```
$Id: id3v2.4.0-frames.txt,v 1.1 2003/07/27 18:28:34 id3 Exp $

Informal standard                                          M. Nilsson
Document: id3v2.4.0-frames.txt                       1st November 2000


                  ID3 tag version 2.4.0 - Native Frames

Status of this document

   This document is an informal standard and replaces the ID3v2.3.0
   standard [ID3v2]. A formal standard will use another revision number
   even if the content is identical to document. The contents in this
   document may change for clarifications but never for added or altered
   functionallity.

   Distribution of this document is unlimited.


Abstract

   This document describes the frames natively supported by ID3v2.4.0,
   which is a revised version of the ID3v2 informal standard [ID3v2.3.0]
   version 2.3.0. The ID3v2 offers a flexible way of storing audio meta
   information within audio file itself. The information may be
   technical information, such as equalisation curves, as well as title,
   performer, copyright etc.

   ID3v2.4.0 is meant to be as close as possible to ID3v2.3.0 in order
   to allow for implementations to be revised as easily as possible.


1.   Table of contents

   2.   Conventions in this document
   3.   Default flags
   4.   Declared ID3v2 frames
     4.1.   Unique file identifier
     4.2.   Text information frames
       4.2.1.   Identification frames
       4.2.2.   Involved persons frames
       4.2.3.   Derived and subjective properties frames
       4.2.4.   Rights and license frames
       4.2.5.   Other text frames
       4.2.6.   User defined text information frame
     4.3.   URL link frames
       4.3.1.   URL link frames - details
       4.3.2.   User defined URL link frame
     4.4.   Music CD Identifier
```

2.   Conventions in this document

   Text within "" is a text string exactly as it appears in a tag.
   Numbers preceded with $ are hexadecimal and numbers preceded with %
   are binary. $xx is used to indicate a byte with unknown content. %x
   is used to indicate a bit with unknown content. The most significant
   bit (MSB) of a byte is called 'bit 7' and the least significant bit
   (LSB) is called 'bit 0'.

   A tag is the whole tag described the ID3v2 main structure document
   [ID3v2-strct]. A frame is a block of information in the tag. The tag
   consists of a header, frames and optional padding. A field is a piece
   of information; one value, a string etc. A numeric string is a string
   that consists of the characters "0123456789" only.

   The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
   "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
   document are to be interpreted as described in RFC 2119 [KEYWORDS].

3.   Default flags

   The default settings for the frames described in this document can be

divided into the following classes. The flags may be set differently
if found more suitable by the software.

1. Discarded if tag is altered, discarded if file is altered.

   None.

2. Discarded if tag is altered, preserved if file is altered.

   None.

3. Preserved if tag is altered, discarded if file is altered.

   ASPI, AENC, ETCO, EQU2, MLLT, POSS, SEEK, SYLT, SYTC, RVA2, TENC,
   TLEN

4. Preserved if tag is altered, preserved if file is altered.

   The rest of the frames.

4.   Declared ID3v2 frames

  The following frames are declared in this draft.

  4.19 AENC Audio encryption
  4.14 APIC Attached picture
  4.30 ASPI Audio seek point index

  4.10 COMM Comments
  4.24 COMR Commercial frame

  4.25 ENCR Encryption method registration
  4.12 EQU2 Equalisation (2)
  4.5   ETCO Event timing codes

  4.15 GEOB General encapsulated object
  4.26 GRID Group identification registration

  4.20 LINK Linked information

  4.4   MCDI Music CD identifier
  4.6   MLLT MPEG location lookup table

  4.23 OWNE Ownership frame

  4.27 PRIV Private frame
  4.16 PCNT Play counter
  4.17 POPM Popularimeter
  4.21 POSS Position synchronisation frame

  4.18 RBUF Recommended buffer size
  4.11 RVA2 Relative volume adjustment (2)
  4.13 RVRB Reverb

  4.29 SEEK Seek frame

```
4.28  SIGN Signature frame
4.9   SYLT Synchronised lyric/text
4.7   SYTC Synchronised tempo codes

4.2.1 TALB Album/Movie/Show title
4.2.3 TBPM BPM (beats per minute)
4.2.2 TCOM Composer
4.2.3 TCON Content type
4.2.4 TCOP Copyright message
4.2.5 TDEN Encoding time
4.2.5 TDLY Playlist delay
4.2.5 TDOR Original release time
4.2.5 TDRC Recording time
4.2.5 TDRL Release time
4.2.5 TDTG Tagging time
4.2.2 TENC Encoded by
4.2.2 TEXT Lyricist/Text writer
4.2.3 TFLT File type
4.2.2 TIPL Involved people list
4.2.1 TIT1 Content group description
4.2.1 TIT2 Title/songname/content description
4.2.1 TIT3 Subtitle/Description refinement
4.2.3 TKEY Initial key
4.2.3 TLAN Language(s)
4.2.3 TLEN Length
4.2.2 TMCL Musician credits list
4.2.3 TMED Media type
4.2.3 TMOO Mood
4.2.1 TOAL Original album/movie/show title
4.2.5 TOFN Original filename
4.2.2 TOLY Original lyricist(s)/text writer(s)
4.2.2 TOPE Original artist(s)/performer(s)
4.2.4 TOWN File owner/licensee
4.2.2 TPE1 Lead performer(s)/Soloist(s)
4.2.2 TPE2 Band/orchestra/accompaniment
4.2.2 TPE3 Conductor/performer refinement
4.2.2 TPE4 Interpreted, remixed, or otherwise modified by
4.2.1 TPOS Part of a set
4.2.4 TPRO Produced notice
4.2.4 TPUB Publisher
4.2.1 TRCK Track number/Position in set
4.2.4 TRSN Internet radio station name
4.2.4 TRSO Internet radio station owner
4.2.5 TSOA Album sort order
4.2.5 TSOP Performer sort order
4.2.5 TSOT Title sort order
4.2.1 TSRC ISRC (international standard recording code)
4.2.5 TSSE Software/Hardware and settings used for encoding
4.2.1 TSST Set subtitle
4.2.2 TXXX User defined text information frame

4.1   UFID Unique file identifier
4.22  USER Terms of use
4.8   USLT Unsynchronised lyric/text transcription

4.3.1 WCOM Commercial information
```

4.1.   Unique file identifier

This frame's purpose is to be able to identify the audio file in a
database, that may provide more information relevant to the content.
Since standardisation of such a database is beyond this document, all
UFID frames begin with an 'owner identifier' field. It is a null-
terminated string with a URL [URL] containing an email address, or a
link to a location where an email address can be found, that belongs
to the organisation responsible for this specific database
implementation. Questions regarding the database should be sent to
the indicated email address. The URL should not be used for the
actual database queries. The string
"http://www.id3.org/dummy/ufid.html" should be used for tests. The
'Owner identifier' must be non-empty (more than just a termination).
The 'Owner identifier' is then followed by the actual identifier,
which may be up to 64 bytes. There may be more than one "UFID" frame
in a tag, but only one with the same 'Owner identifier'.

