INTERNATIONAL STANDARD

ISO/IEC 15444-12

Second edition
2005-04-01

Corrected version
2005-10-01

Information technology — JPEG 2000 image coding system —

Part 12: ISO base media file format

Technologies de l'information — Système de codage d'image JPEG 2000 —

Partie 12: Format ISO de base pour les fichiers médias

Reference number
ISO/IEC 15444-12:2005(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 vi

Introduction vii

1 Scope 1

2 Normative references 1

3 Definitions 2

4 Object-structured File Organization 3
4.1 File Structure 3
4.2 Object Structure 3
4.3 File Type Box 4

5 Design Considerations 5
5.1 Usage 5
5.1.1 Introduction 5
5.1.2 Interchange 5
5.1.3 Content Creation 6
5.1.4 Preparation for streaming 6
5.1.5 Local presentation 7
5.1.6 Streamed presentation 7
5.2 Design principles 7

6 ISO Base Media File organization 8
6.1 Presentation structure 8
6.1.1 File Structure 8
6.1.2 Object Structure 8
6.1.3 Meta Data and Media Data 8
6.1.4 Track Identifiers 8
6.2 Metadata Structure (Objects) 9
6.2.1 Box 9
6.2.2 Data Types and fields 9
6.2.3 Box Order 10

7 Streaming Support 13
7.1 Handling of Streaming Protocols 13
7.2 Protocol 'hint' tracks 13
7.3 Hint Track Format 14

8 Box Definitions 14
8.1 Movie Box 14
8.2 Media Data Box 15
8.3 Movie Header Box 15
8.4 Track Box 16
8.5 Track Header Box 17
8.6 Track Reference Box 18
8.7 Media Box 19
8.8 Media Header Box 19
8.9 Handler Reference Box 20
8.10 Media Information Box 21
8.11 Media Information Header Boxes 21
8.11.2 Video Media Header Box 21
8.11.3 Sound Media Header Box 22
8.11.4 Hint Media Header Box 22
8.11.5 Null Media Header Box 22

8.12 Data Information Box 23
8.13 Data Reference Box 23
8.14 Sample Table Box 24
8.15 Time to Sample Boxes 24
8.15.2 Decoding Time to Sample Box 25
8.15.3 Composition Time to Sample Box 26
8.16 Sample Description Box 27
8.17 Sample Size Boxes 29
8.17.2 Sample Size Box 30
8.17.3 Compact Sample Size Box 30
8.18 Sample To Chunk Box 31
8.19 Chunk Offset Box 31
8.20 Sync Sample Box 32
8.21 Shadow Sync Sample Box 33
8.22 Degradation Priority Box 34
8.23 Padding Bits Box 34
8.24 Free Space Box 35
8.25 Edit Box 35
8.26 Edit List Box 35
8.27 User Data Box 36
8.28 Copyright Box 37
8.29 Movie Extends Box 37
8.30 Movie Extends Header Box 38
8.31 Track Extends Box 38
8.32 Movie Fragment Box 39
8.33 Movie Fragment Header Box 39
8.34 Track Fragment Box 40
8.35 Track Fragment Header Box 40
8.36 Track Fragment Run Box 41
8.37 Movie Fragment Random Access Box 42
8.38 Track Fragment Random Access Box 43
8.39 Movie Fragment Random Access Offset Box 44
8.40 AVC Extensions 44
8.40.2 Independent and Disposable Samples Box 44
8.40.3 Sample Groups 45
8.40.4 Random Access Recovery Points 48
8.41 Sample Scale Box 49
8.42 Sub-Sample Information Box 50
8.43 Progressive Download Information Box 51
8.44 Metadata Support 52
8.44.1 The Metadata Box 52
8.44.2 XML Boxes 53
8.44.3 The Item Location Box 53
8.44.4 Primary Item Box 55
8.44.5 Item Protection Box 55
8.44.6 Item Information Box 56
8.44.7 URL Forms for meta boxes 56
8.44.8 Static Metadata 57
8.45 Support for Protected Streams 58
8.45.1 Protection Scheme Information Box 59
8.45.2 Original Format Box 59
8.45.3 IPMPInfoBox 60
8.45.4 IPMP Control Box 60
8.45.5 Scheme Type Box 61
8.45.6 Scheme Information Box 62

