INTERNATIONAL STANDARD

ISO/IEC 14496-15

First edition
2004-04-15

Information technology — Coding of audio-visual objects —

Part 15: Advanced Video Coding (AVC) file format

Technologies de l'information — Codage des objets audiovisuels —
Partie 15: Format de fichier de codage vidéo avancé (AVC)

Reference number
ISO/IEC 14496-15:2004(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.

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 14496-15 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*

Introduction

The Advanced Video Coding (AVC) standard, jointly developed by the ITU-T and ISO/IEC SC29/WG11 (MPEG), offers not only increased coding efficiency and enhanced robustness, but also many features for the systems that use it. To enable the best visibility of, and access to, those features, and to enhance the opportunities for the interchange and interoperability of media, this part of ISO/IEC 14496 defines a storage format for video streams compressed using AVC.

This part of ISO/IEC 14496 defines a storage format based on, and compatible with, the ISO Base Media File Format (ISO/IEC 14496-12 and ISO/IEC 15444-12), which is used by the MP4 file format (ISO/IEC 14496-14) and the Motion JPEG 2000 file format (ISO/IEC 15444-3) among others. This part of ISO/IEC 14496 enables AVC video streams to:

- be used in conjunction with other media streams, such as audio;
- be used in an MPEG-4 systems environment, if desired;
- be formatted for delivery by a streaming server, using hint tracks;
- inherit all the use cases and features of the ISO Base Media File Format on which MP4 and MJ2 are based.

This part of ISO/IEC 14496 may be used as a standalone specification; it specifies how AVC content shall be stored in an ISO Base Media File Format compliant format. However, it is normally used in the context of a specification, such as the MP4 file format, derived from the ISO Base Media File Format, that permits the use of AVC video.

The ISO Base Media File Format is becoming increasingly common as a general-purpose media container format for the exchange of digital media, and its use in this context should accelerate both adoption and interoperability.

Extensions to the ISO Base Media File Format are defined here to support the new systems aspects of the AVC codec.

Information technology — Coding of audio-visual objects —

Part 15:
Advanced Video Coding (AVC) file format

1 Scope

This part of ISO/IEC 14496 specifies the storage format for AVC (ISO/IEC 14496-10 | ITU-T Rec. H.264) video streams.

The storage of AVC content uses the existing capabilities of the ISO Base Media File Format but also defines extensions to support the following features of the AVC codec:

- **Switching pictures**: To enable switching between different coded streams and substitution of pictures within the same stream.
- **Sub-sequences and layers**: Provides a structuring of the dependencies of a group of pictures to provide for a flexible stream structure (e.g. in terms of temporal scalability and layering).
- **Parameter sets**: The sequence and picture parameter set mechanism decouples the transmission of infrequently changing information from the transmission of coded macroblock data. Each slice containing the coded macroblock data references the picture parameter set containing its decoding parameters. In turn, the picture parameter set references a sequence parameter set that contains sequence level decoding parameter information.

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 14496-10, *Information technology — Coding of audio-visual objects — Part 10: Advanced video coding* | ITU-T Rec. H.264, *Advanced video coding for generic audiovisual services*

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 Terms, definitions, symbols and abbreviated terms

3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 14496-1, ISO/IEC 14496-10 | ITU-T Rec. H.264 and the following apply.

3.1.1
parameter set
a sequence parameter set or a picture parameter set, as defined in ISO/IEC 14496-10

NOTE This term is used to refer to both types of parameter sets.

3.1.2
parameter set elementary stream
elementary stream containing samples made up of only sequence and picture parameter set NAL units synchronized with the video elementary stream

3.1.3
video elementary stream
elementary stream containing access units made up of NAL units for coded picture data

3.2 Symbols and abbreviated terms

AVC	Advanced Video Coding [ISO/IEC 14496-10]
HRD	Hypothetical Reference Decoder
IDR	Instantaneous Decoding Refresh
NAL	Network Abstraction Layer
PPS	Picture Parameter Set
SEI	Supplementary Enhancement Information
SPS	Sequence Parameter Set

4 Extensions to the ISO Base Media File Format

4.1 Introduction

This clause documents technical additions to the ISO Base Media File Format, which can be used when storing AVC streams. However, these additions could also be used by other media, if they are defined to use them. They are therefore documented here separately.

4.2 File identification

The brand ‘avc1’ shall be used to indicate that extensions conformant with this section are used in a file. The use of ‘avc1’ as a major-brand may be permitted by specifications; in that case, that specification defines the file extension and required behaviour.