       <Header for 'Unique file identifier', ID: "UFID">
       Owner identifier   <text string> $00
       Identifier   <up to 64 bytes binary data>

4.2.   Text information frames

The text information frames are often the most important frames,
containing information like artist, album and more. There may only be
one text information frame of its kind in an tag. All text
information frames supports multiple strings, stored as a null
separated list, where null is reperesented by the termination code
for the charater encoding. All text frame identifiers begin with "T".
Only text frame identifiers begin with "T", with the exception of the
"TXXX" frame. All the text information frames have the following
format:

       <Header for 'Text information frame', ID: "T000" - "TZZZ",
       excluding "TXXX" described in 4.2.6.>
       Text encoding $xx
       Information   <text string(s) according to encoding>

4.2.1.   Identification frames

  TIT1
  The 'Content group description' frame is used if the sound belongs to
  a larger category of sounds/music. For example, classical music is

often sorted in different musical sections (e.g. "Piano Concerto",
"Weather - Hurricane").

TIT2
The 'Title/Songname/Content description' frame is the actual name of
the piece (e.g. "Adagio", "Hurricane Donna").

TIT3
The 'Subtitle/Description refinement' frame is used for information
directly related to the contents title (e.g. "Op. 16" or "Performed
live at Wembley").

TALB
The 'Album/Movie/Show title' frame is intended for the title of the
recording (or source of sound) from which the audio in the file is
taken.

TOAL
The 'Original album/movie/show title' frame is intended for the title
of the original recording (or source of sound), if for example the
music in the file should be a cover of a previously released song.

TRCK
The 'Track number/Position in set' frame is a numeric string
containing the order number of the audio-file on its original
recording. This MAY be extended with a "/" character and a numeric
string containing the total number of tracks/elements on the original
recording. E.g. "4/9".

TPOS
The 'Part of a set' frame is a numeric string that describes which
part of a set the audio came from. This frame is used if the source
described in the "TALB" frame is divided into several mediums, e.g. a
double CD. The value MAY be extended with a "/" character and a
numeric string containing the total number of parts in the set. E.g.
"1/2".

TSST
The 'Set subtitle' frame is intended for the subtitle of the part of
a set this track belongs to.

TSRC
The 'ISRC' frame should contain the International Standard Recording
Code [ISRC] (12 characters).

4.2.2.   Involved persons frames

TPE1
The 'Lead artist/Lead performer/Soloist/Performing group' is
used for the main artist.

TPE2
The 'Band/Orchestra/Accompaniment' frame is used for additional
information about the performers in the recording.

TPE3
 The 'Conductor' frame is used for the name of the conductor.

TPE4
 The 'Interpreted, remixed, or otherwise modified by' frame contains
 more information about the people behind a remix and similar
 interpretations of another existing piece.

TOPE
 The 'Original artist/performer' frame is intended for the performer
 of the original recording, if for example the music in the file
 should be a cover of a previously released song.

TEXT
 The 'Lyricist/Text writer' frame is intended for the writer of the
 text or lyrics in the recording.

TOLY
 The 'Original lyricist/text writer' frame is intended for the
 text writer of the original recording, if for example the music in
 the file should be a cover of a previously released song.

TCOM
 The 'Composer' frame is intended for the name of the composer.

TMCL
 The 'Musician credits list' is intended as a mapping between
 instruments and the musician that played it. Every odd field is an
 instrument and every even is an artist or a comma delimited list of
 artists.

TIPL
 The 'Involved people list' is very similar to the musician credits
 list, but maps between functions, like producer, and names.

TENC
 The 'Encoded by' frame contains the name of the person or
 organisation that encoded the audio file. This field may contain a
 copyright message, if the audio file also is copyrighted by the
 encoder.

4.2.3.   Derived and subjective properties frames

TBPM
 The 'BPM' frame contains the number of beats per minute in the
 main part of the audio. The BPM is an integer and represented as a
 numerical string.

TLEN
 The 'Length' frame contains the length of the audio file in
 milliseconds, represented as a numeric string.

TKEY
 The 'Initial key' frame contains the musical key in which the sound
 starts. It is represented as a string with a maximum length of three

characters. The ground keys are represented with "A","B","C","D","E",
"F" and "G" and halfkeys represented with "b" and "#". Minor is
represented as "m", e.g. "Dbm" $00. Off key is represented with an
"o" only.

TLAN
 The 'Language' frame should contain the languages of the text or
 lyrics spoken or sung in the audio. The language is represented with
 three characters according to ISO-639-2 [ISO-639-2]. If more than one
 language is used in the text their language codes should follow
 according to the amount of their usage, e.g. "eng" $00 "sve" $00.

TCON
 The 'Content type', which ID3v1 was stored as a one byte numeric
 value only, is now a string. You may use one or several of the ID3v1
 types as numerical strings, or, since the category list would be
 impossible to maintain with accurate and up to date categories,
 define your own. Example: "21" $00 "Eurodisco" $00

 You may also use any of the following keywords:

    RX   Remix
    CR   Cover

TFLT
 The 'File type' frame indicates which type of audio this tag defines.
 The following types and refinements are defined:

    MIME   MIME type follows
    MPG   MPEG Audio
      /1   MPEG 1/2 layer I
      /2   MPEG 1/2 layer II
      /3   MPEG 1/2 layer III
      /2.5   MPEG 2.5
      /AAC   Advanced audio compression
    VQF   Transform-domain Weighted Interleave Vector Quantisation
    PCM   Pulse Code Modulated audio

 but other types may be used, but not for these types though. This is
 used in a similar way to the predefined types in the "TMED" frame,
 but without parentheses. If this frame is not present audio type is
 assumed to be "MPG".

TMED
 The 'Media type' frame describes from which media the sound
 originated. This may be a text string or a reference to the
 predefined media types found in the list below. Example:
 "VID/PAL/VHS" $00.

  DIG   Other digital media
    /A   Analogue transfer from media

  ANA   Other analogue media
    /WAC Wax cylinder
    /8CA 8-track tape cassette

```
       CD     CD
         /A    Analogue transfer from media
         /DD   DDD
         /AD   ADD
         /AA   AAD

       LD     Laserdisc

       TT     Turntable records
         /33    33.33 rpm
         /45    45 rpm
         /71    71.29 rpm
         /76    76.59 rpm
         /78    78.26 rpm
         /80    80 rpm

       MD     MiniDisc
         /A    Analogue transfer from media

       DAT    DAT
         /A    Analogue transfer from media
         /1    standard, 48 kHz/16 bits, linear
         /2    mode 2, 32 kHz/16 bits, linear
         /3    mode 3, 32 kHz/12 bits, non-linear, low speed
         /4    mode 4, 32 kHz/12 bits, 4 channels
         /5    mode 5, 44.1 kHz/16 bits, linear
         /6    mode 6, 44.1 kHz/16 bits, 'wide track' play

       DCC    DCC
         /A    Analogue transfer from media

       DVD    DVD
         /A    Analogue transfer from media

       TV     Television
         /PAL     PAL
         /NTSC    NTSC
         /SECAM   SECAM

       VID    Video
         /PAL     PAL
         /NTSC    NTSC
         /SECAM   SECAM
         /VHS     VHS
         /SVHS    S-VHS
         /BETA    BETAMAX

       RAD    Radio
         /FM    FM
         /AM    AM
         /LW    LW
         /MW    MW

       TEL    Telephone
         /I     ISDN
```

```
   MC    MC (normal cassette)
     /4   4.75 cm/s (normal speed for a two sided cassette)
     /9   9.5 cm/s
     /I   Type I cassette (ferric/normal)
     /II  Type II cassette (chrome)
     /III Type III cassette (ferric chrome)
     /IV  Type IV cassette (metal)

   REE   Reel
     /9   9.5 cm/s
     /19  19 cm/s
     /38  38 cm/s
     /76  76 cm/s
     /I   Type I cassette (ferric/normal)
     /II  Type II cassette (chrome)
     /III Type III cassette (ferric chrome)
     /IV  Type IV cassette (metal)
```

TMOO
  The 'Mood' frame is intended to reflect the mood of the audio with a
  few keywords, e.g. "Romantic" or "Sad".

