

ISO/IEC JTC 1

Secretariat: **ANSI**

Voting begins on:  
**2004-09-30**

Voting terminates on:  
**2004-11-30**

---

---

**Information technology — JPEG 2000  
image coding system —**

**Part 12:  
ISO base media file format**

**AMENDMENT 1: File format extensions and  
guidelines**

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

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

*ISO/IEC 15444-12:2004/FDAM 1*

*<https://standards.iteh.org/en/standards/iso-iec-15444-12-2004-fdam-1>  
Partie 12: Format ISO de base pour les fichiers medias*

*AMENDEMENT 1: Extensions de format de fichier et lignes directrices*

Please see the administrative notes on page iii

RECIPIENTS OF THIS DRAFT ARE INVITED TO SUBMIT, WITH THEIR COMMENTS, NOTIFICATION OF ANY RELEVANT PATENT RIGHTS OF WHICH THEY ARE AWARE AND TO PROVIDE SUPPORTING DOCUMENTATION.

IN ADDITION TO THEIR EVALUATION AS BEING ACCEPTABLE FOR INDUSTRIAL, TECHNOLOGICAL, COMMERCIAL AND USER PURPOSES, DRAFT INTERNATIONAL STANDARDS MAY ON OCCASION HAVE TO BE CONSIDERED IN THE LIGHT OF THEIR POTENTIAL TO BECOME STANDARDS TO WHICH REFERENCE MAY BE MADE IN NATIONAL REGULATIONS.



Reference number  
ISO/IEC 15444-12:2004/FDAM 1:2004(E)

**PDF disclaimer**

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

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

[ISO/IEC 15444-12:2004/FDAM 1](https://standards.iteh.ai/catalog/standards/sist/6ab44e45-9bf0-4ed6-9e4d-bec814545d97/iso-iec-15444-12-2004-fdam-1)

<https://standards.iteh.ai/catalog/standards/sist/6ab44e45-9bf0-4ed6-9e4d-bec814545d97/iso-iec-15444-12-2004-fdam-1>

**Copyright notice**

This ISO document is a Draft International Standard and is copyright-protected by ISO. Except as permitted under the applicable laws of the user's country, neither this ISO draft nor any extract from it may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, photocopying, recording or otherwise, without prior written permission being secured.

Requests for permission to reproduce should be addressed to either ISO at the address below or ISO's member body in the country of the requester.

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](mailto:copyright@iso.org)  
Web [www.iso.org](http://www.iso.org)

Reproduction may be subject to royalty payments or a licensing agreement.

Violators may be prosecuted.

In accordance with the provisions of Council Resolution 21/1986, this document is **circulated in the English language only**.

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

[ISO/IEC 15444-12:2004/FDAM 1](https://standards.iteh.ai/catalog/standards/sist/6ab44e45-9bf0-4ed6-9e4d-bec814545d97/iso-iec-15444-12-2004-fdam-1)

<https://standards.iteh.ai/catalog/standards/sist/6ab44e45-9bf0-4ed6-9e4d-bec814545d97/iso-iec-15444-12-2004-fdam-1>

## 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.

Amendment 1 to ISO/IEC 15444-12:2004 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

ITEH STANDARD PREVIEW

(standards.iteh.ai)

[ISO/IEC 15444-12:2004/FDAM 1](https://standards.iteh.ai/catalog/standards/sist/6ab44e45-9bf0-4ed6-9e4d-bec814545d97/iso-iec-15444-12-2004-fdam-1)

<https://standards.iteh.ai/catalog/standards/sist/6ab44e45-9bf0-4ed6-9e4d-bec814545d97/iso-iec-15444-12-2004-fdam-1>

# Information technology — JPEG 2000 image coding system —

## Part 12: ISO base media file format

### AMENDMENT 1: File format extensions and guidelines

*In clause 2, add the following references:*

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*

ISO/IEC 15444-3, *Information technology – JPEG 2000 image coding system – Part 3: 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/>>

*In 4.3.1, Definition of File Type Box, replace:*

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

*with:*

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.

*Add at the end of 4.3.1:*

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 subclauses [8.40] through [8.45], or the SRTP support in sub-clause [0], 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.

*In 4.3.3, delete the sentence:*

"Only one brand is defined here..."

In 6.1.2, add “Meta-data, “ after “Move Fragments, “.

In 6.2.3, insert the following as point 8 in the recommendations:

8) It is **recommended** that the progressive download information box be placed as early as possible in files, for maximum utility.