9 Extensibility 62
9.1 Objects 62
9.2 Storage formats 63
9.3 Derived File formats 63

10 RTP and SRTP Hint Track Format 63
10.1 Introduction 63
10.2 Sample Description Format 64
10.2.1 SRTP Process box 'srpp': 65
10.3 Sample Format 66
10.3.1 Packet Entry format 66
10.3.2 Constructor format 67
10.4 SDP Information 68
10.4.1 Movie SDP information 68
10.4.2 Track SDP Information 68
10.5 Statistical Information 69

Annex A (informative) **Overview and Introduction** 70
A.1 Section Overview 70
A.2 Core Concepts 70
A.3 Physical structure of the media 70
A.4 Temporal structure of the media 71
A.5 Interleave 71
A.6 Composition 71
A.7 Random access 72
A.8 Fragmented movie files 72

Annex B (informative) **Patent Statements** 74

Annex C (informative) **Guidelines on deriving from this specification** 75
C.1 Introduction 75
C.2 General Principles 75
C.3 Brand Identifiers 75
C.3.1 Introduction 75
C.3.2 Usage of the Brand 75
C.3.3 Introduction of a new brand 76
C.3.4 Player Guideline 76
C.3.5 Example 76
C.4 Box layout and order 77
C.5 Storage of new media types 77
C.6 Use of Template fields 77
C.7 Construction of fragmented movies 78

Annex D (informative) **Registration Authority** 79
D.1 Code points to be registered 79
D.2 Procedure for the request of an MPEG-4 registered identifier value 79
D.3 Responsibilities of the Registration Authority 80
D.4 Contact information for the Registration Authority 80
D.5 Responsibilities of Parties Requesting a RID 80
D.6 Appeal Procedure for Denied Applications 81
D.7 Registration Application Form 81
D.7.1 Contact Information of organization requesting a RID 81
D.7.2 Request for a specific RID 81
D.7.3 Short description of RID that is in use and date system was implemented 82
D.7.4 Statement of an intention to apply the assigned RID 82
D.7.5 Date of intended implementation of the RID 82
D.7.6 Authorized representative 82
D.7.7 For official use of the Registration Authority 83

Bibliography 84

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.

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.

ISO/IEC 15444-12 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information Technology*, Subcommittee SC 29, *Coding of Audio, Picture, Multimedia and Hypermedia Information*.

This second edition cancels and replaces the first edition (ISO/IEC 15444-12:2004) which has been technically revised.

ISO/IEC 15444 consists of the following parts, under the general title *Information technology — JPEG 2000 image coding system*:

— *Part 1: Core coding system*

— *Part 2: Extensions*

— *Part 3: Motion JPEG 2000*

— *Part 4: Conformance testing*

— *Part 5: Reference software*

— *Part 6: Compound image file format*

— *Part 8: Secure JPEG 2000*

— *Part 9: Interactivity tools, APIs and protocols*

— *Part 11: Wireless JEP 2000*

— *Part 12: ISO base media file format*

The following parts are under preparation:

— *Part 10: Extensions for 3-D and floating point data*

This corrected version of ISO/IEC 15444-12:2005 (E) incorporates the following corrections:

— the Foreword has been altered to reflect that ISO/IEC 15444-11 is now published;

— in 8.16.2, 8.17.2.1, 8.17.3.1, 8.18.2 and 8.19.2, "•" has been replaced with "⩽".

Introduction

The ISO Base Media File Format is designed to contain timed media information for a presentation in a flexible, extensible format that facilitates interchange, management, editing, and presentation of the media. This presentation may be 'local' to the system containing the presentation, or may be via a network or other stream delivery mechanism.

The file structure is object-oriented; a file can be decomposed into constituent objects very simply, and the structure of the objects inferred directly from their type.

The file format is designed to be independent of any particular network protocol while enabling efficient support for them in general.

The ISO Base Media File Format is a base format for media file formats.

It is intended that the ISO Base Media File Format shall be jointly maintained by WG1 and WG11. Consequently, a subdivision of work created 15444-12 and 14496-12 in order to document the ISO Base Media File Format and to facilitate the joint maintenance.

This technically identical text is published as ISO/IEC 14496-12 for MPEG-4, and as ISO/IEC 15444-12 for JPEG 2000, and reference to this specification should be made accordingly. The recommendation is to reference one, for example ISO/IEC 14496-12, and append to the reference a parenthetical comment identifying the other, for example "(technically identical to ISO/IEC 15444-12)".