4.2.4.   Rights and license frames

TCOP
  The 'Copyright message' frame, in which the string must begin with a
  year and a space character (making five characters), is intended for
  the copyright holder of the original sound, not the audio file
  itself. The absence of this frame means only that the copyright
  information is unavailable or has been removed, and must not be
  interpreted to mean that the audio is public domain. Every time this
  field is displayed the field must be preceded with "Copyright " (C) "
  ", where (C) is one character showing a C in a circle.

TPRO
  The 'Produced notice' frame, in which the string must begin with a
  year and a space character (making five characters), is intended for
  the production copyright holder of the original sound, not the audio
  file itself. The absence of this frame means only that the production
  copyright information is unavailable or has been removed, and must
  not be interpreted to mean that the audio is public domain. Every
  time this field is displayed the field must be preceded with
  "Produced " (P) " ", where (P) is one character showing a P in a
  circle.

TPUB
  The 'Publisher' frame simply contains the name of the label or
  publisher.

TOWN
  The 'File owner/licensee' frame contains the name of the owner or
  licensee of the file and it's contents.

TRSN
  The 'Internet radio station name' frame contains the name of the

internet radio station from which the audio is streamed.

TRSO
The 'Internet radio station owner' frame contains the name of the owner of the internet radio station from which the audio is streamed.

4.2.5.   Other text frames

TOFN
The 'Original filename' frame contains the preferred filename for the file, since some media doesn't allow the desired length of the filename. The filename is case sensitive and includes its suffix.

TDLY
The 'Playlist delay' defines the numbers of milliseconds of silence that should be inserted before this audio. The value zero indicates that this is a part of a multifile audio track that should be played continuously.

TDEN
The 'Encoding time' frame contains a timestamp describing when the audio was encoded. Timestamp format is described in the ID3v2 structure document [ID3v2-strct].

TDOR
The 'Original release time' frame contains a timestamp describing when the original recording of the audio was released. Timestamp format is described in the ID3v2 structure document [ID3v2-strct].

TDRC
The 'Recording time' frame contains a timestamp describing when the audio was recorded. Timestamp format is described in the ID3v2 structure document [ID3v2-strct].

TDRL
The 'Release time' frame contains a timestamp describing when the audio was first released. Timestamp format is described in the ID3v2 structure document [ID3v2-strct].

TDTG
The 'Tagging time' frame contains a timestamp describing then the audio was tagged. Timestamp format is described in the ID3v2 structure document [ID3v2-strct].

TSSE
The 'Software/Hardware and settings used for encoding' frame includes the used audio encoder and its settings when the file was encoded. Hardware refers to hardware encoders, not the computer on which a program was run.

TSOA
The 'Album sort order' frame defines a string which should be used instead of the album name (TALB) for sorting purposes. E.g. an album named "A Soundtrack" might preferably be sorted as "Soundtrack".

```
   TSOP
    The 'Performer sort order' frame defines a string which should be
    used instead of the performer (TPE2) for sorting purposes.

   TSOT
    The 'Title sort order' frame defines a string which should be used
    instead of the title (TIT2) for sorting purposes.
```

4.2.6.   User defined text information frame

```
    This frame is intended for one-string text information concerning the
    audio file in a similar way to the other "T"-frames. The frame body
    consists of a description of the string, represented as a terminated
    string, followed by the actual string. There may be more than one
    "TXXX" frame in each tag, but only one with the same description.

       <Header for 'User defined text information frame', ID: "TXXX">
       Text encoding      $xx
       Description        <text string according to encoding> $00 (00)
       Value              <text string according to encoding>
```

4.3.   URL link frames

```
    With these frames dynamic data such as webpages with touring
    information, price information or plain ordinary news can be added to
    the tag. There may only be one URL [URL] link frame of its kind in an
    tag, except when stated otherwise in the frame description. If the
    text string is followed by a string termination, all the following
    information should be ignored and not be displayed. All URL link
    frame identifiers begins with "W". Only URL link frame identifiers
    begins with "W", except for "WXXX". All URL link frames have the
    following format:

       <Header for 'URL link frame', ID: "W000" - "WZZZ", excluding "WXXX"
       described in 4.3.2.>
       URL               <text string>
```

4.3.1.   URL link frames - details

```
  WCOM
   The 'Commercial information' frame is a URL pointing at a webpage
   with information such as where the album can be bought. There may be
   more than one "WCOM" frame in a tag, but not with the same content.

  WCOP
   The 'Copyright/Legal information' frame is a URL pointing at a
   webpage where the terms of use and ownership of the file is
   described.

  WOAF
   The 'Official audio file webpage' frame is a URL pointing at a file
   specific webpage.
```

WOAR
 The 'Official artist/performer webpage' frame is a URL pointing at
 the artists official webpage. There may be more than one "WOAR" frame
 in a tag if the audio contains more than one performer, but not with
 the same content.

WOAS
 The 'Official audio source webpage' frame is a URL pointing at the
 official webpage for the source of the audio file, e.g. a movie.

WORS
 The 'Official Internet radio station homepage' contains a URL
 pointing at the homepage of the internet radio station.

WPAY
 The 'Payment' frame is a URL pointing at a webpage that will handle
 the process of paying for this file.

WPUB
 The 'Publishers official webpage' frame is a URL pointing at the
 official webpage for the publisher.

4.3.2.   User defined URL link frame

 This frame is intended for URL [URL] links concerning the audio file
 in a similar way to the other "W"-frames. The frame body consists
 of a description of the string, represented as a terminated string,
 followed by the actual URL. The URL is always encoded with ISO-8859-1
 [ISO-8859-1]. There may be more than one "WXXX" frame in each tag,
 but only one with the same description.

    <Header for 'User defined URL link frame', ID: "WXXX">
    Text encoding $xx
    Description   <text string according to encoding> $00 (00)
    URL   <text string>

4.4.   Music CD identifier

 This frame is intended for music that comes from a CD, so that the CD
 can be identified in databases such as the CDDB [CDDB]. The frame
 consists of a binary dump of the Table Of Contents, TOC, from the CD,
 which is a header of 4 bytes and then 8 bytes/track on the CD plus 8
 bytes for the 'lead out', making a maximum of 804 bytes. The offset
 to the beginning of every track on the CD should be described with a
 four bytes absolute CD-frame address per track, and not with absolute
 time. When this frame is used the presence of a valid "TRCK" frame is
 REQUIRED, even if the CD's only got one track. It is recommended that
 this frame is always added to tags originating from CDs. There may
 only be one "MCDI" frame in each tag.