4.3 Independent and Disposable Samples Box

Box Types: ‘sdtp’
Container: Sample Table Box (‘stbl’)
Mandatory: No
Quantity: Exactly one

This optional table answers three questions about sample dependency:

1) Does this sample depend on others (is it an I-picture)?

2) Do no other samples depend on this one?

3) Does this sample contain multiple (redundant) encodings of the data at this time-instant (possibly with different dependencies)?

In the absence of this table:

1) the sync sample table answers the first question; in most video codecs, I-pictures are also sync points,

2) the dependency of other samples on this one is unknown,

3) the existence of redundant coding is unknown.

When performing 'trick' modes, such as fast-forward, it is possible to use the first piece of information to locate independently decodable samples. Similarly, when performing random access, it may be necessary to locate the previous sync point or random access recovery point, and roll-forward from the sync point or the pre-roll starting point of the random access recovery point to the desired point. While rolling forward, samples on which no others depend need not be retrieved or decoded.

The value of 'sample-is-depended-on' is independent of the existence of redundant codings. However, a redundant coding may have different dependencies from the primary coding; if redundant codings are available, the value of 'sample-depends-on' documents only the primary coding.

The size of the table, `sample_count` is taken from the `sample_count` in the Sample Size Box ('stsz') or Compact Sample Size Box ('stz2').

4.3.1.1 Syntax

```
aligned(8) class SampleDependencyTypeBox
   extends FullBox('sdtp', version = 0, 0) {
   for (i=0; i < sample_count; i++){
      unsigned int(2) reserved = 0;
      unsigned int(2) sample-depends-on;
      unsigned int(2) sample-is-depended-on;
      unsigned int(2) sample-has-redundancy;
   }
}
```


4.3.1.2 Semantics

`sample-depends-on` takes one of the following four values:
0: the dependency of this sample is unknown;
1: this sample does depend on others (not an I picture);
2: this sample does not depend on others (I picture);
3: reserved.

`sample-is-depended-on` takes one of the following four values:
0: the dependency of other samples on this sample is unknown;
1: other samples depend on this one (not disposable);
2: no other sample depends on this one (disposable);
3: reserved.

`sample-has-redundancy` takes one of the following four values:
0: it is unknown whether there is redundant coding in this sample;
1: there is redundant coding in this sample;
2: there is no redundant coding in this sample;
3: reserved.

4.4 Sample groups

4.4.1 Introduction

This clause specifies a generic mechanism for representing a partition of the samples in a track. A *sample grouping* is an assignment of each sample in a track to be a member of one *sample group,* based on a

grouping criterion. A sample group in a sample grouping is not limited to being contiguous samples and may contain non-adjacent samples. As there may be more than one sample grouping for the samples in a track, each sample grouping has a type field to indicate the type of grouping. For example, a file might contain two sample groupings for the same track: one based on an assignment of sample to layers and another to sub-sequences.

Sample groupings are represented by two linked data structures: (1) a `SampleToGroup` box represents the assignment of samples to sample groups; (2) a `SampleGroupDescription` box contains a *sample group entry* for each sample group describing the properties of the group. There may be multiple instances of the `SampleToGroup` and `SampleGroupDescription` boxes based on different grouping criteria. These are distinguished by a type field used to indicate the type of grouping.

One example of using these tables is to represent the assignments of samples to *layers.* In this case each sample group represents one layer, with an instance of the `SampleToGroup` box describing which layer a sample belongs to. For more details, please refer to 5.3.12

4.4.2 SampleToGroup Box

4.4.2.1 Definition

Box Type: ‘sbgp’
Container: Sample Table Box (‘stbl’)
Mandatory: No
Quantity: Zero or more.

This table can be used to find the group that a sample belongs to and the associated description of that sample group. The table is compactly coded with each entry giving the index of the first sample of a run of samples with the same sample group descriptor. The sample group description ID is an index that refers to a `SampleGroupDescription` box, which contains entries describing the characteristics of each sample group.

There may be multiple instances of this box if there is more than one sample grouping for the samples in a track. Each instance of the `SampleToGroup` box has a type code that distinguishes different sample groupings. Within a track, there shall be at most one instance of this box with a particular grouping type. The associated `SampleGroupDescription` shall indicate the same value for the grouping type.