Information technology — JPEG 2000 image coding system —

Part 12:
ISO base media file format

1 Scope

This International Standard specifies the ISO base media file format, which is a general format forming the basis for a number of other more specific file formats. This format contains the timing, structure, and media information for timed sequences of media data, such as audio/visual presentations.

This part of ISO/IEC 14496 is applicable to MPEG-4, but its technical content is identical to that of ISO/IEC 15444-12, which is applicable to JPEG 2000.

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 639-2:1998: *Codes for the representation of names of languages — Part 2: Alpha-3 code*

ISO/IEC 11578:1996: *Information technology — Open Systems Interconnection — Remote Procedure Call (RPC)*

ISO/IEC 14496-1:2004: *Information technology — Coding of audio-visual objects — Part 1: Systems*

ISO/IEC 14496-10, *Information technology — Coding of audio-visual objects — Part 10: Advanced Video Coding*

ISO/IEC 14496-14, *Information technology — Coding of audio-visual objects — Part 14: MP4 file format*

ITU-T Rec.T.800 | ISO/IEC 15444-1: *Information technology — JPEG 2000 image coding system: Core coding system*

ISO/IEC 15444-3, *Information technology — JPEG 2000 image coding system: Motion JPEG 2000*

IETF RFC 3711, *"The Secure Real-time Transport Protocol"*, Baugher M. et al., March 2004.

SMIL 1.0 *"Synchronized Multimedia Integration Language (SMIL) 1.0 Specification"*, <http://www.w3.org/TR/REC-smil/>

3 Definitions

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

3.1
Box:

An object-oriented building block defined by a unique type identifier and length (called 'atom' in some specifications, including the first definition of MP4).

3.2
Chunk:

A contiguous set of samples for one track.

3.3
Container Box:

A box whose sole purpose is to contain and group a set of related boxes.

3.4
Hint Track:

A special track which does not contain media data. Instead it contains instructions for packaging one or more tracks into a streaming channel.

3.5
Hinter:

A tool that is run on a file containing only media, to add one or more hint tracks to the file and so facilitate streaming.

3.6
Movie Box:

A container box whose sub-boxes define the metadata for a presentation ('moov').

3.7
Media Data Box:

A container box which can hold the actual media data for a presentation ('mdat').

3.8
ISO Base Media File:

The name of the file format described in this specification.

3.9
Presentation:

One or more motion sequences (q.v.), possibly combined with audio.

3.10
Sample:

In non-hint tracks, a sample is an individual frame of video, a time-contiguous series of video frames, or a time-contiguous compressed section of audio. In hint tracks, a sample defines the formation of one or more streaming packets. No two samples within a track may share the same time-stamp.

3.11
Sample Description:

A structure which defines and describes the format of some number of samples in a track.

3.12

Sample Table:

A packed directory for the timing and physical layout of the samples in a track.

3.13

Track:

A collection of related samples (q.v.) in an ISO base media file. For media data, a track corresponds to a sequence of images or sampled audio. For hint tracks, a track corresponds to a streaming channel.

4 Object-structured File Organization

4.1 File Structure

Files are formed as a series of objects, called boxes in this specification. All data is contained in boxes; there is no other data within the file. This includes any initial signature required by the specific file format.

All object-structured files conformant to this section of this specification (all Object-Structured files) shall contain a File Type Box.

4.2 Object Structure

An object in this terminology is a box.

Boxes start with a header which gives both size and type. The header permits compact or extended size (32 or 64 bits) and compact or extended types (32 bits or full UUIDs). The standard boxes all use compact types (32-bit) and most boxes will use the compact (32-bit) size. Typically only the Media Data Box(es) need the 64-bit size.

The size is the entire size of the box, including the size and type header, fields, and all contained boxes. This facilitates general parsing of the file.

The definitions of boxes are given in the syntax description language (SDL) defined in MPEG-4 (see reference in clause 2). Comments in the code fragments in this specification indicate informative material.