    <Header for 'Music CD identifier', ID: "MCDI">
    CD TOC   <binary data>

```
4.5.   Event timing codes

This frame allows synchronisation with key events in the audio. The
header is:

   <Header for 'Event timing codes', ID: "ETCO">
   Time stamp format     $xx

Where time stamp format is:

   $01  Absolute time, 32 bit sized, using MPEG [MPEG] frames as unit
   $02  Absolute time, 32 bit sized, using milliseconds as unit

Absolute time means that every stamp contains the time from the
beginning of the file.

Followed by a list of key events in the following format:

   Type of event    $xx
   Time stamp       $xx (xx ...)

The 'Time stamp' is set to zero if directly at the beginning of the
sound or after the previous event. All events MUST be sorted in
chronological order. The type of event is as follows:

   $00  padding (has no meaning)
   $01  end of initial silence
   $02  intro start
   $03  main part start
   $04  outro start
   $05  outro end
   $06  verse start
   $07  refrain start
   $08  interlude start
   $09  theme start
   $0A  variation start
   $0B  key change
   $0C  time change
   $0D  momentary unwanted noise (Snap, Crackle & Pop)
   $0E  sustained noise
   $0F  sustained noise end
   $10  intro end
   $11  main part end
   $12  verse end
   $13  refrain end
   $14  theme end
   $15  profanity
   $16  profanity end

   $17-$DF  reserved for future use

   $E0-$EF  not predefined synch 0-F

   $F0-$FC  reserved for future use

   $FD  audio end (start of silence)
```

```
        $FE   audio file ends
        $FF   one more byte of events follows (all the following bytes with
              the value $FF have the same function)
```

   Terminating the start events such as "intro start" is OPTIONAL. The
   'Not predefined synch's ($E0-EF) are for user events. You might want
   to synchronise your music to something, like setting off an explosion
   on-stage, activating a screensaver etc.

   There may only be one "ETCO" frame in each tag.

4.6.   MPEG location lookup table

   To increase performance and accuracy of jumps within a MPEG [MPEG]
   audio file, frames with time codes in different locations in the file
   might be useful. This ID3v2 frame includes references that the
   software can use to calculate positions in the file. After the frame
   header follows a descriptor of how much the 'frame counter' should be
   increased for every reference. If this value is two then the first
   reference points out the second frame, the 2nd reference the 4th
   frame, the 3rd reference the 6th frame etc. In a similar way the
   'bytes between reference' and 'milliseconds between reference' points
   out bytes and milliseconds respectively.

   Each reference consists of two parts; a certain number of bits, as
   defined in 'bits for bytes deviation', that describes the difference
   between what is said in 'bytes between reference' and the reality and
   a certain number of bits, as defined in 'bits for milliseconds
   deviation', that describes the difference between what is said in
   'milliseconds between reference' and the reality. The number of bits
   in every reference, i.e. 'bits for bytes deviation'+'bits for
   milliseconds deviation', must be a multiple of four. There may only
   be one "MLLT" frame in each tag.

```
     <Header for 'Location lookup table', ID: "MLLT">
     MPEG frames between reference  $xx xx
     Bytes between reference        $xx xx xx
     Milliseconds between reference $xx xx xx
     Bits for bytes deviation       $xx
     Bits for milliseconds dev.     $xx
```

   Then for every reference the following data is included;

```
     Deviation in bytes         %xxx....
     Deviation in milliseconds  %xxx....
```


4.7.   Synchronised tempo codes

   For a more accurate description of the tempo of a musical piece, this
   frame might be used. After the header follows one byte describing
   which time stamp format should be used. Then follows one or more
   tempo codes. Each tempo code consists of one tempo part and one time
   part. The tempo is in BPM described with one or two bytes. If the
   first byte has the value $FF, one more byte follows, which is added

```
        to the first giving a range from 2 - 510 BPM, since $00 and $01 is
        reserved. $00 is used to describe a beat-free time period, which is
        not the same as a music-free time period. $01 is used to indicate one
        single beat-stroke followed by a beat-free period.

        The tempo descriptor is followed by a time stamp. Every time the
        tempo in the music changes, a tempo descriptor may indicate this for
        the player. All tempo descriptors MUST be sorted in chronological
        order. The first beat-stroke in a time-period is at the same time as
        the beat description occurs. There may only be one "SYTC" frame in
        each tag.

          <Header for 'Synchronised tempo codes', ID: "SYTC">
          Time stamp format    $xx
          Tempo data           <binary data>

        Where time stamp format is:

          $01  Absolute time, 32 bit sized, using MPEG [MPEG] frames as unit
          $02  Absolute time, 32 bit sized, using milliseconds as unit

        Absolute time means that every stamp contains the time from the
        beginning of the file.
```

4.8.   Unsynchronised lyrics/text transcription

```
    This frame contains the lyrics of the song or a text transcription of
    other vocal activities. The head includes an encoding descriptor and
    a content descriptor. The body consists of the actual text. The
    'Content descriptor' is a terminated string. If no descriptor is
    entered, 'Content descriptor' is $00 (00) only. Newline characters
    are allowed in the text. There may be more than one 'Unsynchronised
    lyrics/text transcription' frame in each tag, but only one with the
    same language and content descriptor.

      <Header for 'Unsynchronised lyrics/text transcription', ID: "USLT">
      Text encoding        $xx
      Language             $xx xx xx
      Content descriptor   <text string according to encoding> $00 (00)
      Lyrics/text          <full text string according to encoding>
```

