

Encapsulation of Dirac in Ogg

v1.0 2008-10-08T13:12:37+0000

British Broadcasting Corporation

Contents	Page
1 Scope	1
2 Normative references	1
3 Terms and definitions	1
4 Logical stream identification	2
5 Logical stream termination	2
6 Storage of Dirac	2
6.1 Definition of a packet containing Dirac	2
6.2 Page encapsulation of packets containing Dirac	2
6.3 Definition of granule_position	2
Annex A (informative) Decomposition of granule_position	4
Annex B (informative) How to seek	5

Foreword

The British Broadcasting Corporation is a public service broadcaster based in the United Kingdom. Its network includes several TV and radio channels as well as a popular Internet web site.

The BBC's research department (formerly BBC Research & Development) has invented a video compression system called Dirac that is well suited to a large range of applications; from low bitrates (e.g. web streaming) to high and very high bitrates (e.g. broadcasting and video production).

A constrained profile of Dirac aimed at video production applications is being standardised by the SMPTE as (S2042) VC-2.

Further information on the activities of the BBC's research department are presented at: `<http://www.bbc.co.uk/rd/>`

Introduction

This document addresses the need to store Dirac coded bytestreams in the Ogg file format.

Encapsulation of Dirac in Ogg

1 Scope

This specification defines the semantics required to reliably encapsulate Dirac coded bytestreams in Ogg.

2 Normative references

The following normative documents contain provisions which, through reference in this text, constitute provisions of this Encapsulation of Dirac in Ogg. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this Encapsulation of Dirac in Ogg are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references, the latest edition of the normative document referred to applies. Members of ISO and IEC maintain registers of currently valid International Standards.

RFC 3533, *The Ogg Encapsulation Format Version 0*

Dirac Video Specification, <<http://www.diracvideo.org/specification>>

3 Terms and definitions

3.1

Data Unit

An object in a Dirac bytestream, delimited by a `parse_info_header`

3.2

Encapsulation Unit

One or more Dirac Data Units as per the rules in clause 6.1

4 Logical stream identification

The BOS page of a logical stream containing Dirac shall commence with a Dirac Sequence Header data unit. This should be followed by a Dirac End of Sequence data unit.

NOTE This implies that the first four bytes of the Ogg packet payload are “BBCD”.

NOTE The presence of a Dirac End of Sequence data unit guards against poor implementations of streaming servers that may join a client to a stream at an inappropriate point.

5 Logical stream termination

The EOS page of a logical stream containing Dirac shall terminate with a Dirac End-of-sequence data unit.

6 Storage of Dirac

This clause specifies the sample structure used to store Dirac bytestreams inside the Ogg file format.

6.1 Definition of a packet containing Dirac

The unit of encapsulation for Ogg is the packet; a packet of Dirac shall contain:

— zero or more non-picture Dirac data units

followed by a single:

— Dirac picture data unit

OR — Dirac end-of-sequence data unit

NOTE It follows that no Dirac data unit shall span multiple packets.

NOTE It is advisable that, the first packet after the logical stream BOS page contains an additional Dirac Sequence Header data unit to facilitate random access.

6.2 Page encapsulation of packets containing Dirac

In a logical stream of Dirac, an Ogg page should not terminate multiple Ogg packets.

NOTE It is advisable to flush the Ogg page after each encapsulation unit

6.3 Definition of granule_position

This sub-clause describes the mapping of granule_position in logical streams containing Dirac to a unique picture presentation time. The granule_position applies to the picture contained within the first packet that terminates in an Ogg page; any subsequent packets that terminate within the same page do not have a granule_position. If no packet terminates within the page, the granule_position has the value -1 .

The granule_position is composed of the following:

pt Picture number (in display order). When picture_coding_mode = 0(progressive), *pt* increments by two for each picture in display order. When picture_coding_mode = 1(interlace), *pt* increments by one for each field in display order.

It is permissible for there to be a constant offset between *pt* and the picture number found within the Dirac picture data unit.

Encapsulation of Dirac in Ogg

It is advised against any muxer rewriting the `picture_number` within the payload.

dist Number of pictures in coded order (equal to number of packets) since an appropriate sync point that allows for correct decoding of *pt*. This field is split into two parts, `dist_h` and `dist_l`. $dist = (dist_h \ll 8) | dist_l$

delay Delay (in pictures) between decoding time and presentation time of *pt*.

These values shall be packed into `granule_position` as follows:

```

6          3 3 2      2 2
3          1 0 9      2 1          9 8 7      0
+-----+-----+-----+-----+
|          pt - delay          |0| dist_h |      delay      |0| dist_l |
+-----+-----+-----+-----+
|<----- high_word ----->|<----- low_word ----->|
granule_shift = 22;
```


Annex A (informative) Decomposition of granule_position

The following example GOP is used in illustrating decoder use-cases of granule_position:

dt is the decode time.

GOP	I	P	B	B	I	B	B	P	B	B	I
dt	-1	0	1	2	3	4	5	6	7	8	9
pt	0	3	1	2	6	4	5	9	7	8	10
$delay$	1	3	0	0	3	0	0	3	0	0	1
$dist$	0	1	2	3	0	5	6	3	4	5	0

To decode granule_position using previously defined terms:

```
 $pt \leftarrow (\text{high\_word} + \text{low\_word}) \gg 9$   
 $dist\_h \leftarrow (\text{granule\_position} \gg 22) \& 0\text{xff}$   
 $dist\_l \leftarrow (\text{granule\_position}) \& 0\text{xff}$   
 $dist \leftarrow (dist\_h \ll 8) | dist\_l$   
 $delay \leftarrow (\text{granule\_position} \gg 9) \& 0\text{x1fff}$   
 $dt \leftarrow pt - delay$ 
```

Annex B
(informative)
How to seek

To find a the point in the stream with a particular presentation time, a search using `granule_position` will find a point in the stream earlier than intended (due to picture reordering).

To find the correct picture, a final search would be required using `high_word + low_word`.

How to find the appropriate sync point for a particular picture: the dt of the sync point is: `dt - dist`; this may be converted to the higher order bits of `granule_position`: `(dt - dist) >> 31`; the lower order bits are undefined.