Encapsulation of Opus in ISO Base Media File Format
Version 0.6.8 (incomplete)

Table of Contents

1 Scope

This specification specifies the fundamental way of the encapsulation of Opus coded bitstreams in ISO Base Media file format and its derivatives. The encapsulation of Opus coded bitstreams in QuickTime file format is outside the scope of this specification.

2 Normative References

[1] ISO/IEC 14496-12:2015 Corrected version
Information technology — Coding of audio-visual objects — Part 12: ISO base media file format

[2] RFC 6716
Definition of the Opus Audio Codec

[3] draft-ietf-codec-oggopus-06
Ogg Encapsulation for the Opus Audio Codec

3 Terms and Definitions

3.1 active track
enabled track from the non-alternate group or selected track from alternate group

3.2 actual duration
duration constructed from valid samples

3.3 edit
entry in the Edit List Box

3.4 padded samples
PCM samples after decoding Opus sample(s) which are not valid samples
An Opus bitstream always contains them partially at the beginning and may contain them in part at the end, as long as not physically removed yet at the beginning and/or the end.

3.5 priming samples
padded samples at the beginning of the Opus bitstream

3.6 sample-accurate
for any PCM sample, a timestamp exactly matching its sampling timestamp is present in the media timeline.

3.7 valid samples
PCM samples after decoding Opus sample(s) corresponding to input PCM samples

4 Design Rules of Encapsulation

4.1 File Type Indentification
This specification does not define any brand to declare files are conformant to this specification. However, files conformant to this specification shall contain at least one brand, which supports the requirements and the requirements described in this clause without contradiction, in the compatible brands list of the File Type Box.
As an example, the minimal support of the encapsulation of Opus bitstreams in ISO Base Media file format requires the 'iso2' brand in the compatible brands list since support of roll groups is required.

4.2 Overview of Track Structure
This clause summarizes requirements of the encapsulation of Opus coded bitstream as media data in audio tracks in file formats compliant with the ISO Base Media File Format. The details are described in clauses after this clause.

+ The handler_type field in the Handler Reference Box shall be set to 'soun'.

+ The Media Information Box shall contain the Sound Media Header Box.
+ The codingname of the sample entry is 'Opus'.
    This specification does not define any encapsulation using MP4AudioSampleEntry with objectTypeIndication specified by the MPEG-4 Registration Authority (http://www.mp4ra.org/).
    See 4.3.1 Sample entry format to get the details about the sample entry.
+ The 'dOps' box is added to the sample entry to convey initializing information for the decoder.
    See 4.3.2 Opus Specific Box to get the details.
+ An Opus sample is exactly one Opus packet for each of different Opus bitstreams.
    See 4.3.3 Sample format to get the details.
+ Every Opus sample is a sync sample but requires pre-roll for every random access to get correct output.
    See 4.3.6 Random Access to get the details.

4.3 Definitions of Opus sample

4.3.1 Sample entry format

For any track containing Opus bitstreams, at least one sample entry describing corresponding Opus bitstream shall be present inside the Sample Table Box. This version of the specification defines only one sample entry format named OpusSampleEntry whose codingname is 'Opus'. This sample entry includes exactly one Opus Specific Box defined in 4.3.2 as a mandatory box and indicates that Opus samples described by this sample entry are stored by the sample format described in 4.3.3.

The syntax and semantics of the OpusSampleEntry is shown as follows.

```
class OpusSampleEntry() extends AudioSampleEntry ('Opus'){
    OpusSpecificBox();
}
```

+ channelcount:
    The channelcount field shall be set to the sum of the total number of Opus bitstreams and the number of Opus bitstreams producing two channels. This value is indentical with (M+N), where M is the value of the *Coupled Stream Count* field and N is the value of the *Stream Count* field in the *Channel Mapping Table* in the identification header defined in Ogg Opus [3].
+ samplesize:
    The samplesize field shall be set to 16.
+ samplerate:
    The samplerate field shall be set to 48000<<16.
+ OpusSpecificBox
    This box contains initializing information for the decoder as defined in 4.3.2.