4.9.   Synchronised lyrics/text

```
    This is another way of incorporating the words, said or sung lyrics,
    in the audio file as text, this time, however, in sync with the
    audio. It might also be used to describing events e.g. occurring on a
    stage or on the screen in sync with the audio. The header includes a
    content descriptor, represented with as terminated text string. If no
    descriptor is entered, 'Content descriptor' is $00 (00) only.

      <Header for 'Synchronised lyrics/text', ID: "SYLT">
      Text encoding        $xx
      Language             $xx xx xx
      Time stamp format    $xx
```

```
     Content type       $xx
     Content descriptor   <text string according to encoding> $00 (00)


   Content type:   $00 is other
                   $01 is lyrics
                   $02 is text transcription
                   $03 is movement/part name (e.g. "Adagio")
                   $04 is events (e.g. "Don Quijote enters the stage")
                   $05 is chord (e.g. "Bb F Fsus")
                   $06 is trivia/'pop up' information
                   $07 is URLs to webpages
                   $08 is URLs to images


   Time stamp format:

     $01  Absolute time, 32 bit sized, using MPEG [MPEG] frames as unit
     $02  Absolute time, 32 bit sized, using milliseconds as unit

   Absolute time means that every stamp contains the time from the
   beginning of the file.

   The text that follows the frame header differs from that of the
   unsynchronised lyrics/text transcription in one major way. Each
   syllable (or whatever size of text is considered to be convenient by
   the encoder) is a null terminated string followed by a time stamp
   denoting where in the sound file it belongs. Each sync thus has the
   following structure:

     Terminated text to be synced (typically a syllable)
     Sync identifier (terminator to above string)    $00 (00)
     Time stamp                                       $xx (xx ...)

   The 'time stamp' is set to zero or the whole sync is omitted if
   located directly at the beginning of the sound. All time stamps
   should be sorted in chronological order. The sync can be considered
   as a validator of the subsequent string.

   Newline characters are allowed in all "SYLT" frames and MUST be used
   after every entry (name, event etc.) in a frame with the content type
   $03 - $04.

   A few considerations regarding whitespace characters: Whitespace
   separating words should mark the beginning of a new word, thus
   occurring in front of the first syllable of a new word. This is also
   valid for new line characters. A syllable followed by a comma should
   not be broken apart with a sync (both the syllable and the comma
   should be before the sync).

   An example: The "USLT" passage

     "Strangers in the night" $0A "Exchanging glances"

   would be "SYLT" encoded as:

     "Strang" $00 xx xx "ers" $00 xx xx " in" $00 xx xx " the" $00 xx xx
     " night" $00 xx xx 0A "Ex" $00 xx xx "chang" $00 xx xx "ing" $00 xx
```

```
      xx "glan" $00 xx xx "ces" $00 xx xx
```

   There may be more than one "SYLT" frame in each tag, but only one
   with the same language and content descriptor.

4.10.   Comments

   This frame is intended for any kind of full text information that
   does not fit in any other frame. It consists of a frame header
   followed by encoding, language and content descriptors and is ended
   with the actual comment as a text string. Newline characters are
   allowed in the comment text string. There may be more than one
   comment frame in each tag, but only one with the same language and
   content descriptor.

```
      <Header for 'Comment', ID: "COMM">
      Text encoding          $xx
      Language               $xx xx xx
      Short content descrip. <text string according to encoding> $00 (00)
      The actual text        <full text string according to encoding>
```


4.11.   Relative volume adjustment (2)

   This is a more subjective frame than the previous ones. It allows the
   user to say how much he wants to increase/decrease the volume on each
   channel when the file is played. The purpose is to be able to align
   all files to a reference volume, so that you don't have to change the
   volume constantly. This frame may also be used to balance adjust the
   audio. The volume adjustment is encoded as a fixed point decibel
   value, 16 bit signed integer representing (adjustment*512), giving
   +/- 64 dB with a precision of 0.001953125 dB. E.g. +2 dB is stored as
   $04 00 and -2 dB is $FC 00. There may be more than one "RVA2" frame
   in each tag, but only one with the same identification string.

```
      <Header for 'Relative volume adjustment (2)', ID: "RVA2">
      Identification         <text string> $00
```

   The 'identification' string is used to identify the situation and/or
   device where this adjustment should apply. The following is then
   repeated for every channel

```
      Type of channel        $xx
      Volume adjustment      $xx xx
      Bits representing peak  $xx
      Peak volume            $xx (xx ...)


      Type of channel:  $00  Other
                        $01  Master volume
                        $02  Front right
                        $03  Front left
                        $04  Back right
                        $05  Back left
                        $06  Front centre
```

```
                        $07  Back centre
                        $08  Subwoofer
```

Bits representing peak can be any number between 0 and 255. 0 means
that there is no peak volume field. The peak volume field is always
padded to whole bytes, setting the most significant bits to zero.

4.12.   Equalisation (2)

This is another subjective, alignment frame. It allows the user to
predefine an equalisation curve within the audio file. There may be
more than one "EQU2" frame in each tag, but only one with the same
identification string.

```
   <Header of 'Equalisation (2)', ID: "EQU2">
   Interpolation method  $xx
   Identification        <text string> $00
```

The 'interpolation method' describes which method is preferred when
an interpolation between the adjustment point that follows. The
following methods are currently defined:

```
   $00  Band
        No interpolation is made. A jump from one adjustment level to
        another occurs in the middle between two adjustment points.
   $01  Linear
        Interpolation between adjustment points is linear.
```

The 'identification' string is used to identify the situation and/or
device where this adjustment should apply. The following is then
repeated for every adjustment point

```
   Frequency           $xx xx
   Volume adjustment   $xx xx
```

The frequency is stored in units of 1/2 Hz, giving it a range from 0
to 32767 Hz.

The volume adjustment is encoded as a fixed point decibel value, 16
bit signed integer representing (adjustment*512), giving +/- 64 dB
with a precision of 0.001953125 dB. E.g. +2 dB is stored as $04 00
and -2 dB is $FC 00.

Adjustment points should be ordered by frequency and one frequency
should only be described once in the frame.

4.13.   Reverb

Yet another subjective frame, with which you can adjust echoes of
different kinds. Reverb left/right is the delay between every bounce
in ms. Reverb bounces left/right is the number of bounces that should
be made. $FF equals an infinite number of bounces. Feedback is the
amount of volume that should be returned to the next echo bounce. $00
is 0%, $FF is 100%. If this value were $7F, there would be 50% volume

reduction on the first bounce, 50% of that on the second and so on.
Left to left means the sound from the left bounce to be played in the
left speaker, while left to right means sound from the left bounce to
be played in the right speaker.

'Premix left to right' is the amount of left sound to be mixed in the
right before any reverb is applied, where $00 id 0% and $FF is 100%.
'Premix right to left' does the same thing, but right to left.
Setting both premix to $FF would result in a mono output (if the
reverb is applied symmetric). There may only be one "RVRB" frame in
each tag.

```
<Header for 'Reverb', ID: "RVRB">
Reverb left (ms)                   $xx xx
Reverb right (ms)                  $xx xx
Reverb bounces, left               $xx
Reverb bounces, right              $xx
Reverb feedback, left to left      $xx
Reverb feedback, left to right     $xx
Reverb feedback, right to right    $xx
Reverb feedback, right to left     $xx
Premix left to right               $xx
Premix right to left               $xx
```

4.14.   Attached picture

This frame contains a picture directly related to the audio file.
Image format is the MIME type and subtype [MIME] for the image. In
the event that the MIME media type name is omitted, "image/" will be
implied. The "image/png" [PNG] or "image/jpeg" [JFIF] picture format
should be used when interoperability is wanted. Description is a
short description of the picture, represented as a terminated
text string. There may be several pictures attached to one file, each
in their individual "APIC" frame, but only one with the same content
descriptor. There may only be one picture with the picture type
declared as picture type $01 and $02 respectively. There is the
possibility to put only a link to the image file by using the 'MIME
type' "-->" and having a complete URL [URL] instead of picture data.
The use of linked files should however be used sparingly since there
is the risk of separation of files.

```
<Header for 'Attached picture', ID: "APIC">
Text encoding       $xx
MIME type           <text string> $00
Picture type        $xx
Description         <text string according to encoding> $00 (00)
Picture data        <binary data>
```

```
Picture type:  $00  Other
               $01  32x32 pixels 'file icon' (PNG only)
               $02  Other file icon
               $03  Cover (front)
               $04  Cover (back)
               $05  Leaflet page
```

```
                         $06   Media (e.g. label side of CD)
                         $07   Lead artist/lead performer/soloist
                         $08   Artist/performer
                         $09   Conductor
                         $0A   Band/Orchestra
                         $0B   Composer
                         $0C   Lyricist/text writer
                         $0D   Recording Location
                         $0E   During recording
                         $0F   During performance
                         $10   Movie/video screen capture
                         $11   A bright coloured fish
                         $12   Illustration
                         $13   Band/artist logotype
                         $14   Publisher/Studio logotype
```

