID` field with the same value contained in the 'sync' track reference. This means that this stream is synchronized to the stream indicated in the `OCR_ES_ID`. Other streams may also be synchronized to the same stream, either explicitly or implicitly.
- If a track (stream) does not contain a track reference of type 'sync', then the default behaviour applies. The hinter or server shall set the `OCRStreamFlag` field in the MPEG-4 `ESDescriptor` to `TRUE` and shall insert an `OCR_ES_ID` field with a value selected based on the rules below. This means that this stream is synchronized to the stream indicated in the `OCR_ES_ID`. The rules for selecting the `OCR_ES_ID` are as follows:
 - if no track (stream) in the file contains a track reference of type 'sync', then the hinter picks one `TrackId` and uses that value for the `OCR_ES_ID` field of all `ESDescriptors`. There is one possible exception where the `ESDescriptor` of the stream which corresponds to that `TrackId`, for which the `OCRStreamFlag` may be set to `FALSE`.
 - if one or more tracks (streams) in the file contain a track reference of type 'sync', and all such track references indicate consistently a single `TrackId`, then the hinter uses that `TrackId`. In a track reference of type 'sync' the value 0 is equivalent to the `TrackId` of the track itself.

- if two or more tracks (streams) in the file contain a track reference of type 'sync', and such track references do not indicate a single `TrackId`, then the hinter cannot make a deterministic selection and the behaviour is undefined. In a track reference of type 'sync' the value 0 is equivalent to the `TrackId` of the track itself.

4.4 Composition

In MPEG-4 both visual and aural composition are done using the BIFS system. Therefore structures marked as “template” in the ISO Base Media Format which pertain to composition, including fields such as matrices, layers, graphics modes (and their opcolors), volumes, and balance values, from the `MovieHeaderBox` and `TrackHeaderBox`, are all set to their default values in the file format. These fields do not define visual or audio composition in MPEG-4; in MPEG-4, the BIFS system defines the composition.

The fields `width` and `height` in the `VisualSampleEntry` and in the `TrackHeaderBox` shall be set to the pixel dimensions of the visual stream.

4.5 Handling of M4Mux

An intermediate, optional, fragmentation and packetization step, previously called FlexMux and now called M4Mux, is defined in ISO/IEC 14496-1. Some streaming protocols may carry an M4Mux stream rather than packetized elementary streams. M4Mux may be employed for a variety of purposes, including, but not limited to:

- reducing wasted network bandwidth caused by SL Packet header overhead when the payload is small;
- reducing required server resources when providing many streams, by reducing the number of disk reads or network writes.

The process of building M4Mux PDUs is necessarily aware of the characteristics of the streaming protocol into which the M4Mux is placed. It is not therefore possible to design a streaming protocol-independent handling of M4Mux. Instead, in those streaming protocols where M4Mux is used, the hint tracks for that protocol will encapsulate and include the formation of M4Mux packets. It is expected that the design of the hint tracks will, in this case, closely reflect the way that M4Mux is used. For example, a compact table resembling the `MuxCode` (a method used to associate the payload to M4Mux Channels) mode may be needed if the `interleave` offered by that mode is needed.

In some cases, it may not be possible to create a static M4Mux multiplex via a hint track. Notably, if stream selection is dynamic (for example, based on application feedback) or the choice of `MuxCode` modes or other aspects of M4Mux is dynamic, the M4Mux is therefore created dynamically. This is a necessary cost of run-time multiplexing. It may be difficult for a server to create such a multiplex dynamically at runtime, but with this cost comes added flexibility. A server that wished to provide such functionality could weigh the costs and benefits, and choose to perform the multiplexing without the aid of hint tracks.

Several ISO/IEC 14496 structures are intrinsically linked to M4Mux, and therefore need to be addressed in the context of an M4Mux-aware hint track. For example, a stream map table is required to be supplied to the receiving terminal which maps M4Mux channel IDs to elementary stream IDs. Similarly, if the `MuxCode` mode of M4Mux is used, a `MuxCode` mode structure for each `MuxCode` index used needs to be defined and supplied to the terminal.

These mappings and definitions may change over time, and there is no normative way in ISO/IEC 14496 to supply these to the terminals; instead, some mechanism, associated with the overall system design or protocol used, has to be employed. The hinter needs to store the mappings and definitions. Because they are intimately associated with a particular time-segment of a particular hint track, it is recommended that they be placed in the sample description(s) for that hint track. This description would normally be in the form of:

- a table mapping M4Mux channels to elementary stream IDs;
- a set of `MuxCode` mode structure definitions.