The fields in the objects are stored with the most significant byte first, commonly known as network byte order or big-endian format.

```
aligned(8) class Box (unsigned int(32) boxtype,
         optional unsigned int(8)[16] extended_type) {
   unsigned int(32) size;
   unsigned int(32) type = boxtype;
   if (size==1) {
      unsigned int(64) largesize;
   } else if (size==0) {
      // box extends to end of file
   }
   if (boxtype==‘uuid’) {
      unsigned int(8)[16] usertype = extended_type;
   }
}
```

The semantics of these two fields are:

`size` is an integer that specifies the number of bytes in this box, including all its fields and contained boxes; if size is 1 then the actual size is in the field `largesize`; if size is 0, then this box is the last one in the file, and its contents extend to the end of the file (normally only used for a Media Data Box)

`type` identifies the box type; standard boxes use a compact type, which is normally four printable characters, to permit ease of identification, and is shown so in the boxes below. User extensions use an extended type; in this case, the type field is set to ‘uuid’.

Boxes with an unrecognized type shall be ignored and skipped.

Many objects also contain a version number and flags field:

```
aligned(8) class FullBox(unsigned int(32) boxtype, unsigned int(8) v, bit(24) f)
   extends Box(boxtype) {
   unsigned int(8)   version = v;
   bit(24)     flags = f;
}
```

The semantics of these two fields are:

`version` is an integer that specifies the version of this format of the box.
`flags` is a map of flags

Boxes with an unrecognized version shall be ignored and skipped.

4.3 File Type Box

4.3.1 Definition

Box Type: **`ftyp’**
Container: **File**
Mandatory: **Yes**
Quantity: **Exactly one**

A media-file structured to this part of this specification may be compatible with more than one detailed specification, and it is therefore not always possible to speak of a single ‘type’ or ‘brand’ for the file. This means that the utility of the file name extension and mime type are somewhat reduced.

This box must be placed as early as possible in the file (e.g. after any obligatory signature, but before any significant variable-size boxes such as a Movie Box, Media Data Box, or Free Space). It identifies which specification is the ‘best use’ of the file, and a minor version of that specification; and also a set of other specifications to which the file complies. Readers implementing this format should attempt to read files that are marked as compatible with any of the specifications that the reader implements. Any incompatible change in a specification should therefore register a new ‘brand’ identifier to identify files conformant to the new specification.

The minor version is informative only. It does not appear for compatible-brands, and must not be used to determine the conformance of a file to a standard. It may allow more precise identification of the major specification, for inspection, debugging, or improved decoding.

The type ‘isom’ (ISO Base Media file) is defined in this section of this specification, as identifying files that conform to the first version of ISO Base Media File Format.

More specific identifiers can be used to identify precise versions of specifications providing more detail. This brand should not be used as the major brand; this base file format should be derived into another specification to be used. There is therefore no defined normal file extension, or mime type assigned to this brand, nor definition of the minor version when ‘isom’ is the major brand.

Files would normally be externally identified (e.g. with a file extension or mime type) that identifies the ‘best use’ (major brand), or the brand that the author believes will provide the greatest compatibility.

The brand ‘iso2’ shall be used to indicate compatibility with this amended version of the ISO Base Media File Format; it may be used in addition to or instead of the ‘isom’ brand and the same usage rules apply. If used without the brand 'isom' identifying the first version of this specification, it indicates that support for some or all

of the technology introduced by this amendment is required, such as the functionality in sub-clauses [8.40] through [8.45], or the SRTP support in sub-clause [10], is required.

The brand ‘avc1’ shall be used to indicate that the file is conformant with the ‘AVC Extensions’ in sub-clause [8.40]. If used without other brands, this implies that support for those extensions is required. The use of ‘avc1’ as a major-brand may be permitted by specifications; in that case, that specification defines the file extension and required behavior.

If a Meta-box with an MPEG-7 handler type is used at the file level, then the brand ‘mp71’ should be a member of the compatible-brands list in the file-type box.

4.3.2 Syntax

```
aligned(8) class FileTypeBox
   extends Box(‘ftyp’) {
   unsigned int(32)  major_brand;
   unsigned int(32)  minor_version;
   unsigned int(32) compatible_brands[];    // to end of the box
}
```

4.3.3 Semantics

This box identifies the specifications to which this file complies.

Each brand is a printable four-character code, registered with ISO, that identifies a precise specification.

`major_brand` – is a brand identifier
`minor_version` – is an informative integer for the minor version of the major brand
`compatible_brands` – is a list, to the end of the box, of brands

5 Design Considerations

5.1 Usage

5.1.1 Introduction

The file format is intended to serve as a basis for a number of operations. In these various roles, it may be used in different ways, and different aspects of the overall design exercised.