4.15.   General encapsulated object

In this frame any type of file can be encapsulated. After the header,
'Frame size' and 'Encoding' follows 'MIME type' [MIME] represented as
as a terminated string encoded with ISO 8859-1 [ISO-8859-1]. The
filename is case sensitive and is encoded as 'Encoding'. Then follows
a content description as terminated string, encoded as 'Encoding'.
The last thing in the frame is the actual object. The first two
strings may be omitted, leaving only their terminations. MIME type is
always an ISO-8859-1 text string. There may be more than one "GEOB"
frame in each tag, but only one with the same content descriptor.

```
   <Header for 'General encapsulated object', ID: "GEOB">
   Text encoding          $xx
   MIME type              <text string> $00
   Filename               <text string according to encoding> $00 (00)
   Content description    <text string according to encoding> $00 (00)
   Encapsulated object    <binary data>
```

4.16.   Play counter

This is simply a counter of the number of times a file has been
played. The value is increased by one every time the file begins to
play. There may only be one "PCNT" frame in each tag. When the
counter reaches all one's, one byte is inserted in front of the
counter thus making the counter eight bits bigger. The counter must
be at least 32-bits long to begin with.

```
   <Header for 'Play counter', ID: "PCNT">
   Counter        $xx xx xx xx (xx ...)
```

4.17.   Popularimeter

The purpose of this frame is to specify how good an audio file is.
Many interesting applications could be found to this frame such as a
playlist that features better audio files more often than others or
it could be used to profile a person's taste and find other 'good'

files by comparing people's profiles. The frame contains the email
address to the user, one rating byte and a four byte play counter,
intended to be increased with one for every time the file is played.
The email is a terminated string. The rating is 1-255 where 1 is
worst and 255 is best. 0 is unknown. If no personal counter is wanted
it may be omitted. When the counter reaches all one's, one byte is
inserted in front of the counter thus making the counter eight bits
bigger in the same away as the play counter ("PCNT"). There may be
more than one "POPM" frame in each tag, but only one with the same
email address.

```
<Header for 'Popularimeter', ID: "POPM">
Email to user    <text string> $00
Rating           $xx
Counter          $xx xx xx xx (xx ...)
```

4.18.   Recommended buffer size

Sometimes the server from which an audio file is streamed is aware of
transmission or coding problems resulting in interruptions in the
audio stream. In these cases, the size of the buffer can be
recommended by the server using this frame. If the 'embedded info
flag' is true (1) then this indicates that an ID3 tag with the
maximum size described in 'Buffer size' may occur in the audio
stream. In such case the tag should reside between two MPEG [MPEG]
frames, if the audio is MPEG encoded. If the position of the next tag
is known, 'offset to next tag' may be used. The offset is calculated
from the end of tag in which this frame resides to the first byte of
the header in the next. This field may be omitted. Embedded tags are
generally not recommended since this could render unpredictable
behaviour from present software/hardware.

For applications like streaming audio it might be an idea to embed
tags into the audio stream though. If the clients connects to
individual connections like HTTP and there is a possibility to begin
every transmission with a tag, then this tag should include a
'recommended buffer size' frame. If the client is connected to a
arbitrary point in the stream, such as radio or multicast, then the
'recommended buffer size' frame SHOULD be included in every tag.

The 'Buffer size' should be kept to a minimum. There may only be one
"RBUF" frame in each tag.

```
<Header for 'Recommended buffer size', ID: "RBUF">
Buffer size                 $xx xx xx
Embedded info flag          %0000000x
Offset to next tag          $xx xx xx xx
```

4.19.   Audio encryption

This frame indicates if the actual audio stream is encrypted, and by
whom. Since standardisation of such encryption scheme is beyond this
document, all "AENC" frames begin with a terminated string with a
URL containing an email address, or a link to a location where an

email address can be found, that belongs to the organisation
responsible for this specific encrypted audio file. Questions
regarding the encrypted audio should be sent to the email address
specified. If a $00 is found directly after the 'Frame size' and the
audio file indeed is encrypted, the whole file may be considered
useless.

After the 'Owner identifier', a pointer to an unencrypted part of the
audio can be specified. The 'Preview start' and 'Preview length' is
described in frames. If no part is unencrypted, these fields should
be left zeroed. After the 'preview length' field follows optionally a
data block required for decryption of the audio. There may be more
than one "AENC" frames in a tag, but only one with the same 'Owner
identifier'.

```
<Header for 'Audio encryption', ID: "AENC">
Owner identifier    <text string> $00
Preview start       $xx xx
Preview length      $xx xx
Encryption info     <binary data>
```

4.20.   Linked information

To keep information duplication as low as possible this frame may be
used to link information from another ID3v2 tag that might reside in
another audio file or alone in a binary file. It is RECOMMENDED that
this method is only used when the files are stored on a CD-ROM or
other circumstances when the risk of file separation is low. The
frame contains a frame identifier, which is the frame that should be
linked into this tag, a URL [URL] field, where a reference to the
file where the frame is given, and additional ID data, if needed.
Data should be retrieved from the first tag found in the file to
which this link points. There may be more than one "LINK" frame in a
tag, but only one with the same contents. A linked frame is to be
considered as part of the tag and has the same restrictions as if it
was a physical part of the tag (i.e. only one "RVRB" frame allowed,
whether it's linked or not).

```
<Header for 'Linked information', ID: "LINK">
Frame identifier         $xx xx xx xx
URL                      <text string> $00
ID and additional data  <text string(s)>
```

Frames that may be linked and need no additional data are "ASPI",
"ETCO", "EQU2", "MCID", "MLLT", "OWNE", "RVA2", "RVRB", "SYTC", the
text information frames and the URL link frames.

The "AENC", "APIC", "GEOB" and "TXXX" frames may be linked with
the content descriptor as additional ID data.

The "USER" frame may be linked with the language field as additional
ID data.

The "PRIV" frame may be linked with the owner identifier as
additional ID data.

The "COMM", "SYLT" and "USLT" frames may be linked with three bytes
of language descriptor directly followed by a content descriptor as
additional ID data.

4.21.   Position synchronisation frame

This frame delivers information to the listener of how far into the
audio stream he picked up; in effect, it states the time offset from
the first frame in the stream. The frame layout is:

```
<Head for 'Position synchronisation', ID: "POSS">
Time stamp format          $xx
Position                   $xx (xx ...)
```

Where time stamp format is:

```
$01  Absolute time, 32 bit sized, using MPEG frames as unit
$02  Absolute time, 32 bit sized, using milliseconds as unit
```

and position is where in the audio the listener starts to receive,
i.e. the beginning of the next frame. If this frame is used in the
beginning of a file the value is always 0. There may only be one
"POSS" frame in each tag.