4.3.2 Opus Specific Box

Exactly one Opus Specific Box shall be present in each OpusSampleEntry.
The Opus Specific Box contains the Version field and this specification defines version 0 of this box.
If incompatible changes occured in the fields after the Version field within the OpusSpecificBox in the future versions of this specification, another version will be defined.
This box refers to Ogg Opus [3] at many parts but all the data are stored as big-endian format.

The syntax and semantics of the Opus Specific Box is shown as follows.

```
class ChannelMappingTable (unsigned int(8) OutputChannelCount){
    unsigned int(8) StreamCount;
    unsigned int(8) CoupledCount;
    unsigned int(8 * OutputChannelCount) ChannelMapping;
}

aligned(8) class OpusSpecificBox extends Box('dOps'){
    unsigned int(8) Version;
    unsigned int(8) OutputChannelCount;
    unsigned int(16) PreSkip;
    unsigned int(32) InputSampleRate;
    signed int(16) OutputGain;
    unsigned int(8) ChannelMappingFamily;
    if (ChannelMappingFamily != 0) {
        ChannelMappingTable(OutputChannelCount);
    }
}
```

+ Version:
    The Version field shall be set to 0.
    In the future versions of this specification, this field may be set to other values. And without support of those values, the reader shall not read the fields after this within the OpusSpecificBox.
+ OutputChannelCount:
    The OutputChannelCount field shall be set to the same value as the *Output Channel Count* field in the identification header defined in Ogg Opus [3].
+ PreSkip:
    The PreSkip field indicates the number of the priming samples, that is, the number of samples at 48000 Hz to discard from the decoder output when starting playback. The value of the PreSkip field shall be at least 80 milliseconds' worth of PCM samples even when removing any number of Opus samples which may or may not contain the priming samples. The PreSkip field is not used for discarding the priming samples at the whole playback at all since it is informative only, and that task falls on the Edit List Box.
+ InputSampleRate:
    The InputSampleRate field shall be set to the same value as the *Input Sample Rate* field in the identification header defined in Ogg Opus [3].
+ OutputGain:
    The OutputGain field shall be set to the same value as the *Output Gain* field in the identification header define in Ogg Opus [3]. Note that the value is stored as 8.8 fixed-point.
+ ChannelMappingFamily:
    The ChannelMappingFamily field shall be set to the same value as the *Channel Mapping Family* field in the identification header defined in Ogg Opus [3].
+ StreamCount:

The StreamCount field shall be set to the same value as the *Stream Count* field in the identification header defined in Ogg Opus [3].

+ CoupledCount:
  The CoupledCount field shall be set to the same value as the *Coupled Count* field in the identification header defined in Ogg Opus [3].
+ ChannelMapping:
  The ChannelMapping field shall be set to the same octet string as *Channel Mapping* field in the identification header defined in Ogg Opus [3].

4.3.3 Sample format

An Opus sample is exactly one Opus packet for each of different Opus bitstreams. Due to support more than two channels, an Opus sample can contain frames from multiple Opus bitstreams but all Opus packets shall share with the total of frame sizes in a single Opus sample. The way of how to pack an Opus packet from each of Opus bitstreams into a single Opus sample follows Appendix B. in RFC 6716 [2].
The endianness has nothing to do with any Opus sample since every Opus packet is processed byte-by-byte.
In this specification, 'sample' means 'Opus sample' except for 'padded samples', 'priming samples', 'valid sample' and 'sample-accurate', i.e. 'sample' is 'sample' in the term defined in ISO/IEC 14496-12 [1].

```
+---------------------------------------+-----------------------------------+
| Opus packet 0 (self-delimiting framing) | Opus packet 1 (undelimited framing) |
+---------------------------------------+-----------------------------------+
|<---------------------------- the size of Opus sample ----------------------->|
```