Replace Table 1 as follows (correctly cross-referenced):

ftyp				*	4.3	file type and compatibility	
pdin					8.43	progressive download information	
moov				*	8.1	container for all the metadata	
	mvhd			*	8.3	movie header, overall declarations	
	trak			*	8.4	container for an individual track or stream	
		tkhd		*	8.5	track header, overall information about the track	
		tref			8.6	track reference container	
		edts			8.25	edit list container	
			elst		8.26	an edit list	
		mdia		*	8.7	container for the media information in a track	
			mdhd	*	8.8	media header, overall information about the media	
			hdlr	*	8.9	handler, declares the media (handler) type	
			minf	*	8.10	media information container	
			vmhd		8.11.2	video media header, overall information (video track only)	
			smhd		8.11.3	sound media header, overall information (sound track only)	
			hmhd		8.11.4	hint media header, overall information (hint track only)	
			nmhd		8.11.5	Null media header, overall information (some tracks only)	
			dinf	*	8.12	data information box, container	
				dref	*	8.13	data reference box, declares source(s) of media data in track
			stbl	*	8.14	sample table box, container for the time/space map	
				stsd	*	8.16	sample descriptions (codec types, initialization etc.)
				stts	*	8.15.2	(decoding) time-to-sample
				ctts		8.15.3	(composition) time to sample
				stsc	*	8.18	sample-to-chunk, partial data-offset information
				stsz		8.17.2	sample sizes (framing)
				stz2		8.17.3	compact sample sizes (framing)
				stco	*	8.19	chunk offset, partial data-offset information
				co64		8.19	64-bit chunk offset
				stss		8.20	sync sample table (random access points)
				stsh		8.21	shadow sync sample table
				padb		8.23	sample padding bits
				stdp		8.22	sample degradation priority
				sdtg		8.40.2	independent and disposable samples
				sbgp		8.40.3.2	sample-to-group
				sgpd		8.40.3.3	sample group description
				subs		8.42	sub-sample information
	mvex					8.29	movie extends box
		mehd				8.30	movie extends header box
					*	8.31	track extends defaults
						8.45.4	IPMP Control Box
moof						8.32	movie fragment
		mfhd		*		8.33	movie fragment header

	traf					8.34	track fragment
		tfhd			*	8.35	track fragment header
		trun				8.36	track fragment run
		sdtp				8.40.2	independent and disposable samples
		sbgp				8.40.3.2	sample-to-group
		subs				8.42	sub-sample information
mfra						8.37	movie fragment random access
	tfra					8.38	track fragment random access
	mfro				*	8.39	movie fragment random access offset
mdat						8.2	media data container
free						8.24	free space
skip						8.24	free space
	udta					8.27	user-data
	cppt					8.28	copyright etc.
meta						8.44.1	metadata
	hdlr				*	8.9	handler, declares the metadata (handler) type
	dinf					8.12	data information box, container
	dref					8.13	data reference box, declares source(s) of metadata items
	ipmc					8.45.4	IPMP Control Box
	iloc					8.44.3	item location
	ipro					8.44.5	item protection
	sinf					8.45.1	protection scheme information box
		frma				8.45.2	original format box
		imif				8.45.3	IPMP Information box
		schm				8.45.5	scheme type box
		schI				8.45.6	scheme information box
	iinf					8.44.6	item information
	xml					8.44.2	XML container
	bxml					8.44.2	binary XML container
	pitm					8.44.4	primary item reference

<https://standards.iteh.ai/catalog/standards/sist/6ab44e45-9bf0-4ed6-9e4d-bec814545d97/iso-iec-15444-12-2004-fdam-1>

In 8.2.1, Media Data Box, add: [bec814545d97/iso-iec-15444-12-2004-fdam-1](https://standards.iteh.ai/catalog/standards/sist/6ab44e45-9bf0-4ed6-9e4d-bec814545d97/iso-iec-15444-12-2004-fdam-1)

“, and the item location box, subclause 8.44.3” before the closing parenthesis at the end of the first paragraph.

In 8.6.3, Track Reference Box Semantics, after “The Track Reference Box contains track reference type boxes” insert:

track\_ID is an integer that provides a reference from the containing track to another track in the presentation. track\_IDs are never re-used and cannot be equal to zero.

In 8.9.1, Definition of Handler Reference Box, change:

**Container: Media Box ('mdia')**

to:

**Container: Media Box ('mdia') or Meta Box ('meta')**

Add at the end of 8.9.1:

This box when present within a Meta Box, declares the structure or format of the 'meta' box contents.

In 8.9.3, change:

handler\_type is an integer containing one of the following

to:

handler\_type when present in a media box, is an integer containing one of the following

and add before documentation of the 'name' field:

handler\_type when present in a meta box, contains an appropriate value to indicate the format of the meta box contents

In 8.12.1, Definition of Data information box, change:

**Box Type:** 'dinf'  
**Container:** Media Information Box ('minf')  
**Mandatory:** Yes  
**Quantity:** Exactly one

to:

**Box Type:** 'dinf'  
**Container:** Media Information Box ('minf') or Meta Box ('meta')  
**Mandatory:** Yes (required within 'minf' box) and No (optional within 'meta' box)  
**Quantity:** Exactly one

THIS STANDARD PREVIEW  
(standards.iteh.ai)

[ISO/IEC 15444-12:2004/FDAM 1](#)