4.22.   Terms of use frame

This frame contains a brief description of the terms of use and
ownership of the file. More detailed information concerning the legal
terms might be available through the "WCOP" frame. Newlines are
allowed in the text. There may be more than one 'Terms of use' frame
in a tag, but only one with the same 'Language'.

```
<Header for 'Terms of use frame', ID: "USER">
Text encoding          $xx
Language               $xx xx xx
The actual text        <text string according to encoding>
```

4.23.   Ownership frame

The ownership frame might be used as a reminder of a made transaction
or, if signed, as proof. Note that the "USER" and "TOWN" frames are
good to use in conjunction with this one. The frame begins, after the
frame ID, size and encoding fields, with a 'price paid' field. The
first three characters of this field contains the currency used for
the transaction, encoded according to ISO 4217 [ISO-4217] alphabetic
currency code. Concatenated to this is the actual price paid, as a
numerical string using "." as the decimal separator. Next is an 8
character date string (YYYYMMDD) followed by a string with the name
of the seller as the last field in the frame. There may only be one
"OWNE" frame in a tag.

```
<Header for 'Ownership frame', ID: "OWNE">
```

```
     Text encoding      $xx
     Price paid         <text string> $00
     Date of purch.     <text string>
     Seller             <text string according to encoding>
```

4.24.   Commercial frame

This frame enables several competing offers in the same tag by
bundling all needed information. That makes this frame rather complex
but it's an easier solution than if one tries to achieve the same
result with several frames. The frame begins, after the frame ID,
size and encoding fields, with a price string field. A price is
constructed by one three character currency code, encoded according
to ISO 4217 [ISO-4217] alphabetic currency code, followed by a
numerical value where "." is used as decimal separator. In the price
string several prices may be concatenated, separated by a "/"
character, but there may only be one currency of each type.

The price string is followed by an 8 character date string in the
format YYYYMMDD, describing for how long the price is valid. After
that is a contact URL, with which the user can contact the seller,
followed by a one byte 'received as' field. It describes how the
audio is delivered when bought according to the following list:

```
     $00  Other
     $01  Standard CD album with other songs
     $02  Compressed audio on CD
     $03  File over the Internet
     $04  Stream over the Internet
     $05  As note sheets
     $06  As note sheets in a book with other sheets
     $07  Music on other media
     $08  Non-musical merchandise
```

Next follows a terminated string with the name of the seller followed
by a terminated string with a short description of the product. The
last thing is the ability to include a company logotype. The first of
them is the 'Picture MIME type' field containing information about
which picture format is used. In the event that the MIME media type
name is omitted, "image/" will be implied. Currently only "image/png"
and "image/jpeg" are allowed. This format string is followed by the
binary picture data. This two last fields may be omitted if no
picture is attached. There may be more than one 'commercial frame' in
a tag, but no two may be identical.

```
     <Header for 'Commercial frame', ID: "COMR">
     Text encoding       $xx
     Price string        <text string> $00
     Valid until         <text string>
     Contact URL         <text string> $00
     Received as         $xx
     Name of seller      <text string according to encoding> $00 (00)
     Description         <text string according to encoding> $00 (00)
     Picture MIME type   <string> $00
     Seller logo         <binary data>
```

4.25.   Encryption method registration

To identify with which method a frame has been encrypted the
encryption method must be registered in the tag with this frame. The
'Owner identifier' is a null-terminated string with a URL [URL]
containing an email address, or a link to a location where an email
address can be found, that belongs to the organisation responsible
for this specific encryption method. Questions regarding the
encryption method should be sent to the indicated email address. The
'Method symbol' contains a value that is associated with this method
throughout the whole tag, in the range $80-F0. All other values are
reserved. The 'Method symbol' may optionally be followed by
encryption specific data. There may be several "ENCR" frames in a tag
but only one containing the same symbol and only one containing the
same owner identifier. The method must be used somewhere in the tag.
See the description of the frame encryption flag in the ID3v2
structure document [ID3v2-strct] for more information.

```
<Header for 'Encryption method registration', ID: "ENCR">
Owner identifier     <text string> $00
Method symbol        $xx
Encryption data      <binary data>
```

4.26.   Group identification registration

This frame enables grouping of otherwise unrelated frames. This can
be used when some frames are to be signed. To identify which frames
belongs to a set of frames a group identifier must be registered in
the tag with this frame. The 'Owner identifier' is a null-terminated
string with a URL [URL] containing an email address, or a link to a
location where an email address can be found, that belongs to the
organisation responsible for this grouping. Questions regarding the
grouping should be sent to the indicated email address. The 'Group
symbol' contains a value that associates the frame with this group
throughout the whole tag, in the range $80-F0. All other values are
reserved. The 'Group symbol' may optionally be followed by some group
specific data, e.g. a digital signature. There may be several "GRID"
frames in a tag but only one containing the same symbol and only one
containing the same owner identifier. The group symbol must be used
somewhere in the tag. See the description of the frame grouping flag
in the ID3v2 structure document [ID3v2-strct] for more information.

```
<Header for 'Group ID registration', ID: "GRID">
Owner identifier       <text string> $00
Group symbol           $xx
Group dependent data   <binary data>
```

4.27.   Private frame

This frame is used to contain information from a software producer
that its program uses and does not fit into the other frames. The
frame consists of an 'Owner identifier' string and the binary data.

The 'Owner identifier' is a null-terminated string with a URL [URL]
containing an email address, or a link to a location where an email
address can be found, that belongs to the organisation responsible
for the frame. Questions regarding the frame should be sent to the
indicated email address. The tag may contain more than one "PRIV"
frame but only with different contents.

```
<Header for 'Private frame', ID: "PRIV">
Owner identifier      <text string> $00
The private data      <binary data>
```

4.28.   Signature frame

This frame enables a group of frames, grouped with the 'Group
identification registration', to be signed. Although signatures can
reside inside the registration frame, it might be desired to store
the signature elsewhere, e.g. in watermarks. There may be more than
one 'signature frame' in a tag, but no two may be identical.

```
<Header for 'Signature frame', ID: "SIGN">
Group symbol      $xx
Signature         <binary data>
```

4.29.   Seek frame

This frame indicates where other tags in a file/stream can be found.
The 'minimum offset to next tag' is calculated from the end of this
tag to the beginning of the next. There may only be one 'seek frame'
in a tag.

```
<Header for 'Seek frame', ID: "SEEK">
Minimum offset to next tag      $xx xx xx xx
```

4.30.   Audio seek point index

Audio files with variable bit rates are intrinsically difficult to
deal with in the case of seeking within the file. The ASPI frame
makes seeking easier by providing a list a seek points within the
audio file. The seek points are a fractional offset within the audio
data, providing a starting point from which to find an appropriate
point to start decoding. The presence of an ASPI frame requires the
existence of a TLEN frame, indicating the duration of the file in
milliseconds. There may only be one 'audio seek point index' frame in
a tag.

```
<Header for 'Seek Point Index', ID: "ASPI">
Indexed data start (S)          $xx xx xx xx
Indexed data length (L)         $xx xx xx xx
Number of index points (N)      $xx xx
Bits per index point (b)        $xx
```

Then for every index point the following data is included;

```
  Fraction at index (Fi)          $xx (xx)
```

'Indexed data start' is a byte offset from the beginning of the file.
'Indexed data length' is the byte length of the audio data being
indexed. 'Number of index points' is the number of index points, as
the name implies. The recommended number is 100. 'Bits per index
point' is 8 or 16, depending on the chosen precision. 8 bits works
well for short files (less than 5 minutes of audio), while 16 bits is
advantageous for long files. 'Fraction at index' is the numerator of
the fraction representing a relative position in the data. The
denominator is 2 to the power of b.