4.4.2.2 Syntax

```
aligned(8) class SampleToGroupBox
   extends FullBox('sbgp', version = 0, 0)
{
   unsigned int(32)  grouping_type;
   unsigned int(32)  entry_count;
   for (i=1; i <= entry_count; i++)
   {
      unsigned int(32)  sample_count;
      unsigned int(32)  group_description_index;
   }
}
```

4.4.2.3 Semantics

`version` is an integer that specifies the version of this box.

`grouping_type` is an integer that identifies the type (i.e. criterion used to form the sample groups) of the sample grouping and links it to its sample group description table with the same value for grouping type. At most one occurrence of this box with the same value for `grouping_type` shall exist for a track.

`entry_count` is an integer that gives the number of entries in the following table.

`sample_count` is an integer that gives the number of consecutive samples with the same sample group descriptor.

`group_description_index` is an integer that gives the index of the sample group entry which describes the samples in this group. The index ranges from 1 to the number of sample group entries in the `SampleGroupDescription` Box, or takes the value 0 to indicate that this sample is a member of no group of this type.

4.4.3 SampleGroupDescription Box

4.4.3.1 Definition

Box Types: 'sgpd'
Container: Sample Table Box ('stbl')
Mandatory: No
Quantity: Zero or more, with one for each SampleToGroup Box.

This description table gives information about the characteristics of sample groups. The descriptive information is any other information needed to define or characterize the sample group.

There may be multiple instances of this box if there is more than one sample grouping for the samples in a track. Each instance of the `SampleGroupDescription` box has a type code that distinguishes different sample groupings. Within a track, there shall be at most one instance of this box with a particular grouping type. The associated `SampleToGroup` shall indicate the same value for the grouping type.

The information is stored in the sample group description box after the entry-count. An abstract entry type is defined and sample groupings shall define derived types to represent the description of each sample group. For video tracks, an abstract `VisualSampleGroupEntry` is used with similar types for audio and hint tracks.

4.4.3.2 Syntax

```
// Sequence Entry
abstract class SampleGroupDescriptionEntry (unsigned int(32) handler_type)
{
}
```



```
// Visual Sequence
abstract class VisualSampleGroupEntry (type) extends SampleGroupDescriptionEntry
(type)
{
}
```

```
// Audio Sequences
abstract class AudioSampleGroupEntry (type) extends SampleGroupDescriptionEntry
(type)
{
}
```

```
aligned(8) class SampleGroupDescriptionBox (unsigned int(32) handler_type)
   extends FullBox('sgpd', 0, 0){
   unsigned int(32) grouping_type;
   unsigned int(32) entry_count;
   int i;
   for (i = 1 ; i <= entry_count ; i++){
      switch (handler_type){
         case ‘vide’: // for video tracks
            VisualSampleGroupEntry ();
            break;
         case ‘soun’: // for audio tracks
            AudioSampleGroupEntry();
            break;
         case ‘hint’: // for hint tracks
            HintSampleGroupEntry();
            break;
      }
   }
}
```

4.4.3.3 Semantics

`version` is an integer that specifies the version of this box.

`grouping_type` is an integer that identifies the `SampleToGroup` box that is associated with this sample group description.

`entry_count` is an integer that gives the number of entries in the following table.

4.5 Random access recovery points

In some coding systems it is possible to random access into a stream and achieve correct decoding after having decoded a number of samples. This is known as gradual decoding refresh. For example, in video, the encoder might encode intra-coded macroblocks in the stream, such that it knows that within a certain period the entire picture consists of pixels that are only dependent on intra-coded macroblocks supplied during that period.

Samples for which such gradual refresh is possible are marked by being a member of this group. The definition of the group allows the marking to occur at either the beginning of the period or the end. However, when used with a particular media type, the usage of this group may be restricted to marking only one end (i.e. restricted to only positive or negative roll values). A roll-group is defined as that group of samples having the same roll distance.

4.5.1 Syntax

```
class VisualRollRecoveryEntry() extends VisualSampleGroupEntry (’roll’)
{
   signed int(16) roll-distance;
}
```

4.5.2 Semantics

`roll-distance` is a signed integer that gives the number of samples that must be decoded in order for a sample to be decoded correctly. A positive value indicates the number of samples after the sample that is a group member that must be decoded before recovery is complete. A negative value indicates the number of samples before the sample that is a group member that must be decoded in order for recovery to be complete at the marked sample. The value zero must not be used; the sync sample table documents random access points for which no recovery roll is needed.