Figure 1 - Example structure of an Opus sample containing two Opus bitstreams

4.3.4 Duration of Opus sample

The duration of Opus sample is given by multiplying the total of frame sizes for a single Opus bitstream expressed in seconds by the value of the timescale field in the Media Header Box.
Let's say an Opus sample consists of two Opus bitstreams, where the frame size of one bitstream is 40 milliseconds and the frame size of another is 60 milliseconds, and the timescale field in the Media Header Box is set to 48000, then the duration of that Opus sample shall be 120 milliseconds since three 40 millisecond frame and two 60 millisecond frames shall be contained because of the maximum duration of Opus packet, 120 milliseconds, and 5760 in the timescale indicated in the Media Header Box.

To indicate the valid samples excluding the padded samples at the end of Opus bitstream, the duration of the last Opus sample of an Opus bitstream is given by multiplying the number of the valid samples by the value produced by dividing the value of the timescale field in the Media Header Box by 48000.

4.3.5 Sub-sample

The structure of the last Opus packet in an Opus sample is different from the others in the same Opus sample, and the others are invalid Opus packets as an Opus sample because of self-delimiting framing. To avoid complexities, sub-sample is not defined for Opus sample in this specification.

4.3.6 Random Access

This subclause describes the nature of the random access of Opus sample.

4.3.6.1 Random Access Point

All Opus samples can be independently decoded i.e. every Opus sample is a sync sample. Therefore, the Sync Sample Box shall not be present as long as there are no samples other than Opus samples in the same track. And the sample_is_non_sync_sample field for Opus samples shall be set to 0.

4.3.6.2 Pre-roll

Opus bitstream requires at least 80 millisecond pre-roll after each random access to get correct output. Pre-roll is indicated by the roll_distance field in AudioRollRecoveryEntry. AudioPreRollEntry shall not be used since every Opus sample is a sync sample in Opus bitstream. Note that roll_distance is expressed in sample units in a term of ISO Base Media File Format, and always takes negative values.

For any track containing Opus bitstreams, at least one Sample Group Description Box and at least one Sample to Group Box within the Sample Table Box shall be present and these have the grouping_type field set to 'roll'. If any Opus sample is contained in a track fragment, the Sample to Group Box with the grouping_type field set to 'roll' shall be present for that track fragment.

For the requirement of AudioRollRecoveryEntry, the compatible_brands field in the File Type Box shall contain at least one brand which requires support for roll groups.

4.4 Trimming of Actual Duration

Due to the priming samples (or the padding at the beginning) derived from the pre-roll for the startup and the padded samples at the end, we need trim from media to get the actual duration. An edit in the Edit List Box can achieve this demand, and the Edit Box and the Edit List Box shall be present.

For sample-accurate trimming, proper timescale should be set to the timescale field in the Movie Header Box and the Media Header Box inside Track Box(es) for Opus bitstream. The timescale field in the Media Header Box is typically set to 48000. It is recommended that the timescale field in the Movie Header Box be set to the same value of the timescale field in the Media Header Box in order to avoid the rounding problem when specifying duration of edit if the timescales in all of the Media Header Boxes are set to the same value.

For example, to indicate the actual duration of an Opus bitstream in a track with the timescale fields of both the Movie Header Box and the Media Header Box set to 48000, we would use the following edit:

```
segment_duration = the number of the valid samples
media_time = the number of the priming samples
media_rate = 1 << 16
```

The Edit List Box is applied to whole movie including all movie fragments. Therefore, it is impossible to tell the actual duration in the case producing movie fragments on the fly such as live-streaming. In such cases, the duration of the last Opus sample may be helpful by setting zero to the segment_duration field since the value 0 represents implicit duration equal to the sum of the duration of all samples.

4.5 Channel Layout (informative)

By the application of alternate_group in the Track Header Box, whole audio channels in all active tracks from non-alternate group and/or different alternate group from each other are composited into the presentation. If an Opus sample consists of multiple Opus bitstreams, it can be splitted into individual Opus bitstreams and reconstructed into new Opus samples as long as every Opus bitstream has the same total duration in each Opus sample. This nature can be utilized to encapsulate a single Opus bitstream in each track without breaking the original channel layout.