Here are the algorithms to be used in the calculation. The known data
must be the offset of the start of the indexed data (S), the offset
of the end of the indexed data (E), the number of index points (N),
the offset at index i (Oi). We calculate the fraction at index i
(Fi).

Oi is the offset of the frame whose start is soonest after the point
for which the time offset is (i/N * duration).

The frame data should be calculated as follows:

  $Fi = Oi/L * 2^b$ (rounded down to the nearest integer)

Offset calculation should be calculated as follows from data in the
frame:

  $Oi = (Fi/2^b)*L$ (rounded up to the nearest integer)

5. Copyright

Copyright (C) Martin Nilsson 2000. All Rights Reserved.

This document and translations of it may be copied and furnished to
others, and derivative works that comment on or otherwise explain it
or assist in its implementation may be prepared, copied, published
and distributed, in whole or in part, without restriction of any
kind, provided that a reference to this document is included on all
such copies and derivative works. However, this document itself may
not be modified in any way and reissued as the original document.

The limited permissions granted above are perpetual and will not be
revoked.

This document and the information contained herein is provided on an
"AS IS" basis and THE AUTHORS DISCLAIMS ALL WARRANTIES, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF
THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED
WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

6.   References

[CDDB] Compact Disc Data Base

    <url:http://www.cddb.com>

[ID3v2.3.0] Martin Nilsson, "ID3v2 informal standard".

    <url:http://www.id3.org/id3v2.3.0.txt>

[ID3v2-strct] Martin Nilsson,
"ID3 tag version 2.4.0 - Main Structure"

    <url:http//www.id3.org/id3v2.4.0-structure.txt>

[ISO-639-2] ISO/FDIS 639-2.
Codes for the representation of names of languages, Part 2: Alpha-3
code. Technical committee / subcommittee: TC 37 / SC 2

[ISO-4217] ISO 4217:1995.
Codes for the representation of currencies and funds.
Technical committee / subcommittee: TC 68

[ISO-8859-1] ISO/IEC DIS 8859-1.
8-bit single-byte coded graphic character sets, Part 1: Latin
alphabet No. 1. Technical committee / subcommittee: JTC 1 / SC 2

[ISRC] ISO 3901:1986
International Standard Recording Code (ISRC).
Technical committee / subcommittee: TC 46 / SC 9

[JFIF] JPEG File Interchange Format, version 1.02

    <url:http://www.w3.org/Graphics/JPEG/jfif.txt>

[KEYWORDS] S. Bradner, 'Key words for use in RFCs to Indicate
Requirement Levels', RFC 2119, March 1997.

    <url:ftp://ftp.isi.edu/in-notes/rfc2119.txt>

[MIME] Freed, N. and N. Borenstein, "Multipurpose Internet Mail
Extensions (MIME) Part One: Format of Internet Message Bodies",
RFC 2045, November 1996.

    <url:ftp://ftp.isi.edu/in-notes/rfc2045.txt>

[MPEG] ISO/IEC 11172-3:1993.
Coding of moving pictures and associated audio for digital storage
media at up to about 1,5 Mbit/s, Part 3: Audio.
Technical committee / subcommittee: JTC 1 / SC 29
 and
ISO/IEC 13818-3:1995
Generic coding of moving pictures and associated audio information,
Part 3: Audio.
Technical committee / subcommittee: JTC 1 / SC 29
 and
ISO/IEC DIS 13818-3
Generic coding of moving pictures and associated audio information,
Part 3: Audio (Revision of ISO/IEC 13818-3:1995)

```
     [PNG] Portable Network Graphics, version 1.0

        <url:http://www.w3.org/TR/REC-png-multi.html>

     [URL] T. Berners-Lee, L. Masinter & M. McCahill, "Uniform Resource
     Locators (URL).", RFC 1738, December 1994.

        <url:ftp://ftp.isi.edu/in-notes/rfc1738.txt>

     [ZLIB] P. Deutsch, Aladdin Enterprises & J-L. Gailly, "ZLIB
     Compressed
     Data Format Specification version 3.3", RFC 1950, May 1996.

        <url:ftp://ftp.isi.edu/in-notes/rfc1950.txt>


 7.    Appendix


 A.    Appendix A - Genre List from ID3v1

     The following genres is defined in ID3v1

         0.Blues
         1.Classic Rock
         2.Country
         3.Dance
         4.Disco
         5.Funk
         6.Grunge
         7.Hip-Hop
         8.Jazz
         9.Metal
        10.New Age
        11.Oldies
        12.Other
        13.Pop
        14.R&B
        15.Rap
        16.Reggae
        17.Rock
        18.Techno
        19.Industrial
        20.Alternative
        21.Ska
        22.Death Metal
        23.Pranks
        24.Soundtrack
        25.Euro-Techno
        26.Ambient
        27.Trip-Hop
        28.Vocal
        29.Jazz+Funk
        30.Fusion
```

```
                     31.Trance
                     32.Classical
                     33.Instrumental
                     34.Acid
                     35.House
                     36.Game
                     37.Sound Clip
                     38.Gospel
                     39.Noise
                     40.AlternRock
                     41.Bass
                     42.Soul
                     43.Punk
                     44.Space
                     45.Meditative
                     46.Instrumental Pop
                     47.Instrumental Rock
                     48.Ethnic
                     49.Gothic
                     50.Darkwave
                     51.Techno-Industrial
                     52.Electronic
                     53.Pop-Folk
                     54.Eurodance
                     55.Dream
                     56.Southern Rock
                     57.Comedy
                     58.Cult
                     59.Gangsta
                     60.Top 40
                     61.Christian Rap
                     62.Pop/Funk
                     63.Jungle
                     64.Native American
                     65.Cabaret
                     66.New Wave
                     67.Psychedelic
                     68.Rave
                     69.Showtunes
                     70.Trailer
                     71.Lo-Fi
                     72.Tribal
                     73.Acid Punk
                     74.Acid Jazz
                     75.Polka
                     76.Retro
                     77.Musical
                     78.Rock & Roll
                     79.Hard Rock
```

8.   Author's Address

   Written by

   Martin Nilsson

```
        Rydsv�gen 246 C. 30
        SE-584 34 Link�ping
        Sweden

        Email: nilsson at id3.org
```

id3v2.4.0-frames (last edited 2014-05-06 17:06:12 by DanONeill)