In 8.24.1, Definition of Free Space Box, change: [catalog/standards/sist/6ab44e45-9bf0-4ed6-9e4d-bec814545d97/iso-iec-15444-12-2004-fdam-1](#)

**Container:** File

to:

**Container:** File or other box

In 8.26.2, Edit List Box Syntax, change:

```
for (i=1; i ≤ entry_count; i++) {
```

to:

```
for (i=1; i <= entry_count; i++) {
```

Change:

```
int(16) media_rate_fraction;
```

to:

```
int(16) media_rate_fraction = 0;
```



In 8.31.1, Definition of Track Extends Box, replace:

The sample flags field in sample fragments (default\_sample\_flags here and in a Track Fragment Header Box, and sample\_flags and first\_sample\_flags in a Track Fragment Run Box) is coded as a 32-bit value. It has the following structure:

```
bit(12) reserved=0;
bit(3)  sample_padding_value;
bit(1)  sample_is_difference_sample;
        // i.e. when 1 signals a non-key or non-sync sample
unsigned int(16) sample_degradation_priority;
```

with:

The sample flags field in sample fragments (default\_sample\_flags here and in a Track Fragment Header Box, and sample\_flags and first\_sample\_flags in a Track Fragment Run Box) is coded as a 32-bit value. It has the following structure:

```
bit(6)  reserved=0;
unsigned int(2) sample_depends_on;
unsigned int(2) sample_is_depended_on;
unsigned int(2) sample_has_redundancy;
bit(3)  sample_padding_value;
bit(1)  sample_is_difference_sample;
        // i.e. when 1 signals a non-key or non-sync sample
unsigned int(16) sample_degradation_priority;
```

(standards.iteh.ai)

The sample\_depends\_on, sample\_is\_depended\_on and sample\_has\_redundancy values are defined as documented in the Independent and Disposable Samples Box.

<https://standards.iteh.ai/catalog/standards/sist/6ab44e45-9bf0-4ed6-9e4d-bec814545d97/iso-iec-15444-12-2004-fdam-1>

Insert the following subclauses after 8.39 (i.e. starting with the following subclause numbered as 8.40).

## 8.40 AVC Extensions to the ISO Base Media File Format

### 8.40.1 Introduction

This section documents technical additions to the ISO Base Media File Format originally designed for AVC support, but which are more generally applicable.

### 8.40.2 Independent and Disposable Samples Box

#### 8.40.2.1 Definition

Box Types: 'sntp'  
 Container: Sample Table Box ('stbl')  
 Mandatory: No  
 Quantity: Zero or 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’).

### 8.40.2.2 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;
  }
}
```


  
<https://standards.iteh.ai/catalog/standards/sist/6ab44e45-9bf0-4ed6-9e4d-bec814545d97/iso-iec-15444-12-2004-fdam-1>

### 8.40.2.3 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

## 8.40.3 Sample Groups

### 8.40.3.1 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 `SampleToGroupBox` 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 `SampleToGroupBox` 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 `SampleToGroupBox` describing which layer a sample belongs to.

### 8.40.3.2 SampleToGroupBox

#### 8.40.3.2.1 Definition

Box Type: 'sbgp'  
 Container: Sample Table Box ('stbl') or Track Fragment Box ('traf')  
 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 `SampleToGroupBox` 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.

#### 8.40.3.2.2 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;
  }
}
```

#### 8.40.3.2.3 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. If the sum of the sample count in this box is less than the total sample count, then the reader should effectively extend it with an entry that associates the remaining samples with no group.

It is an error for the total in this box to be greater than the `sample_count` documented elsewhere, and the reader behavior would then be undefined.

`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.

### 8.40.3.3 Sample Group Description Box

#### 8.40.3.3.1 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.

*Note:* the base classes for sample group description entries are not boxes and therefore no size is signaled. When defining derived classes, ensure either that they have a fixed size, or that the size is explicitly indicated with a length field. An implied size (e.g. achieved by parsing the data) is not recommended as this makes scanning the array difficult.

#### 8.40.3.3.2 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;
    }
  }
}

```

### 8.40.3.3.3 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.

### 8.40.3.4 Representation of group structures in Movie Fragments

Support for new `SampleGroup` structures within Movie fragments is provided by the use of the `SampleToGroup` Box with the container for this Box being the `Track Fragment Box` ('traf'). The definition, syntax and semantics of this Box is as specified in subclause 8.40.3.2.

The `SampleToGroup` Box can be used to find the group that a sample in a track fragment 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 and present in the `SampleTableBox`.

There may be multiple instances of the `SampleToGroup` Box if there is more than one sample grouping for the samples in a track fragment. Each instance of the `SampleToGroup` Box has a type code that distinguishes different sample groupings. The associated `SampleGroupDescription` shall indicate the same value for the grouping type.

The total number of samples represented in any `SampleToGroup` Box in the track fragment must match the total number of samples in all the track fragment runs. Each `SampleToGroup` Box documents a different grouping of the same samples.

## 8.40.4 Random Access Recovery Points

### 8.40.4.1 Definition

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,