As an example, let's say there is a following track:

```
    OutputChannelCount = 6;
    StreamCount        = 4;
    CoupledCount       = 2;
    ChannelMapping     = {0, 4, 1, 2, 3, 5}; // front left, front center, front right, rear left, rear right, LFE
```

Here, to couple front left to front right channels into the first stream, and couple rear left to rear right channels into the second stream, reordering is needed since coupled streams must precede any non-coupled stream. You extract the four Opus bitstreams from this track and you encapsulate two of the four into a track and the others into another track. The former track is as follows.

```
    OutputChannelCount = 6;
    StreamCount        = 2;
    CoupledCount       = 2;
    ChannelMapping     = {0, 255, 1, 2, 3, 255}; // front left, front center, front right, rear left, rear right, LFE
```

And the latter track is as follows.

```
    OutputChannelCount = 6;
    StreamCount        = 2;
    CoupledCount       = 0;
    ChannelMapping     = {255, 0, 255, 255, 255, 1}; // front left, front center, front right, rear left, rear right, LF
```

In addition, the value of the alternate_group field in the both tracks is set to 0. As the result, the player may play as if channels with 255 are not present, and play the presentation constructed from the both tracks in the same channel layout as the one of the original track. Keep in mind that the way of the composition, i.e. the mixing for playback, is not defined here, and maybe different results could occur except for the channel layout of the original, depending on an implementation or the definition of a derived file format.

Note that some derived file formats may specify the restriction to ignore alternate grouping. In the context of such file formats, this application is not available. This unavailability does not mean incompatibilities among file formats unless the restriction to the value of the alternate_group field is specified and brings about any conflict among their definitions.

4.6 Basic Structure (informative)

4.6.1 Initial Movie

This subclause shows a basic structure of the Movie Box as follows:

```
+----+----+----+----+----+----+----+----+------------------------------+
|moov|    |    |    |    |    |    |    | Movie Box                    |
+----+----+----+----+----+----+----+----+------------------------------+
|    |mvhd|    |    |    |    |    |    | Movie Header Box             |
+----+----+----+----+----+----+----+----+------------------------------+
|    |trak|    |    |    |    |    |    | Track Box                    |
+----+----+----+----+----+----+----+----+------------------------------+
|    |    |tkhd|    |    |    |    |    | Track Header Box             |
+----+----+----+----+----+----+----+----+------------------------------+
|    |    |edts|    |    |    |    |    | Edit Box                     |
+----+----+----+----+----+----+----+----+------------------------------+
|    |    |    |elst|    |    |    |    | Edit List Box                |
+----+----+----+----+----+----+----+----+------------------------------+
|    |    |mdia|    |    |    |    |    | Media Box                    |
+----+----+----+----+----+----+----+----+------------------------------+
|    |    |    |mdhd|    |    |    |    | Media Header Box             |
+----+----+----+----+----+----+----+----+------------------------------+
|    |    |    |hdlr|    |    |    |    | Handler Reference Box        |
+----+----+----+----+----+----+----+----+------------------------------+
|    |    |    |minf|    |    |    |    | Media Information Box        |
+----+----+----+----+----+----+----+----+------------------------------+
|    |    |    |    |smhd|    |    |    | Sound Media Header Box       |
+----+----+----+----+----+----+----+----+------------------------------+
|    |    |    |    |dinf|    |    |    | Data Information Box         |
+----+----+----+----+----+----+----+----+------------------------------+
|    |    |    |    |    |dref|    |    | Data Reference Box           |
+----+----+----+----+----+----+----+----+------------------------------+
|    |    |    |    |    |    |url |    | DataEntryUrlBox              |
+----+----+----+----+----+----+ or +----+------------------------------+
|    |    |    |    |    |    |urn |    | DataEntryUrnBox              |
+----+----+----+----+----+----+----+----+------------------------------+
|    |    |    |    |stbl|    |    |    | Sample Table                 |
+----+----+----+----+----+----+----+----+------------------------------+
|    |    |    |    |    |stsd|    |    | Sample Description Box       |
+----+----+----+----+----+----+----+----+------------------------------+
|    |    |    |    |    |    |Opus|    | OpusSampleEntry              |
+----+----+----+----+----+----+----+----+------------------------------+
|    |    |    |    |    |    |    |dOps| Opus Specific Box            |
+----+----+----+----+----+----+----+----+------------------------------+
|    |    |    |    |    |stts|    |    | Decoding Time to Sample Box  |
+----+----+----+----+----+----+----+----+------------------------------+
|    |    |    |    |    |stsc|    |    | Sample To Chunk Box          |
+----+----+----+----+----+----+----+----+------------------------------+
|    |    |    |    |    |stsz|    |    | Sample Size Box              |
+----+----+----+----+----+ or +----+----+------------------------------+
|    |    |    |    |    |stz2|    |    | Compact Sample Size Box      |
+----+----+----+----+----+----+----+----+------------------------------+
|    |    |    |    |    |stco|    |    | Chunk Offset Box             |
+----+----+----+----+----+ or +----+----+-----------------------------+
|    |    |    |    |    |co64|    |    | Chunk Large Offset Box       |
+----+----+----+----+----+----+----+----+-----------------------------+
|    |    |    |    |    |sgpd|    |    | Sample Group Description Box |
+----+----+----+----+----+----+----+----+-----------------------------+
|    |    |    |    |    |sbgp|    |    | Sample to Group Box          |
+----+----+----+----+----+----+----+----+-----------------------------+
|    |mvex|*   |    |    |    |    |    | Movie Extends Box            |
+----+----+----+----+----+----+----+----+-----------------------------+
|    |    |trex|*   |    |    |    |    | Track Extends Box            |
+----+----+----+----+----+----+----+----+-----------------------------+
```