5.1.2 Interchange

When used as an interchange format, the files would normally be self-contained (not referencing media in other files), contain only the media data actually used in the presentation, and not contain any information related to streaming. This will result in a small, protocol-independent, self-contained file, which contains the core media data and the information needed to operate on it.

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

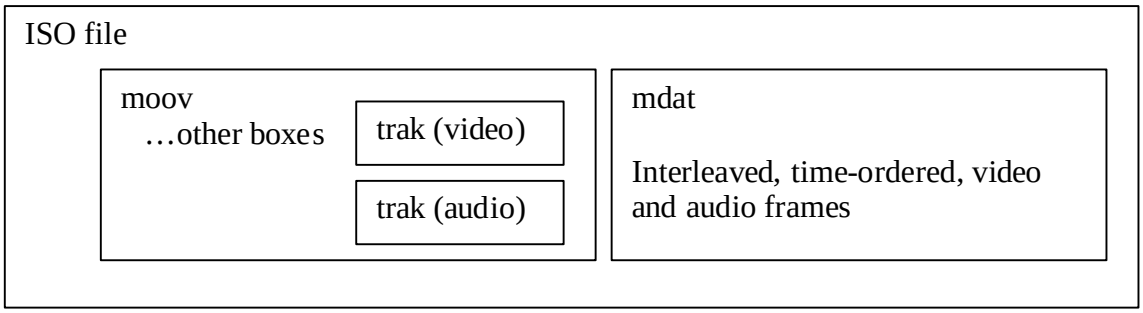

Figure 1 — Simple interchange file

5.1.3 Content Creation

During content creation, a number of areas of the format can be exercised to useful effect, particularly:

- the ability to store each elementary stream separately (not interleaved), possibly in separate files.
- the ability to work in a single presentation that contains media data and other streams (e.g. editing the audio track in the uncompressed format, to align with an already-prepared video track).

These characteristics mean that presentations may be prepared, edits applied, and content developed and integrated without either iteratively re-writing the presentation on disc – which would be necessary if interleave was required and unused data had to be deleted; and also without iteratively decoding and re-encoding the data – which would be necessary if the data must be stored in an encoded state.