Figure 2 - Basic structure of Movie Box

It is strongly recommended that the order of boxes should follow the above structure.
Boxes marked with an asterisk (*) may be present.
For most boxes listed above, the definition is as is defined in ISO/IEC 14496-12 [1]. The additional boxes and the additional requirements, restrictions and recommendations to the other boxes are described in this specification.

4.6.2 Movie Fragments

This subclause shows a basic structure of the Movie Fragment Box as follows:

```
+----+----+----+----+----+----+----+----+-----------------------------+
|moof|    |    |    |    |    |    |    | Movie Fragment Box           |
+----+----+----+----+----+----+----+----+-----------------------------+
|    |mfhd|    |    |    |    |    |    | Movie Fragment Header Box    |
+----+----+----+----+----+----+----+----+-----------------------------+
|    |traf|    |    |    |    |    |    | Track Fragment Box           |
+----+----+----+----+----+----+----+----+-----------------------------+
|    |    |tfhd|    |    |    |    |    | Track Fragment Header Box    |
+----+----+----+----+----+----+----+----+-----------------------------+
|    |    |trun|    |    |    |    |    | Track Fragment Run Box       |
+----+----+----+----+----+----+----+----+-----------------------------+
|    |    |sgpd|*   |    |    |    |    | Sample Group Description Box |
+----+----+----+----+----+----+----+----+-----------------------------+
|    |    |sbgp|*   |    |    |    |    | Sample to Group Box          |
+----+----+----+----+----+----+----+----+-----------------------------+
```

Figure 3 - Basic structure of Movie Fragment Box

It is strongly recommended that the Movie Fragment Header Box and the Track Fragment Header Box be placed first in their container.
Boxes marked with an asterisk (*) may be present.
For the boxes listed above, the definition is as is defined in ISO/IEC 14496-12 [1].

4.7 Example of Encapsulation (informative)

```
[File]
    size = 17790
    [ftyp: File Type Box]
        position = 0
        size = 24
        major_brand = mp42 : MP4 version 2
        minor_version = 0
        compatible_brands
            brand[0] = mp42 : MP4 version 2
            brand[1] = iso2 : ISO Base Media file format version 2
    [moov: Movie Box]
        position = 24
        size = 757
        [mvhd: Movie Header Box]
            position = 32
            size = 108
            version = 0
            flags = 0x000000
            creation_time = UTC 2014/12/12, 18:41:19
            modification_time = UTC 2014/12/12, 18:41:19
            timescale = 48000
            duration = 33600 (00:00:00.700)
            rate = 1.000000
            volume = 1.000000
            reserved = 0x0000
            reserved = 0x00000000
            reserved = 0x00000000
            transformation matrix
                | a, b, u |   | 1.000000, 0.000000, 0.000000 |
                | c, d, v | = | 0.000000, 1.000000, 0.000000 |
                | x, y, w |   | 0.000000, 0.000000, 1.000000 |
            pre_defined = 0x00000000
            pre_defined = 0x00000000
            pre_defined = 0x00000000
            pre_defined = 0x00000000
            pre_defined = 0x00000000
            pre_defined = 0x00000000
            next_track_ID = 2
        [iods: Object Descriptor Box]
            position = 140
            size = 33
            version = 0
            flags = 0x000000
            [tag = 0x10: MP4_IOD]
                expandableClassSize = 16
                ObjectDescriptorID = 1
                URL_Flag = 0
                includeInlineProfileLevelFlag = 0
                reserved = 0xf
                ODProfileLevelIndication = 0xff
                sceneProfileLevelIndication = 0xff
                audioProfileLevelIndication = 0xfe
                visualProfileLevelIndication = 0xff
                graphicsProfileLevelIndication = 0xff
                [tag = 0x0e: ES_ID_Inc]
                    expandableClassSize = 4
                    Track_ID = 1
        [trak: Track Box]
            position = 173
            size = 608
            [tkhd: Track Header Box]
                position = 181
                size = 92
                version = 0
                flags = 0x000007
                    Track enabled
                    Track in movie
                    Track in preview
                creation_time = UTC 2014/12/12, 18:41:19
                modification_time = UTC 2014/12/12, 18:41:19
                track_ID = 1
                reserved = 0x00000000
                duration = 33600 (00:00:00.700)
                reserved = 0x00000000
                reserved = 0x00000000
                layer = 0
                alternate_group = 0
                volume = 1.000000
                reserved = 0x0000
                transformation matrix
                    | a, b, u |   | 1.000000, 0.000000, 0.000000 |
                    | c, d, v | = | 0.000000, 1.000000, 0.000000 |
                    | x, y, w |   | 0.000000, 0.000000, 1.000000 |
                width = 0.000000
                height = 0.000000
            [edts: Edit Box]
                position = 273
                size = 36
                [elst: Edit List Box]
                    position = 281
                    size = 28
                    version = 0
                    flags = 0x000000
                    entry_count = 1
                    entry[0]
                        segment_duration = 33600
                        media_time = 312
                        media_rate = 1.000000
            [mdia: Media Box]
                position = 309
                size = 472
                [mdhd: Media Header Box]
                    position = 317
                    size = 32
                    version = 0
                    flags = 0x000000
                    creation_time = UTC 2014/12/12, 18:41:19
                    modification_time = UTC 2014/12/12, 18:41:19
                    timescale = 48000
                    duration = 34560 (00:00:00.720)
                    language = und
                    pre_defined = 0x0000
                [hdlr: Handler Reference Box]
                    position = 349
                    size = 51
                    version = 0
                    flags = 0x000000
                    pre_defined = 0x00000000
                    handler_type = soun
                    reserved = 0x00000000
                    reserved = 0x00000000
                    reserved = 0x00000000
                    name = Xiph Audio Handler
                [minf: Media Information Box]
                    position = 400
                    size = 381
                    [smhd: Sound Media Header Box]
                        position = 408
                        size = 16
            version = 0
            flags = 0x000000
            balance = 0.000000
            reserved = 0x0000
        [dinf: Data Information Box]
            position = 424
            size = 36
            [dref: Data Reference Box]
                position = 432
                size = 28
                version = 0
                flags = 0x000000
                entry_count = 1
                [url : Data Entry Url Box]
                    position = 448
                    size = 12
                    version = 0
                    flags = 0x000001
                    location = in the same file
        [stbl: Sample Table Box]
            position = 460