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

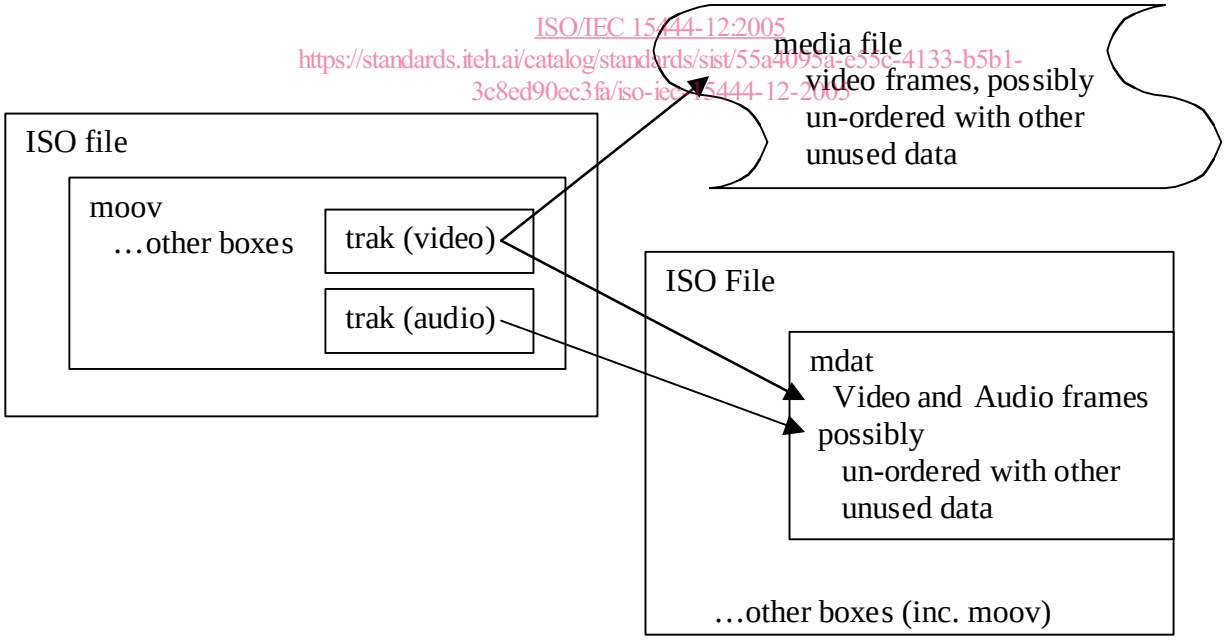

Figure 2 — Content Creation File

5.1.4 Preparation for streaming

When prepared for streaming, the file must contain information to direct the streaming server in the process of sending the information. In addition, it is helpful if these instructions and the media data are interleaved so that excessive seeking can be avoided when serving the presentation. It is also important that the original media data be retained unscathed, so that the files may be verified, or re-edited or otherwise re-used. Finally, it is

helpful if a single file can be prepared for more than one protocol, so differing servers may use it over disparate protocols.

5.1.5 Local presentation

'Locally' viewing a presentation (i.e. directly from the file, not over a streamed interconnect) is an important application; it is used when a presentation is distributed (e.g. on CD or DVD ROM), during the process of development, and when verifying the content on streaming servers. Such local viewing must be supported, with full random access. If the presentation is on CD or DVD ROM, interleave is important as seeking may be slow.

5.1.6 Streamed presentation

When a server operates from the file to make a stream, the resulting stream must be conformant with the specifications for the protocol(s) used, and should contain no trace of the file-format information in the file itself. The server needs to be able to random access the presentation. It can be useful to re-use server content (e.g. to make excerpts) by referencing the same media data from multiple presentations; it can also assist streaming if the media data can be on read-only media (e.g. CD) and not copied, merely augmented, when prepared for streaming.

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

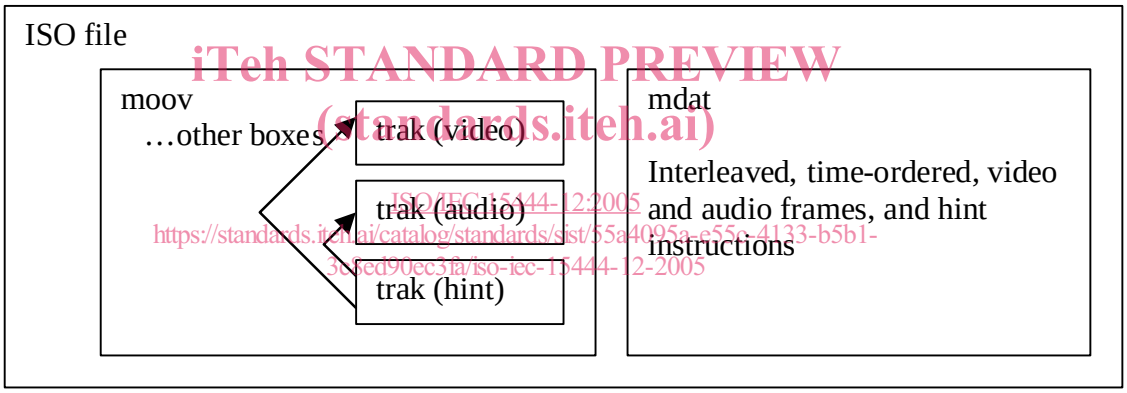

Figure 3 — Hinted Presentation for Streaming

5.2 Design principles

The file structure is object-oriented; a file can be decomposed into constituent objects very simply, and the structure of the objects inferred directly from their type.

Media-data is not 'framed' by the file format; the file format declarations that give the size, type and position of media data units are not physically contiguous with the media data. This makes it possible to subset the media-data, and to use it in its natural state, without requiring it to be copied to make space for framing. The metadata is used to describe the media data by reference, not by inclusion.

Similarly the protocol information for a particular streaming protocol does not frame the media data; the protocol headers are not physically contiguous with the media data. Instead, the media data can be included by reference. This makes it possible to represent media data in its natural state, not favoring any protocol. It also makes it possible for the same set of media data to serve for local presentation, and for multiple protocols.

The protocol information is built in such a way that the streaming servers need to know only about the protocol and the way it should be sent; the protocol information abstracts knowledge of the media so that the servers are, to a large extent, media-type agnostic. Similarly the media-data, stored as it is in a protocol-unaware fashion, enables the media tools to be protocol-agnostic.