            size = 321
            [stsd: Sample Description Box]
                position = 468
                size = 79
                version = 0
                flags = 0x000000
                entry_count = 1
                [Opus: Audio Description]
                    position = 484
                    size = 63
                    reserved = 0x000000000000
                    data_reference_index = 1
                    reserved = 0x0000
                    reserved = 0x0000
                    reserved = 0x00000000
                    channelcount = 6
                    samplesize = 16
                    pre_defined = 0
                    reserved = 0
                    samplerate = 48000.000000
                    [dOps: Opus Specific Box]
                        position = 520
                        size = 27
                        Version = 0
                        OutputChannelCount = 6
                        PreSkip = 312
                        InputSampleRate = 48000
                        OutputGain = 0
                        ChannelMappingFamily = 1
                        StreamCount = 4
                        CoupledCount = 2
                        ChannelMapping
                            0 -> 0: front left
                            1 -> 4: fron center
                            2 -> 1: front right
                            3 -> 2: side left
                            4 -> 3: side right
                            5 -> 5: rear center
            [stts: Decoding Time to Sample Box]
                position = 547
                size = 24
                version = 0
                flags = 0x000000
                entry_count = 1
                entry[0]
                    sample_count = 18
                    sample_delta = 1920
            [stsc: Sample To Chunk Box]
                position = 571
                size = 40
                version = 0
                flags = 0x000000
                entry_count = 2
                entry[0]
                    first_chunk = 1
                    samples_per_chunk = 13
                    sample_description_index = 1
                entry[1]
                    first_chunk = 2
                    samples_per_chunk = 5
                    sample_description_index = 1
            [stsz: Sample Size Box]
                position = 611
                size = 92
                version = 0
                flags = 0x000000
                sample_size = 0 (variable)
```

```
                                        sample_count = 18
                                        entry_size[0] = 977
                                        entry_size[1] = 938
                                        entry_size[2] = 939
                                        entry_size[3] = 938
                                        entry_size[4] = 934
                                        entry_size[5] = 945
                                        entry_size[6] = 948
                                        entry_size[7] = 956
                                        entry_size[8] = 955
                                        entry_size[9] = 930
                                        entry_size[10] = 933
                                        entry_size[11] = 934
                                        entry_size[12] = 972
                                        entry_size[13] = 977
                                        entry_size[14] = 958
                                        entry_size[15] = 949
                                        entry_size[16] = 962
                                        entry_size[17] = 848
                                    [stco: Chunk Offset Box]
                                        position = 703
                                        size = 24
                                        version = 0
                                        flags = 0x000000
                                        entry_count = 2
                                        chunk_offset[0] = 797
                                        chunk_offset[1] = 13096
                                    [sgpd: Sample Group Description Box]
                                        position = 727
                                        size = 26
                                        version = 1
                                        flags = 0x000000
                                        grouping_type = roll
                                        default_length = 2 (constant)
                                        entry_count = 1
                                        roll_distance[0] = -2
                                    [sbgp: Sample to Group Box]
                                        position = 753
                                        size = 28
                                        version = 0
                                        flags = 0x000000
                                        grouping_type = roll
                                        entry_count = 1
                                        entry[0]
                                            sample_count = 18
                                            group_description_index = 1
                [free: Free Space Box]
                    position = 781
                    size = 8
                [mdat: Media Data Box]
                    position = 789
                    size = 17001
```

5 Authors' Address

Yusuke Nakamura
Email: muken.the.vfrmaniac |at| gmail.com